Preface

This volume contains a selection of 52 papers presented at the 19th International Conference on Domain Decomposition, DD19, hosted by the School of Mathematics and Computational Science of the Xiangtan University and the Hunan Key Laboratory for Computation and Simulation in Science and Engineering and held in Zhanjiajie, China, August 17–22, 2009. The conference featured 12 plenary lectures delivered by leaders in the field, 9 Minisymposia, and 33 contributed talks. 128 scientists from 21 countries participated and there were a total of 92 presentations, which accentuates the international scope and relevance of this meeting.

The International Conferences on Domain Decomposition Methods have become the most important market place world wide for exchanging and discussing new ideas about the old algorithmic paradigm of "Divide and Conquer". Since the beginning in Paris in 1987, they have been held in twelve countries in the Far East, Europe, the Middle East, and North America. Much of the reputation of this series results from the close interaction of experts in numerical analysis with practitioners from large scale scientific computing in various fields of applications.

In the time of "petascale" computers with more than 200,000 independent processor cores, there are essentially no alternatives to domain decomposition as a strategy for parallelization. The need for robust and efficient preconditioners thus motivates ongoing theoretical research on new Schwarz and iterative substructuring techniques for very large stationary problems arising in finite element simulations. The development of optimized transmission conditions, to enhance the rate of convergence of these iterative methods, remains a very active field and so does research on space-time domain decomposition. Moreover, different physical properties, in different subdomains, often suggest a splitting of the domain, e.g., into subdomains occupied by fluid or structure or even into bounded and unbounded domains that are glued together by suitable coupling conditions. This kind of heterogeneous domain decomposition has become a well-established approach to mathematical modeling.

We note that multigrid methods based on a decomposition into frequencies, rather than subdomains, can be used as subdomain solvers or as stand-alone methods for a variety of linear and nonlinear problems.

The present volume reviews many of these aspects of domain decomposition. Applications comprise acoustics, biomechanics, computational mechanics, fluid dynamics and fluid-structure interaction, electromagnetics, microelectronics, quantum dots and, of course, large scale computations.

For further information, we recommend the homepage of International Domain Decomposition Conferences, `www.ddm.org`, maintained by Martin Gander. This site features free online access to the proceedings of almost all previous DD conferences, information about past and future meetings, as well as bibliographic and personal information pertaining to domain decomposition. A bibliography with all previous proceedings is provided below, along with some major review articles and monographs. (We apologize for unintentional omissions to our necessarily incomplete list.) No attempts have been made to supplement this list with the larger and closely related literature of multigrid and general iterative methods, except for the books by Hackbusch and Saad, which have significant domain decomposition components.

References

1. M. Bercovier, M.J. Gander, R. Kornhuber, and O. Widlund, editors. *Domain Decomposition Methods in Science and Engineering XVIII*, Jerusalem, 2008. Springer, Heidelberg, 2009.
2. P. Bjørstad, M. Espedal, and D.E. Keyes, editors. *Proceedings of Ninth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Ullensvang, 1997. DDM.org, Bergen, 1998.
3. T.F. Chan, R. Glowinski, J. Périaux, and O.B. Widlund, editors. *Proceedings of the Second International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Los Angeles, 1988. SIAM, Philadelphia, PA, 1989.
4. T.F. Chan, R. Glowinski, J. Périaux, and O.B. Widlund, editors. *Proceedings of the Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Houston, 1989. SIAM, Philadelphia, PA, 1990.
5. T.F. Chan, T. Kako, H. Kawarada, and O. Pironneau, editors. *Proceedings of the Twelfth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Chiba, 1999. DDM.org, Bergen, 2001.
6. T.F. Chan and T.P. Mathew. Domain decomposition algorithms. *Acta Numer.*, 3: 61–143, 1994.
7. M. Débit, M. Garbey, R. Hoppe, D. Keyes, Y. Kuznetsov, and J. Périaux, editors. *Proceedings Thirteenth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Lyon, 2000. CINME, Barcelona, 2002.
8. C. Farhat and F.-X. Roux. Implicit parallel processing in structural mechanics. *Comput. Mech. Adv.*, 2:1–124, 1994.
9. R. Glowinski, G.H. Golub, G.A. Meurant, and J. Périaux, editors. *Proceedings First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Paris, 1987. SIAM, Philadelphia, PA, 1988.
10. R. Glowinski, Yu.A. Kuznetsov, G.A. Meurant, J. Périaux, and O.B. Widlund, editors. *Proceedings Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Moscow, 1990. SIAM, Philadelphia, PA, 1991.

11. R. Glowinski, J. Périaux, Z.-C. Shi, and O.B. Widlund, editors. *Proceedings of the Eighth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Beijing, 1995. Wiley, Strasbourg, 1997.
12. W. Hackbusch. *Iterative Methods for Large Sparse Linear Systems*. Springer, Heidelberg, 1993.
13. I. Herrera, D.E. Keyes, O.B. Widlund, and R. Yates, editors. *Proceedings of the Fourteenth International Conference on Domain Decomposition Methods in Science and Engineering*, Cocoyoc, 2002. UNAM, Mexico City, 2003.
14. D.E. Keyes, T.F. Chan, G.A. Meurant, J.S. Scroggs, and R.G. Voigt, editors. *Proceedings of the Fifth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Norfolk, 1991. SIAM, Philadelphia, PA, 1992.
15. D.E. Keyes, Y. Saad, and D.G. Truhlar, editors. *Domain-Based Parallelism and Problem Decomposition Methods in Science and Engineering*, SIAM, Philadelphia, PA, 1995.
16. D.E. Keyes and J. Xu, editors. *Proceedings Seventh International Conference on Domain Decomposition Methods for Partial Differential Equations*, PennState, 1993. AMS, Providence, RI, 1995.
17. B.N. Khoromskij and G. Wittum. *Numerical Solution of Elliptic Differential Equations by Reduction to the Interface*. Lecture Notes in Computational Science and Engineering, Vol. 36, Springer, 2004.
18. V.G. Korneev and U. Langer. Domain decomposition and preconditioning. In E. Stein, R. de Borst, and T.J.R. Hughes, editors, *Encyclopedia of Computational Mechanics*. Wiley, 2004.
19. R. Kornhuber, R.H.W. Hoppe, J. Périaux, O. Pironneau, O. Widlund, and J. Xu, editors. *Proceedings Fifteenth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Berlin, 2003. Springer, Heidelberg, 2004.
20. J. Kruis. *Domain Decomposition for Distributed Computing*. Dun Eaglais, Saxe Coburg, 2005.
21. C.-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, editors. *Proceedings Eleventh International Conference on Domain Decomposition Methods for Partial Differential Equations*, Greenwich, 1999. DDM.org, Bergen, 2000.
22. U. Langer, M. Discacciati, D.E. Keyes, O. Widlund, and W. Zulehner, editors. *Domain Decomposition Methods in Science and Engineering XVII*, Strobl, 2006. Springer, Heidelberg, 2008.
23. U. Langer and Steinbach O. Coupled finite element and boundary element domain decomposition. In M. Schanz and O. Steinbach, editors, *Boundary Element Analysis: Mathematical Aspects and Application*, pp. 29–59. Springer, Berlin, 2007.
24. P. Le Tallec. Domain decomposition methods in computational mechanics. *Comput. Mech. Adv.*, 2:121–220, 1994.
25. V.I. Lebedev and V.I. Agoshkov. Poincaré-Steklov operators and their applications in analysis. *Academy of Sciences*, URSS, Moscow, 1983 (In Russian).
26. J. Mandel, Ch. Farhat, and X.-Ch. Cai, editors. *Proceedings Tenth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Boulder, 1998. AMS, Providence, RI, 1999.
27. T.P.A. Mathew. *Domain Decomposition Methods for the Numerical Solution of Partial Differential Equations*, volume 61 of *Lecture Notes in Computational Science & Engineering*. Springer, Heidelberg, 2008.
28. S. Nepomnyaschikh. Domain decomposition methods. In J. Kraus and U. Langer, editors, *Lectures on Advanced Computational Methods in Mechanics*, Radon Series on Computational and Applied Mathematics. de Gruyter, Berlin, 2007.

29. P. Oswald. *Multilevel Finite Element Approximation: Theory and Applications*. Teubner Skripten zur Numerik. Teubner, Stuttgart, 1994.
30. L. Pavarino and A. Toselli. *Recent Developments in Domain Decomposition Methods*, volume 23 of *Lecture Notes in Computational Science & Engineering*, Springer, 2002.
31. A. Quarteroni, J. Périaux, Yu.A. Kuznetsov, and O.B. Widlund, editors. *Proceedings Sixth International Conference on Domain Decomposition Methods for Partial Differential Equations*, Como, 1992. AMS, Providence, RI, 1994.
32. A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*, Oxford University Press, Oxford, 1999.
33. Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS, Boston, MA, 1996.
34. B.F. Smith, P.E. Bjørstad, and W.D. Gropp. *Domain Decomposition: Parallel Multilevel Algorithms for Elliptic Partial Differential Equations*. Cambridge University Press, Cambridge, 1996.
35. O. Steinbach. *Stability Estimates for Hybrid Coupled Domain Decomposition Methods*, volume 1809 of *Lecture Notes in Mathematics*. Springer, Berlin, 2003.
36. A. Toselli and O. Widlund. *Domain Decomposition Methods*. Springer, Berlin, 2005.
37. O. Widlund and D.E. Keyes, editors. *Domain Decomposition Methods in Science and Engineering XVI*, New York, NY, 2005. Springer, Heidelberg, 2007.
38. B.I. Wohlmuth. *Discretization Methods and Iterative Solvers on Domain Decomposition*. Springer, Heidelberg, 2001.
39. J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.*, 34: 581–613, 1991.
40. J. Xu and J. Zou. Some nonoverlapping domain decomposition methods. *SIAM Rev.*, 40: 857–914, 1998.

The editors wish to thank all members of the International Scientific Committee for Domain Decomposition Conferences, now chaired by Ralf Kornhuber, for their help in setting the scientific direction of this conference. We are also grateful to the organizers of the minisymposia for shaping the profile of the scientific program and attracting high-quality presentations.

The organization was carried out by a local organizing committee from South China Normal University, Tsinghua University, the Chinese Academy of Sciences, and Xiangtan University. We thank all members and notably the chair Yunqing Huang for perfectly taking care of all aspects of preparing and running DD19. This included finding a first class conference venue that provided a relaxed atmosphere for exchanging information among attendees and lecturers as well as opportunities to enjoy the breathtaking countryside of Zhanjiajie. We gratefully acknowledge the financial and logistic support of this conference by the Hunan Key Laboratory for Computation and Simulation in Science and Engineering (LCSSE), the Institute for computational and Applied Mathematics (ICAM) of Xiangtan University, and the National Natural Science Foundation of China.

The timely production of these proceedings has been made possible by excellent cooperation of the authors and referees, who have all helped us to meet our deadlines. We gratefully acknowledge the diligent work of the technical editor Sabrina Nordt, who has compiled the final LaTeX source and the presentation of these proceedings

on the web; the FU Berlin has donated her service. Finally, we would like to thank Martin Peters and Thanh-Ha Le Thi of Springer for their friendly and efficient collaboration.

Yunqing Huang
Xiangtan University, China

Ralf Kornhuber
Freie Universität Berlin, Germany

Olof B. Widlund
Courant Institute, New York, USA

Jinchao Xu
Pennsylvania State University, USA

Contents

Part II Minisymposia

Contributors

List of Contributors

Blanca Ayuso Departamento de Matemáticas, Instituto de Ciencias Matemáticas CSIC-UAM-UC3M-UCM, Universidad Autónoma de Madrid, Madrid 28049, Spain, blanca.ayuso@uam.es

Randolph E. Bank Department of Mathematics, University of California, San Diego, CA 92093-0112, USA, rbank@ucsd.edu

Andrew T. Barker Department of Mathematics, Center for Computation and Technology, Louisiana State University, Baton Rouge, LA 70803-4918, USA, andrewb@math.lsu.edu

Heiko Berninger Fachbereich Mathematik und Informatik, Freie Universität Berlin, Berlin, Germany, berninge@math.fu-berlin.de

Mohamed El BouajajiNACHOS project-team, INRIA Sophia Antipolis – Méditerranée research center, F-06902 Sophia Antipolis Cedex, France, Mohamed.El_bouajaji@inria.fr

Filipa CaetanoDépartement de Mathématiques,

University of Paris-Sud, CNRS, Orsay F-91405, France,

filipa.caetano@math.u-psud.fr

Xiao-Chuan Cai Department of Computer Science, University

of Colorado, Boulder, CO 80309, USA, cai@cs.colorado.edu

Long Chen Department of Mathematics, University of

California, Irvine, CA 92697, USA, chenlong@math.uci.edu

Yanping Chen School of Mathematical Sciences, South China Normal University, Guangzhou 510631, P.R.China, yanpingchen@scnu.edu.cn

Dylan Copeland Institute for Applied Mathematics and Computational Science, Texas A&M University, College Station, TX, USA, copeland@math.tamu.edu

Biswa N. Datta Department of Math, Northern Illinois University, DeKalb, IL, USA, dattab@math.niu.edu

Stéphane Descombes Laboratoire J.-A. Dieudonné, Université de Nice Sophia-Antipolis, UMR CNRS 6621, 06018 Nice 02, France, Stéphane.Descombes@unice.fr

Thomas Dickopf Institute for Numerical Simulation, University of Bonn, 53115 Bonn, Germany, dickopf@ins.uni-bonn.de

Victorita Dolean Laboratoire J.-A. Dieudonné, University de Nice Sophia-Antipolis, UMR CNRS 6621, 06108 Nice Cedex 02, France, dolean@unice.fr; Victorita.Dolean@unice.fr

Maksymilian Dryja Department of Mathematics, Warsaw University, Warsaw 02-097, Poland, M.Dryja@mimuw.edu.pl

Yalchin Efendiev Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA, efendiev@math.tamu.edu

Jun FangLSEC, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China, fangjun@lsec.cc.ac.cn

Juan Galvis Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA, jugal@math.tamu.edu

Martin J. Gander Mathematics Section, University of Geneva, Geneva CH-1211, Switzerland, Martin.Gander@unige.ch; martin.gander@unige.ch; martin.gander@math.unige.ch

Xingyu GaoHPCC, Institute of Applied Physics and Computational Mathematics, Beijing 100094, China, gao xingyu@iapcm.ac.cn

Roland Glowinski Department of Mathematics, University of Houston, Houston, TX 77204, USA; Institute of Advanced Study, The Hong Kong University of Science and Technology, Kowloon, Hong Kong, angelarim@aol.com

Xingao Gong Department of Physics, Fudan University, Shanghai 200433, China, xggong@fudan.edu.cn

Loïc Gouarin Laboratoire Analyse, Géométrie et Applications UniversitéParis XIII, Villetaneuse 93430, France, gouarinhalpern@math.univ-paris13.fr

Laurence HalpernLAGA, UniversitéParis XIII, Villetaneuse 93430, France, halpern@math.univ-paris13.fr

Ronald D. Haynes Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's, NL, Canada A1C 5S7, rhaynes@mun.ca

Qiaolin He Department of Mathematics, The Hong Kong University of Science and Technology, Kowloon, Hong Kong, hqlaa@ust.hk

Yunqing Huang Hunan Key Laboratory for Computation and Simulation in Science and Engineering, School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, Hunan, P.R. China, huangyq@xtu.edu.cn

Tsung-Ming Huang Department of Mathematics, National Taiwan Normal University, Taipei 116, Taiwan, min@math.ntnu.edu.tw

Feng-Nan Hwang Department of Mathematics, National Central University, Jhongli 320, Taiwan, hwangf@math.ncu.edu.tw

Caroline JaphetLAGA, Université Paris XIII, 93430 Villetaneuse, France; CSCAMM, University of Maryland, College Park, MD 20742, USA, japhet@math.univ-paris13.fr; japhet@cscamm.umd.edu

Zhu Jialin College of Mathematics and Statistics, Chongqing University, Chongqing 400044, P.R. China

Lijian Jiang Institute for Mathematics and its Applications, University of Minnesota, Minneapolis, MN, USA, lijiang@ima.umn.edu

Takashi Kako Department of Computer Science, The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan, kako@im.uec.ac.jp

Hyea Hyun Kim Lawrence Livermore National Laboratory, Department of Mathematics, Chonnam National University, Gwangju, Korea, hyeahyun@gmail.com; hkim@jnu.ac.kr

Jonas KokoLIMOS, Université Blaise-Pascal – CNRS UMR 6158 Campus des Cézeaux, 63173 Aubière Cedex, France, koko@isima.fr

Michael Kolmbauer Institute of Computational Mathematics, Johannes Kepler University, Linz, Austria, kolmbauer@numa.uni-linz.ac.at

Ralf Kornhuber Fachbereich Mathematik und Informatik, Freie Universität Berlin, Berlin, Germany, kornhuber@math.fu-berlin.de

Daisuke Koyama The University of Electro-Communications, Chofu, Japan, koyama@im.uec.ac.jp

Rolf Krause Institute of Computational Science, University of Lugano, 6904 Lugano, Switzerland, rolf.krause@usi.ch

Radek Kučera Technical University of Ostrava, Ostrava, Czech Republic, radek.kucera@vsb.cz

Felix Kwok Section de mathématiques, Université de Genève, Geneva, Switzerland, Felix.Kwok@unige.ch

Ulrich Langer Institute of Computational Mathematics, Johannes Kepler University, Linz, Austria; Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Linz, Austria, ulanger@numa.uni-linz.ac.at; ulrich.langer@assoc.oeaw.ac.at

Stéphane LanteriNACHOS project-team, INRIA Sophia Antipolis – Méditerranée research center, F-06902 Sophia Antipolis Cedex, France, Stephane.Lanteri@inria.fr

Chang-Ock Lee Department of Mathematical Sciences, KAIST, Daejeon 305-701, South Korea, colee@kaist.edu

Si Liu Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309, USA, sliu@colorado.edu

Zuliang Lu College of Mathematics and Computer Sciences, Chongqing Three Gorges University, Chongqing 404000, P.R.China, zulianglux@126.com

Shiu Hong Lui Department of Mathematics, University of Manitoba, Winnipeg, MB R3T 2N2, Canada, luish@cc.umanitoba.ca

Jan Mandel Institute of Thermomechanics, Academy of Sciences of the Czech Republic, 182 00 Prague 8, Czech Republic; Department of Mathematical and Statistical Sciences, University of Colorado Denver, Denver, CO 80217, USA, jan.mandel@ucdenver.edu

Li Maojun College of Mathematics and Statistics, Chongqing University, Chongqing 400044, P.R. China, limaojun216@163.com

Veronique MartinLAMFA UMR-CNRS 6140, Université de Picardie Jules Verne, Amiens 80039, France, veronique.martin@u-picardie.fr

Hieu Nguyen Department of Mathematics, University of California, San Diego, CA 92093-0112, USA, htn005@math.ucsd.edu

Ricardo H. Nochetto Department of Mathematics, University of Maryland, College Park, MD, USA, rhn@math.umd.edu

Günther Of Institute of Computational Mathematics, TU Graz, Steyrergassse 30, A 8010 Graz, Austria, of@tugraz.at

Yoshiharu Ohi Department of Computer Science, The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan, ohi@sazae.im.uec.ac.jp

Eun-Hee Park Center for Computation and Technology,

Louisiana State University, Baton Rouge, LA 70803, USA,

epark2@cct.lsu.edu

Clemens Pechstein Institute of Computational Mathematics,

Johannes Kepler University, Linz 4040, Austria,

Clemens.Pechstein@jku.at

Wujian Peng Department of Math, Zhaoqing University,

Zhaoqing, China, douglas peng@yahoo.com

Ronan Perrussel Laboratoire Ampère, CNRS UMR 5005,

Ecully Cedex F-69134, France, ronan.perrussel@ec-lyon.fr

Julien RitonLMNO, University of Caen, Caen, France,

riton.julien@math.unicaen.fr

Oliver Sander Fachbereich Mathematik und Informatik, Freie
Universität Berlin, Berlin, Germany,
sander@math.fu-berlin.de

Marcus Sarkis Instituto Nacional de Matemática Pura e
Aplicada, Rio de Janeiro 22460-320, Brazil; Department of
Mathematical Sciences, Worcester Polytechnic Institute, Worcester,
MA 01609, USA, msarkis@wpi.edu

Taoufik SassiLMNO, Université de Caen – CNRS UMR
6139, 14032 Caen Cedex, France,
Taoufik.Sassi@math.unicaen.fr

Achim Schädle Mathematisches Institut,
Heinrich-Heine-Universität,
Düsseldorf D-40225,
Germany, schaedle@am.uni-duesseldorf.de

Robert Scheichl Department of Mathematical Sciences,
University of Bath, Claverton Down,
Bath BA2 7AY, UK,
R.Scheichl@bath.ac.uk

Shi Shu School of Mathematics and Computational Science,
Xiangtan University, Xiangtan 411105,
P.R. China, shushi@xtu.edu.cn

Bedřich Sousedík Department of Mathematical and
Statistical Sciences, University of Colorado Denver, Denver, CO
80217, USA; Institute of Thermomechanics, Academy of Sciences of the
Czech Republic, 182 00 Prague 8, Czech Republic,
bedrich.sousedik@ucdenver.edu

Olaf Steinbach Institute of Computational Mathematics, TU
Graz, A 8010 Graz, Austria,
o.steinbach@tugraz.at

Jérémie Szeftel Department of Mathematics,
Princeton University, Princeton, NJ 08544-1000, USA; C.N.R.S., MAB,
Université Bordeaux 1, 33405 Talence Cedex, France;
Département de mathématiques et applications, Ecole Normale
supérieure, 75230 Paris Cedex 05, France,
jszeftel@math.princeton.edu; Jeremie.Szeftel@ens.fr;
szeftel@dma.ens.fr

Panayot S. Vassilevski Center for Applied Scientific
Computing, Livermore, CA 94550, USA,
vassilevskil@llnl.gov

Weichung Wang Department of Mathematics, National Taiwan
University, Taipei 106, Taiwan,
wwang@math.ntu.edu.tw

Junxian Wang School of Mathematics and Computational
Science, Xiangtan University, Xiangtan 411105, P.R. China,
xianxian.student@sina.com

Huayi Wei Hunan Key Laboratory for Computation and
Simulation in Science and Engineering, School of Mathematics and
Computational Science, Xiangtan University, Xiangtan 411105, Hunan,
P.R. China, huayiwei1984@gmail.com

Zih-Hao Wei Department of Mathematics, National Central
University, Jhongli 320, Taiwan,
socrates.wei@gmail.com

Markus Windisch Institute of Computational Mathematics, TU
Graz, A 8010 Graz, Austria,
markus.windisch@tugraz.at

Jinchao Xu Department of Mathematics, Pennsylvania State
University, University Park, PA 16802,
USA, xu@math.psu.edu

Wei Yang Hunan Key Laboratory for Computation and
Simulation in Science and Engineering, School of Mathematics and
Computational Science, Xiangtan University, Xiangtan 411105, Hunan,
P.R. China, yangweixtu@126.com

Chao Yang Institute of Software, Chinese Academy of
Sciences, Beijing 100190, P.R. China,
yang@mail.rdcps.ac.cn

Huidong Yang Institute of Computational Mathematics,
Johannes Kepler University, 4040 Linz, Austria,
huidong@numa.uni-linz.ac.at

Haijian Yang College of Mathematics and Econometrics, Hunan
University, Changsha 410082, P.R. China, haijian.yang@colorado.edu

Nianyu Yi Hunan Key Laboratory for Computation and
Simulation in Science and Engineering, School of Mathematics and
Computational Science, Xiangtan University, Xiangtan 411105, Hunan,
P.R. China, yinianyu365109@126.com

Chen-Song Zhang Department of Mathematics, The Pennsylvania
State University, University Park, PA 16802, USA, zhangcs@psu.edu

Liuqiang Zhong School of Mathematics Sciences, South China
Normal University, Guangzhou 510631, P.R. China, zhonglq@xtu.edu.cn

Aihui ZhouLSEC, Institute of Computational Mathematics and
Scientific/Engineering Computing, Academy of Mathematics and Systems
Science, Chinese Academy of Sciences, Beijing 100190, China,
azhou@lsec.cc.ac.cn

Yunrong Zhu Department of Mathematics, University of
California, San Diego, CA 92093-0112, USA, zhu@math.ucsd.edu

Ludmil T. Zikatanov Department of Mathematics, Penn State
University, University Park, PA 16802, USA, ltz@math.psu.edu

Qingsong Zou Department of Scientific Computing and
Computer Applications, Sun Yat-sen University, Guangzhou 510275,
P.R. China, mcszqs@mail.sysu.edu.cn

Walter Zulehner Institute of Computational Mathematics,
Johannes Kepler University, 4040 Linz, Austria,
zulehner@numa.uni-linz.ac.at

Part I

Plenary Presentations

Domain Decomposition and hp-Adaptive Finite Elements

Randolph E. Bank [*1] and Hieu Nguyen [†2]

[1] Department of Mathematics, University of California, San Diego, La Jolla, CA 92093-0112, USA, rbank@ucsd.edu.
[2] Department of Mathematics, University of California, San Diego, La Jolla, CA 92093-0112, USA, htn005@math.ucsd.edu

1 Introduction

In this work, we report on an ongoing project to implement an hp-adaptive finite element method. The inspiration of this work came from the development of certain a posteriori error estimates for high order finite elements based on superconvergence [7, 8, 9]. We wanted to create an environment where these estimates could be evaluated in terms of their ability to estimate global errors for a wide range of problems, and to be used as the basis for adaptive enrichment algorithms.

Their use in a traditional h-refinement scheme for fixed degree p is straightforward, as is their use for mesh smoothing, again with fixed p. What is less clear and thus more interesting is their use in a traditional adaptive p-refinement scheme. One issue we hope to resolve, at least empirically, is the extent to which the superconvergence forming the foundation of these estimates continues to hold on meshes of variable degree. If superconvergence fails to hold globally (for example, in our preliminary experiments, superconvergence seems to hold in the interiors of regions of constant p but fails to hold along interfaces separating elements of different degrees), we would still like to determine if they remain robust enough to form the basis of an adaptive p-refinement algorithm.

As this is written, we have implemented in the PLTMG package [2] adaptive h-refinement/coarsening, adaptive p-refinement/coarsening, and adaptive mesh smoothing. These three procedures can be used separately, or mixed in arbitrary combinations. For example, one could compose an adaptive algorithm consisting of alternating steps of h and p-refinement. Since this requires that all procedures are able to process meshes with both variable h and p, many of the internal data structures

[*] The work of this author was supported by the U.S. National Science Foundation under contract DMS-0915220. The Beowulf cluster used for the numerical experiments was funded by NSF SCREMS-0619173.
[†] The work of this author was supported in part by a grant from the Vietnam Education Foundation (VEF).

and existing algorithms in the PLTMG package had to be generalized and extended. However, at present there remains open the more delicate and challenging issue of hp-refinement; that is, how to use these error estimates to decide if it is better to refine a given element into several child elements (h-refinement), or increase its degree (p-refinement). We hope to be able to report progress on this point at some time in the future.

Since PLTMG has options for parallel adaptive enrichment, this aspect also needs to be addressed. Fortunately, the parallel adaptive meshing paradigm implemented in PLTMG, see [1, 3, 4], formally works as well for p and hp-adaptivity as it does for h-adaptivity for which it was originally developed. As its final step, the paradigm requires the solution of a large global set of equations. A special DD algorithm (see [5, 6]) taking advantage of the structure of the parallel adaptive procedure was developed for this purpose.

2 A Posteriori Error Estimate

In the case of two dimensions, we consider an element t with vertices ν_i, and edges e_i, $1 \le i \le 3$, with e_i opposite ν_i. Let h_t denote the diameter of t. The barycentric coordinates for element t are denoted c_i, $1 \le i \le 3$. The restriction of the $\mathcal{C}^0$ piecewise polynomial space of degree $p \ge 1$ to element t consists of the $(p+1)(p+2)/2$-dimensional space $\mathcal{P}_p$ of polynomials of degree p, with degrees of freedom given by nodal values at the barycentric coordinates

$$(c_1, c_2, c_3) = (j/p, k/p, (p-j-k)/p)$$

for $0 \le j \le p$, $0 \le k \le p-j$.

Superconvergent derivative recovery schemes for this family of elements were developed in [7, 8, 9]. For the continuous piecewise polynomial space of degree p, let $\partial^p u_h$ denote any of the (discontinuous piecewise constant) p-th derivatives. The recovered p-th derivative is denoted by $R(\partial^p u_h) \equiv S^m Q(\partial^p u_h)$. Here Q is the L^2 projection from discontinuous piecewise constants into the space of continuous piecewise linear polynomials, and S is a multigrid smoother for the Laplace operator; m is a small integer, typically one or two. Under appropriate smoothness assumptions, it was shown that $\|\partial^p u - R(\partial^p u_h)\|$ has better than the first order convergence of $\|\partial^p (u - u_h)\|$.

To describe our a posteriori estimate for the case of an element of degree p, we write

$$\mathcal{P}_{p+1}(t) = \mathcal{P}_p(t) \oplus \mathcal{E}_{p+1}(t)$$

where the hierarchical extension $\mathcal{E}_{p+1}(t)$ consists of those polynomials in $\mathcal{P}_{p+1}(t)$ that are zero at all degrees of freedom associated with $\mathcal{P}_p(t)$. In the case of two dimensions, this is a subspace of dimension $p+2$, with a convenient basis given by

$$\psi_{p+1,k} = \prod_{j=0}^{k-1} (c_1 - j/p) \prod_{m=0}^{p-k} (c_2 - m/p)$$

for $0 \le k \le p+1$. Using this basis, we approximate the error $u - u_{h,p}$ on element t as

$$u - u_{h,p} \approx e_{h,p} \equiv \alpha_t \sum_{k=0}^{p+1} \frac{\partial_{c_1}^k \partial_{c_2}^{p+1-k} \hat{u}}{k!(p+1-k)!} \psi_{p+1,k}. \tag{1}$$

The partial derivatives of order $p+1$ appearing in (1) are formally $O(h_t^{p+1})$ when expressed in terms of ∂_x and ∂_y. The derivative $\partial_x^k \partial_y^{p+1-k} \hat{u}$ is constant on element t, computed by differentiating the recovered p-th derivatives of u_h, which are linear polynomials on element t.

$$\partial_x^k \partial_y^{p+1-k} \hat{u} = \begin{cases} \partial_y R(\partial_y^p u_h), & k = 0, \\ (\partial_x R(\partial_x^{k-1} \partial_y^{p+1-k} u_h) + \partial_y R(\partial_x^k \partial_y^{p-k} u_h))/2, & 1 \le k \le p, \\ \partial_x R(\partial_x^p u_h), & k = p+1. \end{cases}$$

The constant α_t is chosen such that

$$\sum_{k=0}^{p} \|\partial_x^k \partial_y^{p-k} e_{h,p}\|_t^2 = \sum_{k=0}^{p} \|\partial_x^k \partial_y^{p-k} u_h - R(\partial_x^k \partial_y^{p-k} u_h)\|_t^2$$

Normally, one should expect $\alpha_t \approx 1$, except for elements where the true solution u is not smooth enough to support p derivatives.

3 Basis Functions

One aspect of our study that is a bit unconventional is our use of nodal basis functions, rather than a hierarchical family of functions. The standard element of degree p uses standard nodal basis functions, as illustrated in Fig. 1, left. Along edges shared by elements of different degrees, the element of lower degree inherits the degrees of freedom of the higher degree element. This results in elements of degree p with one or two *transition edges* of higher degree. Some typical cases are illustrated in Fig. 1.

To illustrate the construction of the nodal basis for transition elements, consider the case of an element t of degree p with one transition edge of degree $p+1$. Without loss of generality take this to be edge three. We define one special polynomial of degree $p+1$, zero at all nodes of the standard element of degree p, and identically zero on edges one and two, by

$$\tilde{\phi}_{p+1} = \begin{cases} \prod_{k=0}^{(p-1)/2} (c_1 - k/p)(c_2 - k/p), & \text{for } p \text{ odd}, \\ (c_1 - c_2) \prod_{k=0}^{(p-2)/2} (c_1 - k/p)(c_2 - k/p), & \text{for } p \text{ even}. \end{cases}$$

The polynomial space for the transition element is given by $\mathcal{P}_p \oplus \{\tilde{\phi}_{p+1}\}$. We form linear combinations of $\tilde{\phi}_{p+1}$ and the $p+1$ standard nodal basis functions associated

with edge three to form the $p+2$ nodal basis functions for the transition edge. Because each of these $p+2$ polynomials is zero on edges one and two, and zero at all internal nodes for element t, all linear combinations of them also satisfy these properties, so the required calculation effectively reduces to a simple one-dimensional change of basis. If the edge is of degree $p+k$, the polynomial space is given by $\mathcal{P}_p \oplus \{\tilde{\phi}_{p+1}(c_1 - c_2)^m\}_{m=0}^{k-1}$, and a similar construction yields the required nodal basis for the transition edge. If a second transition edge is present, it is treated analogously. Because of our construction, each transition edge can be treated independently. It is also easy to see that the global finite element space constructed in this fashion is $\mathcal{C}^0$.

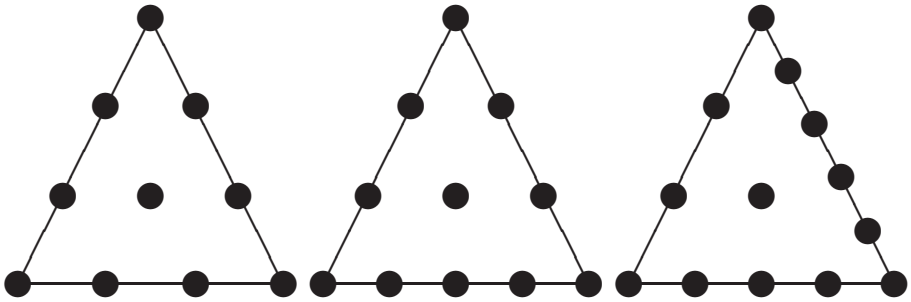


Fig. 1. A standard cubic element (*left*), a cubic element with one quartic edge (*middle*) and a cubic element with one quartic and one quintic edge (*right*).

4 Parallel Adaptive Algorithm

A general approach to parallel adaptive discretization for systems of elliptic partial differential equations was introduced in [3, 4]. This approach was motivated by the desire to keep communications costs low, and to allow sequential adaptive software such as PLTMG to be employed without extensive recoding.

The original paradigm has three main components:

Step I: Load Balancing. We solve a small problem on a coarse mesh, and use a posteriori error estimates to partition the mesh. Each subregion has approximately the same error, although subregions may vary considerably in terms of numbers of elements, or polynomial degree.

Step II: Adaptive Meshing. Each processor is provided the complete coarse problem and instructed to sequentially solve the *entire* problem, with the stipulation that its adaptive enrichment (h or p) should be limited largely to its own partition. The target number of degrees of freedom for each processor is the same. At the end of this step, the mesh is regularized such that the global finite element space described in Step III is conforming.

Step III: Global Solve. The final global problem consists of the union of the refined partitions provided by each processor. A final solution is computed using domain decomposition.

A variant of the above approach, in which the load balancing occurs on a much finer space, was described in [1]. The motivation was to address some possible problems arising from the use of a coarse grid in computing the load balance. This variant also has three main components.

Step I: Load Balancing. On a single processor we adaptively create a *fine* space of size N_P, and use a posteriori error estimates to partition the mesh such that each subregion has approximately equal error, similar to Step I of the original paradigm.
Step II: Adaptive Meshing. Each processor is provided the complete adaptive mesh and instructed to sequentially solve the *entire* problem. However, in this case each processor should adaptively *coarsen* regions corresponding to other processors, and adaptively enrich its own subregion. The size of the problem on each processor remains N_P, but this adaptive rezoning strategy concentrates the degrees of freedom in the processor's subregion. At the end of this step, the global space is made conforming as in the original paradigm.
Step III: Global Solve. This step is the same as in the original paradigm.

Using the variant, the initial mesh can be of any size. Indeed, our choice of N_P is mainly for convenience and to simplify notation; any combination of coarsening and refinement could be allowed in Step II.

5 DD Solver

Let $\Omega = \cup_{i=1}^{P} \Omega_i \subset \mathcal{R}^2$ denote the domain, decomposed into P geometrically conforming subdomains. Let Γ denote the interface system. The degree of a vertex x lying on Γ is the number of subregions for which $x \in \bar{\Omega}_i$. A cross point is a vertex $x \in \Gamma$ with degree$(x) \geq 3$. We assume that the maximal degree at cross points is bounded by the constant δ_0. The connectivity of Ω_i is the number of other regions Ω_j for which $\bar{\Omega}_i \cap \bar{\Omega}_j \neq \emptyset$. We assume the connectivity of Ω_i is bounded by the constant δ_1.

In our algorithm, we employ several triangulations. The mesh $\mathcal{T}$ is a globally refined, shape regular, and conforming mesh of size h. We assume that the fine mesh $\mathcal{T}$ is aligned with the interface system Γ. The triangulations $\mathcal{T}^i \subset \mathcal{T}$, $1 \leq i \leq P$ are partially refined triangulations; they coincide with the fine triangulation $\mathcal{T}$ within Ω_i, but are generally much coarser elsewhere, although as in the case for the variant paradigm, along the interface system Γ, $\mathcal{T}^i$ may have some intermediate level of refinement.

Let $\mathcal{S}$ denote the hp space of piecewise polynomials, associated with the triangulation $\mathcal{T}$, that are continuous in each of the Ω_i, but can be discontinuous along the interface system Γ. Let $\bar{\mathcal{S}} \subset \mathcal{S}$ denote the subspace of globally continuous piecewise polynomials. The usual basis for $\mathcal{S}$ is just the union of the nodal basis functions

corresponding to each of the subdomains Ω_i; such basis functions have their support in $\bar{\Omega}_i$ and those associated with nodes on Γ will have a jump at the interface. In our discussion, we will have occasion to consider another basis, allowing us to write $\mathcal{S} = \bar{\mathcal{S}} \oplus \mathcal{X}$, where $\mathcal{X}$ is a subspace associated exclusively with jumps on Γ. In particular, we will use the global conforming nodal basis for the space $\bar{\mathcal{S}}$, and construct a basis for $\mathcal{X}$ as follows. Let z_k be a node lying on Γ shared by two regions Ω_i and Ω_j (for now, z_k is not a crosspoint). Let $\phi_{i,k}$ and $\phi_{j,k}$ denote the usual nodal basis functions corresponding to z_k in Ω_i and Ω_j, respectively. The continuous nodal basis function for z_k in $\bar{\mathcal{S}}$ is $\phi_k \equiv \phi_{i,k} + \phi_{j,k}$, and the "jump" basis function in $\mathcal{X}$ is $\hat{\phi}_k \equiv \phi_{i,k} - \phi_{j,k}$. The direction of the jump is arbitrary at each z_k, but once chosen, will be used consistently. In this example, at point z_k we will refer to i and the "master" index and j as the "slave" index. At a cross point where $\ell > 2$ subregions meet, there will be one nodal basis function corresponding to $\bar{\mathcal{S}}$ and $\ell - 1$ jump basis functions. These are constructed by choosing one master index for the point, and making the other $\ell - 1$ indices slaves. We can construct $\ell - 1$ basis functions for $\mathcal{X}$ as $\phi_{i,k} - \phi_{j,k}$, where i is the master index and j is one of the slave indices.

For each of the triangulations $\mathcal{T}^i$, $1 \le i \le P$ we have a global nonconforming subspace $\mathcal{S}^i \subset \mathcal{S}$, and global conforming subspace $\bar{\mathcal{S}}^i \subset \bar{\mathcal{S}}$. In a fashion similar to $\mathcal{S}$, we have $\mathcal{S}^i = \bar{\mathcal{S}}^i \oplus \mathcal{X}^i$.

For simplicity, let the continuous variational problem be: find $u \in \mathcal{H}^1(\Omega)$ such that

$$a(u, v) = (f, v) \tag{2}$$

for all $v \in \mathcal{H}^1(\Omega)$, where $a(u, v)$ is a self-adjoint, positive definite bilinear form corresponding to the weak form of an elliptic partial differential equation, and $|\!|\!| u |\!|\!|_\Omega^2 = a(u, u)$ is comparable to the usual $\mathcal{H}^1(\Omega)$ norm.

To deal with the nonconforming nature of $\mathcal{S}$, for $u, v \in \mathcal{S}$, we decompose $a(u, v) = \sum_{i=1}^{P} a_{\Omega_i}(u, v)$. For each node z lying on Γ there is one master index and $\ell - 1 > 0$ slave indices. The total number of slave indices is denoted by K, so the total number of constraint equations in our nonconforming method is K. To simplify notation, for each $1 \le j \le K$, let $m(j)$ denote the corresponding master index, and z_j the corresponding node. We define the bilinear form $b(v, \lambda)$ by

$$b(v, \lambda) = \sum_{j=1}^{K} \{v_{m(j)} - v_j\} \lambda_j \tag{3}$$

where $\lambda \in \mathcal{R}^K$. In words, $b(\cdot, \cdot)$ measures the jump between the master value and each of the slave values at each node on Γ. The nonconforming variational formulation of (2) is: find $u_h \in \mathcal{S}$ such that

$$\begin{aligned} a(u_h, v) + b(v, \lambda) &= (f, v) \\ b(u_h, \xi) \qquad\quad &= 0 \end{aligned} \tag{4}$$

for all $v \in \mathcal{S}$ and $\xi \in \mathcal{R}^K$. Although this is formally a saddle point problem, the constraints are very simple; in particular, (4) simply imposes continuity at each of

the nodes lying on Γ, which in turn, implies that $u_h \in \bar{\mathcal{S}}$. Thus u_h also solves the reduced and conforming variational problem: find $u_h \in \bar{\mathcal{S}}$ such that

$$a(u_h, v) = (f, v)$$

for all $v \in \bar{\mathcal{S}}$.

Let $\mathcal{K}_i$ denote the index set of constraint equations in (3) that correspond to nodes present in $\mathcal{T}^i$. Then

$$b_i(v, \lambda) = \sum_{j \in \mathcal{K}_i} \{v_{m(j)} - v_j\} \lambda_j.$$

We are now in a position to formulate our domain decomposition algorithm. Our initial guess $u_0 \in \mathcal{S}$ is generated as follows: for $1 \le i \le P$, we find (in parallel) $u_{0,i} \in \bar{\mathcal{S}}^i$ satisfying

$$a(u_{0,i}, v) = (f, v) \tag{5}$$

for all $v \in \bar{\mathcal{S}}^i$. Here we assume exact solution of these local problems; in practice, these are often solved approximately using iteration. The initial guess $u_0 \in \mathcal{S}$ is composed by taking the part of $u_{0,i}$ corresponding to the fine subregion Ω_i for each i. In particular, let χ_i be the characteristic function for the subregion Ω_i. Then

$$u_0 = \sum_{i=1}^{P} \chi_i u_{0,i}$$

To compute $u_{k+1} \in \mathcal{S}$ from $u_k \in \mathcal{S}$, we solve (in parallel): for $1 \le i \le P$, find $e_{k,i} \in \mathcal{S}^i$ and $\lambda_{k,i} \in \mathcal{R}^K$ such that

$$\begin{aligned} a(e_{k,i}, v) + b_i(v, \lambda_{k,i}) &= (f, v) - a(u_k, v) \\ b_i(e_{k,i}, \xi) &= -b_i(u_k, \xi) \end{aligned} \tag{6}$$

for all $v \in \mathcal{S}^i$ and $\xi \in \mathcal{R}^K$. We then form

$$u_{k+1} = u_k + \sum_{i=1}^{P} \chi_i e_{k,i}.$$

Although the iterates u_k are elements of the nonconforming space $\mathcal{S}$, the limit function $u_\infty = u_h \in \bar{\mathcal{S}}$. In some sense, the purpose of the iteration is to drive the jumps in the approximate solution u_k to zero. Also, although (6) suggests a saddle point problem needs to be solved, by recognizing that only $\chi_i e_{k,i}$ is actually used, one can reduce (6) to a positive definite problem of the form (5). In particular, the Lagrange multipliers $\lambda_{k,i}$ need not be computed or updated.

The information required to be communicated among the processors is only the solution values and the residuals for nodes lying on Γ, which is necessary to compute the right hand sides of (6). This requires one all-to-all communication step at the beginning of each DD iteration.

6 Numerical Results

In this section, we present some numerical results. Our examples were run on a LINUX-based Beowulf cluster, consisting of 38 nodes, each with two quad core Xeon processors (2.33 GHz) and 16 GB of memory. The communication network is a gigabit Ethernet switch. This cluster runs the NPACI ROCKS version of LINUX and employs MPICH2 as its MPI implementation. The computational kernels of PLTMG [2] are written in FORTRAN; the *gfortran* compiler was used in these experiments, invoked using the script *mpif90* and optimization flag *-O*.

In these experiments, we used PLTMG to solve the boundary value problem

$$\begin{aligned} -\Delta u &= 1 \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned}$$

where Ω is a domain shaped like Lake Superior.

In our first experiment, the variant strategy was employed. A mesh of N_P degrees of freedom was created on a single processor using h-adaptive and p-adaptive refinement. Elements on this mesh had different sizes and degrees. This mesh was then broadcast to P processors, where a strategy of combined coarsening and refinement in both h and p was used to transfer approximately $N_P/2$ degrees of freedom from outside Ω_i to inside Ω_i. The global fine mesh was then made h-conforming (geometrically conforming) as described in [3, 4] and p-conforming (degrees agree on shared edges along the interface Γ). Note that the adaptive strategies implemented in PLTMG allow mesh moving and other modifications that yield meshes $\mathcal{T}_i$ that generally are *not* submeshes of the global conforming mesh $\mathcal{T}$ (by definition they are identical on Ω_i and $\partial\Omega_i$). However, PLTMG does insure that the partitions remain geometrically conforming, even in the coarse parts of the domain, and in particular, that the vertices on the interface system in each $\mathcal{T}_i$ are a subset of the vertices of interface system of the global mesh $\mathcal{T}$.

In this experiment, three values of N_P (400, 600, and 800 K), and eight values of P (2^k, $1 \le k \le 8$) were used, yielding global fine meshes ranging in size from about 626 K to 96.5 M unknowns. Because our cluster had only 38 nodes, for larger values of P, we simulated the behavior of a larger cluster in the usual way, by allowing nodes to have multiple processes.

In these experiments, the convergence criterion was

$$\frac{\|\delta\mathcal{U}^k\|_G}{\|\mathcal{U}^k\|_G} \le \frac{\|\delta\mathcal{U}^0\|_G}{\|\mathcal{U}^0\|_G} \times 10^{-3}. \tag{7}$$

This is more stringent than necessary for purposes of computing an approximation to the solution of the partial differential equation, but it allows us to illustrate the behavior of the solver as an iterative method for solving linear systems of equations.

Table 1 summarizes this computation. The columns labeled DD indicate the number of domain decomposition iterations required to satisfy the convergence criteria (7). For comparison, the number of iterations needed to satisfy the actual convergence criterion used in PLTMG, based on reducing the error in the solution of the

linear system to the level of the underlying approximation error, is given in parentheses. From these results it is clear that the number of iterations is stable and largely independent of N and P over this range of values. The size of the global mesh for the variant strategy can be estimated from the formula

$$N \approx \theta P N_P + N_P \tag{8}$$

where $\theta = 1/2$. Equation (8) predicts an upper bound, as it does not account for refinement outside of Ω_i and coarsening inside Ω_i, needed to keep the mesh conforming and for other reasons. For $N_P = 800\,\mathrm{K}$, $P = 256$, (8) predicts $N \approx 103,200,000$, where the observed $N = 96{,}490{,}683$.

Table 1. Convergence results for variant algorithm. Numbers of iterations needed to satisfy (7) are given in the column labeled DD. The numbers in parentheses are the number of iterations required to satisfy the actual convergence criterion used by PLTMG.

	$N_P = 400$ K		$N_P = 600$ K		$N_P = 800$ K	
P	N	DD	N	DD	N	DD
2	625,949	10 (3)	776,381	8 (3)	1,390,124	12 (4)
4	1,189,527	13 (4)	1,790,918	11 (4)	2,288,587	9 (3)
8	1,996,139	10 (4)	2,990,807	13 (4)	3,993,126	10 (3)
16	3,569,375	14 (4)	5,220,706	13 (4)	6,920,269	12 (3)
32	6,723,697	13 (3)	9,736,798	16 (4)	13,142,670	11 (3)
64	12,978,568	11 (4)	18,905,909	14 (4)	25,326,662	11 (3)
128	25,155,124	12 (3)	37,148,571	10 (4)	48,841,965	10 (3)
256	48,874,991	11 (3)	72,902,698	14 (4)	96,490,683	11 (3)

In our second experiment we solved the same problem using the original paradigm. On one processor, an adaptive mesh of size $N_c = 50\,\mathrm{K}$ was created. All elements on this mesh were linear elements. This mesh was then partitioned into P subregions, $P = 2^k$, $1 \le k \le 8$. This coarse mesh was broadcast to P processors (simulated as needed) and each processor continued the adaptive process in both h and p, creating a mesh of size N_P. In this experiment, N_P was chosen to be 400, 600, and $800\,\mathrm{K}$. This resulted in global meshes varying in size from approximately $750\,\mathrm{K}$ to $189\,\mathrm{M}$. These global meshes were regularized to be h-conforming and p-conforming, and a global DD solve was made as in the first experiment. As in the first experiment, the usual convergence criteria was replaced by (7) in order to illustrate the dependence of the convergence rate on N and P. The results are summarized in Table 2.

For the original paradigm the size of the global mesh is predicted by

$$N \approx P N_P - (P-1) N_c. \tag{9}$$

Similar to Eq. (8), Eq. (9) only predicts an upper bound, as it does not account for refinement outside of Ω_i, needed to keep the mesh conforming and for other reasons. For example, for $N_c = 50\,\mathrm{K}$, $N_P = 800\,\mathrm{K}$, $P = 256$, (9) predicts

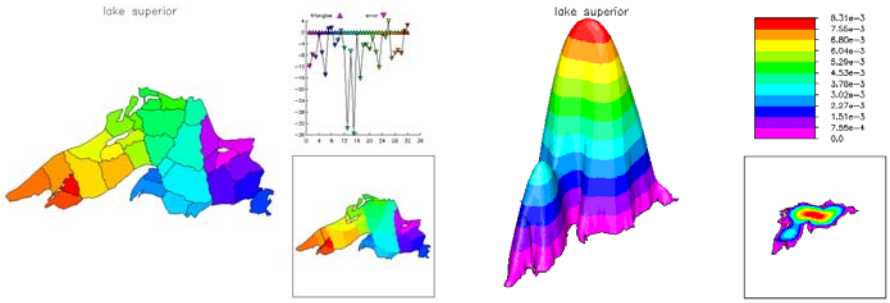


Fig. 2. The load balance (*left*) and solution (*right*) in the case $N_P = 800\,\mathrm{K}$, $P = 32$.

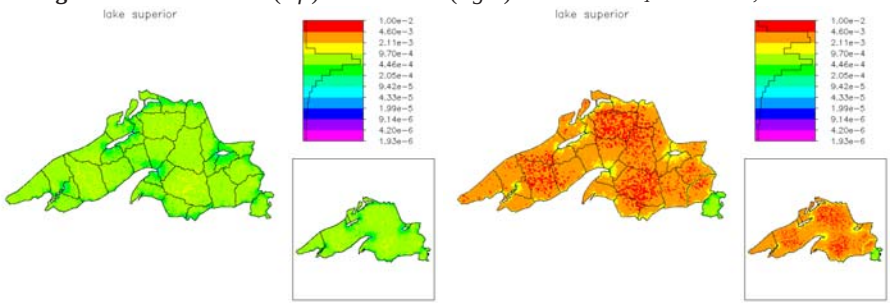


Fig. 3. The mesh density for the global mesh (*left*) and for one of the local meshes (*right*) in the case $N_P = 800\,\mathrm{K}$, $P = 32$.

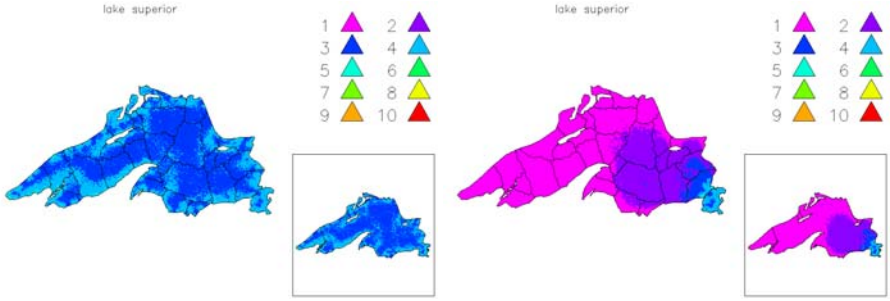


Fig. 4. The degree density for the global mesh (*left*) and for one of the local meshes (*right*) in the case $N_P = 800\,\mathrm{K}$, $P = 32$.

$N \approx 192,050,000$ when actually $N = 189,363,322$. For the case $N_P = 800\,\mathrm{K}$, $P = 32$, the solution and the load balance is shown in Fig. 2. The mesh density and degree density of the global mesh and one local mesh are shown in Figs. 3 and 4. As expected, both the mesh density and the degree density are high in the local region and much lower elsewhere in the local mesh.

Table 2. Convergence results for original Algorithm. Numbers of iterations needed to satisfy (7) are given in the column labeled DD. The numbers in parentheses are the number of iterations required to satisfy the actual convergence criterion used by PLTMG.

	$N_P = 400\,\mathrm{K}$		$N_P = 600\,\mathrm{K}$		$N_P = 800\,\mathrm{K}$	
P	N	DD	N	DD	N	DD
2	750,225	13 (4)	1,150,106	13 (4)	1,549,915	13 (4)
4	1,450,054	13 (4)	2,248,841	13 (4)	3,047,906	13 (4)
8	2,846,963	9 (3)	4,442,665	9 (4)	6,039,743	9 (3)
16	5,635,327	11 (4)	8,821,463	10 (4)	12,010,188	11 (4)
32	11,204,214	12 (4)	17,564,640	10 (4)	23,930,867	11 (4)
64	22,301,910	14 (4)	34,983,543	13 (4)	47,693,190	13 (4)
128	44,408,605	11 (4)	69,696,605	12 (4)	95,026,759	11 (4)
256	88,369,503	11 (3)	138,790,801	11 (3)	189,363,322	11 (4)

References

1. R.E. Bank. Some variants of the Bank-Holst parallel adaptive meshing paradigm. *Comput. Vis. Sci.*, 9(3):133–144, 2006. ISSN 1432-9360. URL http://dx.doi.org/10.1007/s00791-006-0029-6.
2. R.E. Bank. PLTMG: A software package for solving elliptic partial differential equations, users' guide 10.0. Technical Report, Department of Mathematics, University of California at San Diego, 2007. URL http://ccom.ucsd.edu/~reb.
3. R.E. Bank and M. Holst. A new paradigm for parallel adaptive meshing algorithms. *SIAM J. Sci. Comput.*, 22(4):1411–1443 (electronic), 2000. ISSN 1064-8275. URL http://dx.doi.org/10.1137/S1064827599353701.
4. R.E. Bank and M. Holst. A new paradigm for parallel adaptive meshing algorithms. *SIAM Rev.*, 45(2):291–323 (electronic), 2003. ISSN 0036-1445. URL http://dx.doi.org/10.1137/S003614450342061. Reprinted from SIAM J. Sci. Comput. 22 (2000), no. 4, 1411–1443 [MR1797889].
5. R.E. Bank and S. Lu. A domain decomposition solver for a parallel adaptive meshing paradigm. *SIAM J. Sci. Comput.*, 26(1):105–127 (electronic), 2004. ISSN 1064-8275. URL http://dx.doi.org/10.1137/S1064827503428096.
6. R.E. Bank and P.S. Vassilevski. Convergence analysis of a domain decomposition paradigm. *Comput. Vis. Sci.*, 11(4–6):333–350, 2008. ISSN 1432-9360. URL http://dx.doi.org/10.1007/s00791-008-0103-3.
7. R.E. Bank and J. Xu. Asymptotically exact a posteriori error estimators. I. Grids with superconvergence. *SIAM J. Numer. Anal.*, 41(6):2294–2312 (electronic), 2003. ISSN 0036-1429. URL http://dx.doi.org/10.1137/S003614290139874X.
8. R.E. Bank and J. Xu. Asymptotically exact a posteriori error estimators. II. General unstructured grids. *SIAM J. Numer. Anal.*, 41(6):2313–2332 (electronic), 2003. ISSN 0036-1429. URL http://dx.doi.org/10.1137/S0036142901398751.
9. R.E. Bank, J. Xu, and B. Zheng. Superconvergent derivative recovery for Lagrange triangular elements of degree p on unstructured grids. *SIAM J. Numer. Anal.*, 45(5):2032–2046 (electronic), 2007. ISSN 0036-1429. URL http://dx.doi.org/10.1137/060675174.

Domain Decomposition Methods for Electromagnetic Wave Propagation Problems in Heterogeneous Media and Complex Domains

Victorita Dolean[1], Mohamed El Bouajaji[2], Martin J. Gander[3], Stéphane Lanteri[2], and Ronan Perrussel[4]

[1] Laboratoire J.A. Dieudonné, CNRS UMR 6621, F-06108 Nice Cedex, France
[2] NACHOS Project-Team, INRIA Sophia Antipolis – Méditerranée Research Center, F-06902 Sophia Antipolis Cedex, France, Stephane.Lanteri@inria.fr
[3] Mathematics Section, University of Geneva, CH-1211, Geneva, Switzerland
[4] Laboratoire Ampère, CNRS UMR 5005, F-69134 Ecully Cedex, France

1 Introduction

We are interested here in the numerical modeling of time-harmonic electromagnetic wave propagation problems in irregularly shaped domains and heterogeneous media. In this context, we are naturally led to consider volume discretization methods (i.e. finite element method) as opposed to surface discretization methods (i.e. boundary element method). Most of the related existing work deals with the second order form of the time-harmonic Maxwell equations discretized by a conforming finite element method [14]. More recently, discontinuous Galerkin (DG) methods have also been considered for this purpose. While the DG method keeps almost all the advantages of a conforming finite element method (large spectrum of applications, complex geometries, etc.), the DG method has other nice properties which explain the renewed interest it gains in various domains in scientific computing: easy extension to higher order interpolation (one may increase the degree of the polynomials in the whole mesh as easily as for spectral methods and this can also be done locally), no global mass matrix to invert when solving time-domain systems of partial differential equations using an explicit time discretization scheme, easy handling of complex meshes (the mesh may be a classical conforming finite element mesh, a non-conforming one or even a mesh made of various types of elements), natural treatment of discontinuous solutions and coefficient heterogeneities and nice parallelization properties. In this paper, the first order form of the time-harmonic Maxwell equations is discretized using a high order DG method formulated on unstructured simplicial meshes.

Domain decomposition (DD) methods are flexible and powerful techniques for the parallel numerical solution of systems of partial differential equations. Their application to time-harmonic wave propagation problems began with a first algorithm proposed in [4] for solving the Helmholtz equation, and then was extended and

generalized for the time-harmonic Maxwell equations in [1, 3, 5]. A classical DD strategy which takes the form of a Schwarz algorithm where Després type conditions are imposed at the interfaces between neighboring subdomains was adopted in our previous work [8]. These conditions actually translate into a continuity condition for the incoming characteristic variables in the case of the first order Maxwell system. A similar approach (using Robin transmission conditions) but applied to a second order form of the Maxwell system, and in conjunction with a non-conforming finite element discretization, is presented in [13, 18]. The analysis of a larger class of Schwarz algorithms has been performed recently in [7] where optimized transmission conditions are used. The latter extends the idea of the most general, optimized interface conditions designed for the Helmholtz problem in [12].

In this paper, we consider classical and optimized Schwarz algorithms studied in [7], in conjunction with high order DG methods [6] formulated on unstructured simplicial meshes, for the solution of the time-harmonic Maxwell equations. The rest of this paper is organized as follows. In Sect. 2, we formulate the continuous boundary value problem to be solved. Then, in Sect. 3, the adopted Schwarz DD method is introduced. Section 4 is devoted to the discretization of the global and domain decomposed boundary value problems. Finally, in Sect. 5, numerical strategies for solving local problems as well as parallel computing aspects are discussed and experimental results are presented.

2 Continuous Problem

The system of normalized time-harmonic Maxwell's equations is given by:

$$\mathrm{i}\omega\varepsilon_r\mathbf{E} - \operatorname{curl}\mathbf{H} = -\mathbf{J}, \qquad \mathrm{i}\omega\mu_r\mathbf{H} + \operatorname{curl}\mathbf{E} = 0, \tag{1}$$

where $\mathbf{E}$ and $\mathbf{H}$ are the unknown electric and magnetic fields and $\mathbf{J}$ is a known current source; ε_r and μ_r respectively denote the relative electric permittivity and the relative magnetic permeability; we consider here the case of linear isotropic media. The angular frequency of the problem is given by ω. Equations (1) are solved in a bounded domain Ω. On the boundary $\partial\Omega = \Gamma_a \cup \Gamma_m$, the following boundary conditions are imposed:

$$\begin{aligned} &\text{– a perfect electric conductor (PEC) condition on } \Gamma_m : \mathbf{n}\times\mathbf{E} = 0, \\ &\text{– a first order absorbing condition on } \Gamma_a : \mathcal{L}(\mathbf{E},\mathbf{H}) = \mathcal{L}(\mathbf{E}^{\mathrm{inc}},\mathbf{H}^{\mathrm{inc}}), \end{aligned} \tag{2}$$

where $\mathcal{L}(\mathbf{E},\mathbf{H}) = \mathbf{n}\times\mathbf{E} - Z\mathbf{n}\times(\mathbf{H}\times\mathbf{n})$ with $Z = \sqrt{\mu_r/\varepsilon_r}$. The vectors $\mathbf{E}^{\mathrm{inc}}$ and $\mathbf{H}^{\mathrm{inc}}$ represent the components of an incident electromagnetic wave and $\mathbf{n}$ denotes the unit outward normal. Equations (1) and (2) can be further rewritten (assuming $\mathbf{J}$ equals to 0 to simplify the presentation) in the form:

$$\begin{cases} \mathrm{i}\omega G_0\mathbf{W} + G_x\partial_x\mathbf{W} + G_y\partial_y\mathbf{W} + G_z\partial_z\mathbf{W} = 0 \text{ in } \Omega, \\ (M_{\Gamma_m} - G_{\mathbf{n}})\mathbf{W} = 0 \text{ on } \Gamma_m, \\ (M_{\Gamma_a} - G_{\mathbf{n}})(\mathbf{W} - \mathbf{W}^{\mathrm{inc}}) = 0 \text{ on } \Gamma_a, \end{cases} \tag{3}$$

where $\mathbf{W} = (\mathbf{E}, \mathbf{H})^T$ is the new unknown vector and:

$$G_0 = \begin{pmatrix} \varepsilon_r \, \mathrm{I}_3 & 0_3 \\ 0_3 & \mu_r \, \mathrm{I}_3 \end{pmatrix}, \quad G_l = \begin{pmatrix} 0_3 & N_{\mathbf{e}^l} \\ N_{\mathbf{e}^l}^T & 0_3 \end{pmatrix}, \quad N_{\mathbf{v}} = \begin{pmatrix} 0 & v_z & -v_y \\ -v_z & 0 & v_x \\ v_y & -v_x & 0 \end{pmatrix},$$

with the index set $l \in \{x, y, z\}$ for G_l and where $(\mathbf{e^x}, \mathbf{e^y}, \mathbf{e^z})$ is the canonical basis of $\mathbb{R}^3$ and $\mathbf{v} = (v_x, v_y, v_z)^T$. The term I_3 denotes the identity matrix, and 0_3 the null matrix, both of dimension 3×3. The real part of G_0 is symmetric positive definite and its imaginary part, which appears for instance in the case of conductive materials, is symmetric negative definite. In the following we denote by $G_{\mathbf{n}}$ the sum $G_x n_x + G_y n_y + G_z n_z$ and by $G_{\mathbf{n}}^+$ and $G_{\mathbf{n}}^-$ its positive and negative parts.[1] We also define $|G_{\mathbf{n}}| = G_{\mathbf{n}}^+ - G_{\mathbf{n}}^-$. In order to take into account the boundary conditions, the matrices M_{Γ_m} and M_{Γ_a} are given:

$$M_{\Gamma_m} = \begin{pmatrix} 0_3 & N_{\mathbf{n}} \\ -N_{\mathbf{n}}^T & 0_3 \end{pmatrix} \quad \text{and} \quad M_{\Gamma_a} = |G_{\mathbf{n}}|.$$

3 A Family of Schwarz DD Algorithms

We assume that the domain Ω is decomposed into N_s subdomains $\Omega = \bigcup_{i=1}^{N_s} \Omega_i$ and let $\Gamma_{ij} = \partial\Omega_i \cap \overline{\Omega}_j$. In the following, a superscript i indicates that some notations are relative to the subdomain Ω_i and not to the whole domain Ω. We denote by $\mathbf{n}_{ij}$ the unit outward normal vector to the interface Γ_{ij}. We consider a family of Schwarz DD algorithms for solving the problem (3), given by (n denotes the Schwarz iteration):

$$\begin{cases} i\omega G_0 \mathbf{W}^{i,n+1} + \displaystyle\sum_{l \in \{x,y,z\}} G_l \partial_l \mathbf{W}^{i,n+1} = 0 \text{ in } \Omega_i, \\ \mathcal{B}_{\mathbf{n}_{ij}} \mathbf{W}^{i,n+1} = \mathcal{B}_{\mathbf{n}_{ij}} \mathbf{W}^{j,n} \text{ on } \Gamma_{ij}, \\ + \text{B.C. on } \partial\Omega_i \cap \partial\Omega, \end{cases} \tag{4}$$

where the $B_{\mathbf{n}_{ij}}$ are interface operators. Such algorithms have been studied in detail in [7] with the aim of designing optimized overlapping and non-overlapping Schwarz methods for both the time-domain and time-harmonic Maxwell equations. Here, we consider the following situations:

- the classical Schwarz algorithm (for 2D and 3D problems) in which $\mathcal{B}_{\mathbf{n}_{ij}} \equiv G_{\mathbf{n}_{ij}}^-$,
- an optimized Schwarz algorithm (for 2D problems only) characterized by $\mathcal{B}_{\mathbf{n}_{ij}} \equiv G_{\mathbf{n}_{ij}}^- + \mathcal{S}_i G_{\mathbf{n}_{ji}}^-$ with $\mathcal{S}_i = \alpha_i = (i\tilde{\omega})^{-1}[p(1-i)]$ where $\tilde{\omega} = \omega\sqrt{\varepsilon\mu}$.

The optimized Schwarz algorithm selected in this study corresponds to one of several variants proposed and analyzed in [7]. In particular, in the case of a two-subdomain non-overlapping decomposition, a good choice is $p = \frac{\sqrt{C} C^{\frac{1}{4}}_{\tilde{\omega}}}{\sqrt{2}\sqrt{h}}$, which

[1] If $T\Lambda T^{-1}$ is the eigendecomposition of $G_{\mathbf{n}}$, then $G_{\mathbf{n}}^{\pm} = T\Lambda^{\pm}T^{-1}$ where Λ^+ (respectively Λ^-) only gathers the positive (respectively negative) eigenvalues.

leads to the asymptotic convergence factor $\rho = 1 - \frac{\sqrt{2}C_{\tilde{\omega}}^{\frac{1}{4}}}{\sqrt{C}}\sqrt{h}$ (while $\rho = 1$ for the classical Schwarz algorithm in this configuration) where C is a constant and $C_{\tilde{\omega}} = \min(k_+^2 - \tilde{\omega}^2, \tilde{\omega}^2 - k_-^2)$ (k_- and k_+ are frequency parameters, see [7] for more details). Preliminary results on the use of this optimized Schwarz algorithm in conjunction with a high order DG method were presented in [9].

4 Discretization by a High Order DG Method

The subproblems of the Schwarz algorithm (4) are discretized using a DG formulation. In this section, we first introduce this discretization method in the one-domain case. Then we establish the discretization of the interface condition of algorithm (4) with respect to the adopted DG formulation. Let Ω_h denote a discretization of the domain Ω into a union of conforming simplicial elements K. We look for the approximate solution $\mathbf{W}_h$ of (3) in $V_h \times V_h$ where the functional space V_h is defined by $V_h = \left\{\mathbf{U} \in [L^2(\Omega)]^3 \,/\, \forall K \in \Omega_h,\ \mathbf{U}_{|K} \in \mathbb{P}_p(K)\right\}$, where $\mathbb{P}_p(K)$ denotes a space of vectors with polynomial components of degree at most p over the element K.

4.1 Discretization of the Monodomain Problem

The DG discretization of system (3) yields the formulation of the discrete problem which aims at finding $\mathbf{W}_h$ in $V_h \times V_h$ such that:

$$\left\{\begin{array}{l} \displaystyle\int_{\Omega_h} (\mathrm{i}\omega G_0 \mathbf{W}_h)^T \overline{\mathbf{V}} dv + \sum_{K\in\Omega_h} \int_K \left(\sum_{l\in\{x,y,z\}} G_l \partial_l(\mathbf{W}_h)\right)^T \overline{\mathbf{V}} dv \\ \displaystyle + \sum_{F\in\Gamma^m\cup\Gamma^a} \int_F \left(\frac{1}{2}(M_{F,K} - I_{FK} G_{\mathbf{n}_F})\mathbf{W}_h\right)^T \overline{\mathbf{V}} ds \\ \displaystyle - \sum_{F\in\Gamma^0} \int_F (G_{\mathbf{n}_F} [\![\, \mathbf{W}_h \,]\!])^T \{\overline{\mathbf{V}}\} ds + \sum_{F\in\Gamma^0} \int_F (S_F [\![\, \mathbf{W}_h \,]\!])^T [\![\, \overline{\mathbf{V}} \,]\!] ds \\ \displaystyle = \sum_{F\in\Gamma^a} \int_F \left(\frac{1}{2}(M_{F,K} - I_{FK} G_{\mathbf{n}_F})\mathbf{W}^{\mathrm{inc}}\right)^T \overline{\mathbf{V}} ds, \quad \forall \mathbf{V} \in V_h \times V_h, \end{array}\right. \tag{5}$$

where Γ^0, Γ^a and Γ^m respectively denote the set of interior (triangular) faces, the set of faces on Γ_a and the set of faces on Γ_m. The unitary normal associated with the oriented face F is $\mathbf{n}_F$ and I_{FK} stands for the incidence matrix between oriented faces and elements whose entries are equal to 0 if the face F does not belong to element K, 1 if $F \in K$ and their orientations match, and –1 if $F \in K$ and their orientations do not match. For $F = \partial K \cap \partial \tilde{K}$, we also define $[\![\, \mathbf{V} \,]\!] = I_{FK}\mathbf{V}_{|K} + I_{F\tilde{K}}\mathbf{V}_{|\tilde{K}}$ and $\{\mathbf{V}\} = \frac{1}{2}\left(\mathbf{V}_{|K} + \mathbf{V}_{|\tilde{K}}\right)$. Finally, the matrix S_F, which is hermitian

positive definite, permits us to penalize the jump of a field or of some components of this field on the face F, and the matrix $M_{F,K}$, to be defined later, insures the asymptotic consistency with the boundary conditions of the continuous problem. Problem (5) is often interpreted in terms of local problems in each element K of Ω_h coupled by the introduction of an element boundary term called the numerical flux (see also [10]). In this study, we consider two classical numerical fluxes, which lead to distinct definitions for matrices S_F and $M_{F,K}$:

– **a centered flux** (see [11] for the time-domain equivalent):

$$S_F = 0 \text{ and } M_{F,K} = \begin{cases} I_{FK} \begin{pmatrix} 0_3 & N_{\mathbf{n}_F} \\ -N_{\mathbf{n}_F}^T & 0_3 \end{pmatrix} \text{ if } F \in \Gamma^m, \\ |G_{\mathbf{n}_F}| \text{ if } F \in \Gamma^a. \end{cases} \tag{6}$$

– **an upwind flux** (see [10, 15]):

$$S_F = \frac{1}{2} \begin{pmatrix} N_{\mathbf{n}_F} N_{\mathbf{n}_F}^T & 0_3 \\ 0_3 & N_{\mathbf{n}_F}^T N_{\mathbf{n}_F} \end{pmatrix},$$
$$M_{F,K} = \begin{cases} \begin{pmatrix} \frac{1}{2} N_{\mathbf{n}_F} N_{\mathbf{n}_F}^T & I_{FK} N_{\mathbf{n}_F} \\ -I_{FK} N_{\mathbf{n}_F}^T & 0_3 \end{pmatrix} \text{ if } \in \Gamma^m, \\ |G_{\mathbf{n}_F}| \text{ if } F \in \Gamma^a. \end{cases} \tag{7}$$

Remark 1. The formulation of the DG scheme above (in particular, the centered and upwind fluxes) actually applies to homogeneous materials. For describing the flux in the inhomogeneous case, let us define $Z^K = \sqrt{\frac{\mu^K}{\varepsilon^K}} = \frac{1}{Y^K}$, $Z^F = \frac{Z^K + Z^{\tilde{K}}}{2}$ and $Y^F = \frac{Y^K + Y^{\tilde{K}}}{2}$ where $F = \overline{K} \cap \overline{\tilde{K}}$. With these definitions, the DG scheme in the inhomogeneous case can be written formally as (5) but by modifying S_F also as:

$$S_F = \frac{1}{2} \begin{pmatrix} \frac{1}{Z^F} N_{\mathbf{n}_F} N_{\mathbf{n}_F}^T & 0_3 \\ 0_3 & \frac{1}{Y^F} N_{\mathbf{n}_F}^T N_{\mathbf{n}_F} \end{pmatrix}, \tag{8}$$

and by using for the average a weighted average $\{\cdot\}_F$ for each face F:

$$\{\mathbf{V}\}_F = \frac{1}{2} \left(\begin{pmatrix} \frac{Z^{\tilde{K}}}{Z^F} & 0_3 \\ 0_3 & \frac{Y^{\tilde{K}}}{Y^F} \end{pmatrix} \mathbf{V}_{|K} + \begin{pmatrix} \frac{Z^K}{Z^F} & 0_3 \\ 0_3 & \frac{Y^K}{Y^F} \end{pmatrix} \mathbf{V}_{|\tilde{K}} \right). \tag{9}$$

4.2 Discretization of the DD Algorithm

DG Formulation of the Multi-Domain Problem

Let Γ^{ij} denote the set of faces which belongs to $\Gamma_{ij} = \partial\Omega_i \cap \overline{\Omega}_j$. According to algorithm (4), the interface condition on Γ_{ij} is given by:

$$\mathcal{B}_{\mathbf{n}_{ij}}(\mathbf{W}^{i,n+1} - \mathbf{W}^{j,n}) = 0 \quad \text{for all } F \text{ belonging to } \Gamma^{ij}, \tag{10}$$

which is taken into account in a weak sense in the context of the DG formulation described in Sect. 4.1. Then the DG discretization of a local problem of algorithm (4) can be written using (5) as:

$$\begin{cases} \text{Find } \mathbf{W}_h^{i,n+1} \text{ in } V_h^i \times V_h^i \text{ such that:} \\ a^i(\mathbf{W}_h^{i,n+1}, \mathbf{V}) + b^i(\mathbf{W}_h^{i,n+1}, \mathbf{V}) = f_h^i, \quad \forall \mathbf{V} \in V_h^i \times V_h^i, \end{cases} \tag{11}$$

with:

$$a^i(\mathbf{W}_h^{i,n+1}, \mathbf{V}) = \int_{\Omega_h^i} \left(\mathrm{i}\omega G_0 \mathbf{W}_h^{i,n+1}\right)^T \overline{\mathbf{V}} dv + \sum_{K \in \Omega_h^i} \int_K \left(\sum_{l \in \{x,y,z\}} G_l \partial_l (\mathbf{W}_h^{i,n+1}) \right)^T \overline{\mathbf{V}} dv,$$

$$b^i(\mathbf{W}_h^{i,n+1}, \mathbf{V}) = \sum_{F \in \Gamma^{m,i}} \int_F \left(\frac{1}{2}(M_{F,K} - I_{FK} G_{\mathbf{n}_F}) \mathbf{W}_h^{i,n+1} \right)^T \overline{\mathbf{V}} ds + \sum_{F \in \Gamma^{a,i}} \int_F \left(I_{FK} G^-_{\mathbf{n}_F} \mathbf{W}_h^{i,n+1} \right)^T \overline{\mathbf{V}} ds + \sum_{F \in \Gamma^{ij}} \int_F \left(I_{FK} \mathcal{B}_{\mathbf{n}_F} \mathbf{W}_h^{i,n+1} \right)^T \overline{\mathbf{V}} ds + \sum_{F \in \Gamma^{0,i}} \int_F \left[\left(S_F [\![\mathbf{W}_h^{i,n+1}]\!] \right)^T [\![\overline{\mathbf{V}}]\!] - \left(G_{\mathbf{n}_F} [\![\mathbf{W}_h^{i,n+1}]\!] \right)^T \{\overline{\mathbf{V}}\} \right] ds,$$

$$f_h^i = \sum_{F \in \Gamma^{a,i}} \int_F \left(I_{FK} G^-_{\mathbf{n}_F} \mathbf{W}^{\mathrm{inc}} \right)^T \overline{\mathbf{V}} ds + \sum_{F \in \Gamma^{ij}} \int_F \left(I_{FK} \mathcal{B}_{\mathbf{n}_F} \mathbf{W}_h^{j,n} \right)^T \overline{\mathbf{V}} ds.$$

We note that the proposed numerical treatment of the interface condition (10) (see the boundary integral terms on Γ^{ij} in the expressions for b^i and f_h^i) is only valid for the classical interface condition or for a zero-order optimized interface condition such as the one selected in this study.

Formulation of an Interface System

In the two-domain case the Schwarz algorithm can be written formally as:

$$\begin{cases} \mathcal{L}\mathbf{W}^{1,n+1} = \mathbf{f}^1, \text{ in } \Omega_1, \\ \mathcal{B}_{\mathbf{n}_{12}} \mathbf{W}^{1,n+1} = \lambda^{1,n}, \text{ on } \Gamma_{12}, \\ + \text{B.C. on } \partial\Omega_1 \cap \partial\Omega, \end{cases} \quad \begin{cases} \mathcal{L}\mathbf{W}^{2,n+1} = \mathbf{f}^2, \text{ in } \Omega_2, \\ \mathcal{B}_{\mathbf{n}_{21}} \mathbf{W}^{2,n+1} = \lambda^{2,n}, \text{ on } \Gamma_{21}, \\ + \text{B.C. on } \partial\Omega_2 \cap \partial\Omega, \end{cases} \tag{12}$$

and then:

$$\lambda^{1,n+1} = \mathcal{B}_{\mathbf{n}_{12}} \mathbf{W}^{2,n+1} \text{ on } \Gamma_{12}, \quad \lambda^{2,n+1} = \mathcal{B}_{\mathbf{n}_{21}} \mathbf{W}^{1,n+1} \text{ on } \Gamma_{21}, \tag{13}$$

where $\mathcal{L}$ is a linear differential operator and $\mathbf{f}^{1,2}$ denotes the right-hand sides associated with $\Omega_{1,2}$. The Schwarz algorithm (12) and (13) can be rewritten in substructured form as:

$$\lambda^{1,n+1} = \mathcal{B}_{\mathbf{n}_{12}} \mathbf{W}^2(\lambda^{2,n}, \mathbf{f}^2) \quad , \quad \lambda^{2,n+1} = \mathcal{B}_{\mathbf{n}_{21}} \mathbf{W}^1(\lambda^{1,n}, \mathbf{f}^1),$$

where $\mathbf{W}^j = \mathbf{W}^j(\lambda^j, \mathbf{f}^j)$ are the solutions of the local problems. By linearity of the operators involved, an iteration of the Schwarz algorithm is then $\boldsymbol{\lambda}^{n+1} = (\mathrm{Id} - \mathcal{T})\boldsymbol{\lambda}^n + \mathbf{d}$, which is a fixed point iteration to solve the interface system $\mathcal{T}\boldsymbol{\lambda} = \mathbf{d}$, where $\boldsymbol{\lambda} = (\boldsymbol{\lambda}^1, \boldsymbol{\lambda}^2)$. From the discrete point of view, the global problem on domain Ω can be written in the matrix form:

$$\begin{pmatrix} A_1 & 0 & R_{12} & 0 \\ 0 & A_2 & 0 & R_{21} \\ 0 & -B_{21} & \mathrm{I} & 0 \\ -B_{12} & 0 & 0 & \mathrm{I} \end{pmatrix} \begin{pmatrix} \mathbf{W}_h^1 \\ \mathbf{W}_h^2 \\ \boldsymbol{\lambda}_h^1 \\ \boldsymbol{\lambda}_h^2 \end{pmatrix} = \begin{pmatrix} \mathbf{f}_h^1 \\ \mathbf{f}_h^2 \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix},$$

where $A_{1,2}$ are local matrices coupling only internal unknowns, $R_{12,21}$ express the coupling between internal unknowns and interface unknowns, and the subscript h denotes the discrete counterpart of a given quantity (e.g. $\boldsymbol{\lambda}_h^{1,2}$ are the discretized unknown vectors corresponding to $\boldsymbol{\lambda}^{1,2}$). The elimination of the internal unknowns $\mathbf{W}_h^{1,2}$ leads to the discrete interface problem $\mathcal{T}_h \boldsymbol{\lambda}_h = \mathbf{g}_h$ with:

$$\mathcal{T}_h = \begin{pmatrix} \mathrm{I} & B_{21}A_2^{-1}R_{21} \\ B_{12}A_1^{-1}R_{12} & \mathrm{I} \end{pmatrix} \quad \text{and} \quad \mathbf{g}_h = \begin{pmatrix} B_{21}A_2^{-1}\mathbf{f}_h^2 \\ B_{12}A_1^{-1}\mathbf{f}_h^1 \end{pmatrix},$$

where $\mathcal{T}_h$ and $\mathbf{g}_h$ are the discretization of $\mathcal{T}$ and $\mathbf{d}$. This system is then solved by a Krylov subspace method, as discussed in the following section.

5 Numerical Results

5.1 The 2D Case

We first present results for the solution of the 2D transverse magnetic Maxwell equations in the case of a heterogeneous non-conducting media:

$$\begin{cases} \mathrm{i}\omega\mu_r H_x + \dfrac{\partial E_z}{\partial y} = 0, \\ \mathrm{i}\omega\mu_r H_y - \dfrac{\partial E_z}{\partial x} = 0, \\ \mathrm{i}\omega\varepsilon_r E_z - \dfrac{\partial H_y}{\partial x} + \dfrac{\partial H_x}{\partial y} = 0. \end{cases}$$

The considered test problem is the scattering of a plane wave (F=300 MHz) by a dielectric cylinder. For that purpose, we make use of a non-uniform triangular mesh which consists of 2,078 vertices and 3,958 triangles (see Fig. 1 left). The relative permittivity of the inner cylinder is set to 2.25 while vacuum is assumed for the rest of the domain. We compare the solutions obtained using a DGTH-$\mathbb{P}_p$ method with $p = 1, 2, 3, 4$ (i.e. the approximation order p is the same for all the mesh elements) and a

variable order DGTH-$\mathbb{P}_{p_K}$ method (i.e. p_K is the approximation order in element K). In the latter case, the approximation order is defined empirically at the element level based on the triangle area resulting in a distribution for which the number of elements with $p_K = 1, 2, 3, 4$ is respectively equal to 1,495, 2,037, 243 and 183 (contour lines of E_z are shown on Fig. 1 right). The interface system is solved using the BiCGStab method. The convergence of the iterative solution of the interface system is evaluated in terms of the Euclidean norm of the residual normalized to the norm of the right-hand side vector. The corresponding linear threshold has been set to $\varepsilon_i = 10^{-6}$. The subdomain problems are solved using the MUMPS optimized sparse direct solver [2].

Numerical simulations have been conducted on a cluster of 20 Intel Xeon/2.33 GHz based nodes interconnected by a high performance Myrinet network. Each node consists of a dual processor quad core board with 16 GB of shared memory. Performance results are summarized in Table 1 where N_s denotes the number of subdomains and "# iter" is the number of iterations of the BiCGStab method. Moreover, this table also includes the values of the L^2 error on the E_z component for the approximate solutions resulting from each algorithm. We stress that the error is not reduced for increasing approximation order because, in the current implementation of the DG method, we make use of an affine transformation between the reference and the physical elements of the mesh. These results demonstrate that the simple optimized interface condition considered here (see Sect. 3) results in substantial reductions of the required number of BiCGStab iterations for convergence of the Schwarz algorithm. Worthwhile to note, the performance improvement increases with the approximation order in the DG method. Considering the case of the DGTH-$\mathbb{P}_{p_K}$ method and for the decomposition into $N_s = 4$ subdomains, the elapsed time of the simulation is equal to 25.8 and 3.6 s for the classical and optimized Schwarz algorithms respectively.

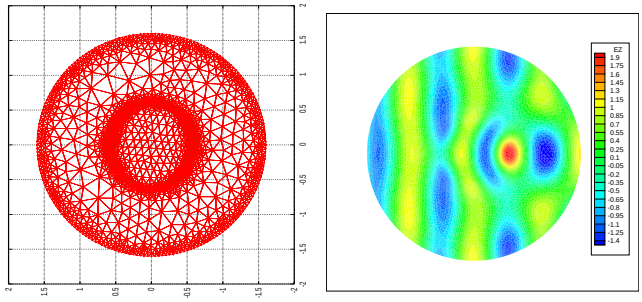


Fig. 1. Scattering of a plane wave by a dielectric cylinder. Unstructured triangular mesh (*left*) and contour lines of E_z (*right*).

Table 1. Scattering of a plane wave by a dielectric cylinder. Classical v.s. optimized Schwarz method. DGTH-$\mathbb{P}_p$ method based on the upwind flux (figures between brackets are the gains in the number of BiCGStab iterations between the classical and optimized Schwarz algorithms).

Method	L^2 error on E_z classical	L^2 error on E_z optimized	N_s	# iter classical	# iter optimized
DGTH-$\mathbb{P}_1$	0.16400	0.16457	4	317	52 (6.1)
–	0.16400	0.16467	16	393	83 (4.7)
DGTH-$\mathbb{P}_2$	0.05701	0.05705	4	650	61 (10.7)
–	0.05701	0.05706	16	734	109 (6.7)
DGTH-$\mathbb{P}_3$	0.05519	0.05519	4	1,067	71 (15.0)
–	0.05519	0.05519	16	1,143	139 (8.2)
DGTH-$\mathbb{P}_4$	0.05428	0.05427	4	1,619	83 (19.5)
–	0.05427	0.05527	16	1,753	170 (10.3)
DGTH-$\mathbb{P}_{p_K}$	0.05487	0.05486	4	352	49 (7.2)
–	0.05487	0.05491	16	414	81 (5.1)

5.2 The 3D Case

We now consider a more realistic 3D problem, namely the simulation of the exposure of a geometrical model of head tissues to a plane wave (F=1,800 MHz). Starting from MR images of the Visible Human project [16], head tissues are segmented and the interfaces of a selected number of tissues (namely, the skin, the skull and the brain) are triangulated (see Fig. 2 left). Then, these triangulated surfaces are used as inputs for the generation of volume meshes. We consider here heterogeneous geometrical models involving four tissues: the skin ($\varepsilon_r = 43.85$ and $\sigma = 1.23$ S/m), the skull ($\varepsilon_r = 15.56$ and $\sigma = 0.43$ S/m), the CSF (Cerebro Spinal Fluid) ($\varepsilon_r = 67.20$ and $\sigma = 2.92$ S/m) and the brain ($\varepsilon_r = 43.55$ and $\sigma = 1.15$ S/m). Note that the exterior of the head must also be meshed, up to a certain distance from the skin, the overall domain being artificially bounded by a sphere on which an absorbing condition is imposed. Two tetrahedral meshes have been used: the first one (referred to as M1) consists of $188,101$ vertices and $1,118,952$ tetrahedra, while the second mesh (referred to as M2) consists of $309,599$ vertices and $1,853,832$ tetrahedra. Contour lines of E_x are shown on Fig. 2 right.

Numerical simulations have been conducted on a Bull Novascale 3045 parallel system consisting of Intel Itanium 2/1.6 GHz nodes interconnected by a high performance Infiniband network. Each node consists of a 8 core board with 21 GB of shared memory. We present performance results for the classical Schwarz algorithm only and the DGTH-$\mathbb{P}_1$ discretization method. The interface system is solved using the BiCGstab(ℓ) [17] method with a linear threshold that has been set to $\varepsilon_i = 10^{-6}$. The subdomain problems are solved using the MUMPS optimized sparse direct solver [2] but this time, the L and U factors are computed in single precision arithmetic in order to reduce the memory requirements for storing the L and U factors associated with the subdomain problems, and an iterative refinement strategy is used to increase the accuracy of the subdomain triangular solves. Performance results are summa-

rized in Table 2 for the factorization and solution phases. In these tables, "RAM LU (min/max)" denotes the minimum and maximum values of the per-process memory requirement for computing and storing the L and U factors. We note that doubling the number of subdomains results in a slight increase in the number of BiCGstab(ℓ) iterations however, at the same time, the size of the local factors is reduced by a factor well above two and as a consequence, a super-linear speedup is observed in the solution phase.

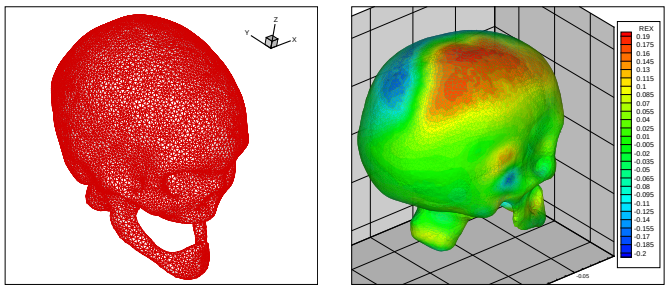


Fig. 2. Propagation of a plane wave in a heterogeneous model of head tissues. DGTH-$\mathbb{P}_1$ method based on a centered flux. Triangulated surface of the skull (*left*) and contour lines of E_x (*right*).

Table 2. Propagation of a plane wave in a heterogeneous model of head tissues. Classical Schwarz method. Performance results of the factorization and solution phases (figures between brackets are relative parallel speedup values).

Mesh	# d.o.f	N_s	RAM LU (min/max)	Elapsed time LU	# iter	Elapsed time
M1	26,854,848	160	2.1 GB/3.1 GB	496 s	30	1,314 s
–	–	320	0.8 GB/1.2 GB	132 s (3.8)	36	528 s (2.5)
M2	44,491,968	256	2.2 GB/3.2 GB	528 s	42	1,824 s
–	–	512	0.8 GB/1.3 GB	142 s (3.7)	49	785 s (2.3)

6 Ongoing and Future Work

We have presented here some results of an ongoing collaborative effort aiming at the design of domain decomposition methods for the solution of the time-harmonic Maxwell equations modeling electromagnetic wave propagation problems in heterogeneous media and complex domains. The discretization in space of the underlying PDE model relies on a high order DG method formulated on unstructured simplicial meshes. For the solution of the resulting complex coefficients, sparse algebraic

systems of equations, we consider using Schwarz algorithms in conjunction with the adopted discretization method. Future work involves the study of optimized Schwarz algorithms based on high order interface conditions for conductive media, and the design of preconditioned iterative strategies for the solution of subdomain problems.

Acknowledgement. This work was granted access to the HPC resources of CCRT under the allocation 2009-t2009065004 made by GENCI (Grand Equipement National de Calcul Intensif).

References

1. A. Alonso-Rodriguez and L. Gerardo-Giorda. New nonoverlapping domain decomposition methods for the harmonic Maxwell system. *SIAM J. Sci. Comput.*, 28(1):102–122, 2006.
2. P.R. Amestoy, I.S. Duff, and J.-Y. L'Excellent. Multifrontal parallel distributed symmetric and unsymmetric solvers. *Comput. Methods App. Mech. Engng.*, 184:501–520, 2000.
3. P. Chevalier and F. Nataf. An OO2 (Optimized Order 2) method for the Helmholtz and Maxwell equations. In *10th International Conference on Domain Decomposition Methods in Science and in Engineering*, pp. 400–407. AMS Boulder, CO, 1997.
4. B. Després. Décomposition de domaine et problème de Helmholtz. *C.R. Acad. Sci. Paris*, 1(6):313–316, 1990.
5. B. Després, P. Joly, and J.E. Roberts. A domain decomposition method for the harmonic Maxwell equations. In *Iterative Methods in Linear Algebra*, pp. 475–484, North-Holland, Amsterdam, 1992.
6. V. Dolean, H. Fol, S. Lanteri, and R. Perrussel. Solution of the time-harmonic Maxwell equations using discontinuous Galerkin methods. *J. Comput. Appl. Math.*, 218(2):435–445, 2008.
7. V. Dolean, L. Gerardo-Giorda, and M. Gander. Optimized Schwarz methods for Maxwell equations. *SIAM J. Sci. Comput.*, 31(3):2193–2213, 2009.
8. V. Dolean, S. Lanteri, and R. Perrussel. A domain decomposition method for solving the three-dimensional time-harmonic Maxwell equations discretized by discontinuous Galerkin methods. *J. Comput. Phys.*, 227(3):2044–2072, 2008.
9. V. Dolean, S. Lanteri, and R. Perrussel. Optimized Schwarz algorithms for solving time-harmonic Maxwell's equations discretized by a discontinuous Galerkin method. *IEEE. Trans. Magn.*, 44(6):954–957, 2008.
10. A. Ern and J.-L. Guermond. Discontinuous Galerkin methods for Friedrichs systems I. General theory. *SIAM J. Numer. Anal.*, 44(2):753–778, 2006.
11. L. Fezoui, S. Lanteri, S. Lohrengel, and S. Piperno. Convergence and stability of a discontinuous Galerkin time-domain method for the 3D heterogeneous Maxwell equations on unstructured meshes. *ESAIM: Math. Model. Numer. Anal.*, 39(6):1149–1176, 2005.
12. M. Gander, F. Magoulès, and F. Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24(1):38–60, 2002.
13. S.C. Lee, M. Vouvakis, and J.F. Lee. A non-overlaping domain decomposition method with non-matching grids for modeling large finite antenna arrays. *J. Comput. Phys.*, 203: 1–21, 2005.

14. P. Monk. *Finite Element Methods for Maxwell's Equations*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, NY, 2003. ISBN 0-19-850888-3.
15. S. Piperno. L^2-stability of the upwind first order finite volume scheme for the Maxwell equations in two and three dimensions on arbitrary unstructured meshes. *M2AN: Math. Model. Numer. Anal.*, 34(1):139–158, 2000.
16. P. Ratiu, B. Hillen, J. Glaser, and D.P. Jenkins. *Medicine Meets Virtual Reality 11 - NextMed: Health Horizon*, volume 11, chapter Visible Human 2.0 – the next generation, pp. 275–281. IOS Press, Fairfax, VA 2003.
17. G.L.G. Sleijpen and D.R. Fokkema. BiCGstab(ℓ) for linear equations involving unsymmetric matrices with complex spectrum. *Electron. Trans. Numer. Anal.*, 1:11–32 (electronic only), 1993.
18. M. Vouvakis, Z. Cendes, and J.F. Lee. A FEM domain decomposition method for photonic and electromagnetic band gap structures. *IEEE Trans. Ant. Prop.*, 54(2):721–733, 2006.

N–N Solvers for a DG Discretization for Geometrically Nonconforming Substructures and Discontinuous Coefficients

Maksymilian Dryja[1], Juan Galvis[2], and Marcus Sarkis[3,4]

[1] Department of Mathematics, Warsaw University, Warsaw 02-097, Poland. This work was supported in part by The Polish Sciences Foundation under grant NN201006933.
[2] Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA
[3] Instituto Nacional de Matemática Pura e Aplicada, Rio de Janeiro 22460-320, Brazil
[4] Department of Mathematical Sciences, Worcester Polytechnic Institute, Worcester, MA 01609, USA

1 Summary

A discontinuous Galerkin discretization for second order elliptic equations with *discontinuous coefficients* in 2-D is considered. The domain of interest Ω is assumed to be a union of polygonal substructures Ω_i of size $O(H_i)$. We allow this substructure decomposition to be geometrically nonconforming. Inside each substructure Ω_i, a conforming finite element space associated to a triangulation $\mathcal{T}_{h_i}(\Omega_i)$ is introduced. To handle the nonmatching meshes across $\partial\Omega_i$, a discontinuous Galerkin discretization is considered. In this paper additive Neumann–Neumann Schwarz methods are designed and analyzed. Under natural assumptions on the coefficients and on the mesh sizes across $\partial\Omega_i$, a condition number estimate $C(1+\max_i \log \frac{H_i}{h_i})^2$ is established with C independent of h_i, H_i, h_i/h_j, and the jumps of the coefficients. The method is well suited for parallel computations and can be straightforwardly extended to three dimensional problems. Numerical results, which are not included in this paper, confirm the theoretical results.

2 Introduction

In this paper a *discontinuous Galerkin* (DG) approximation of elliptic problems with *discontinuous coefficients* is considered [3]. See [1, 9] and references therein for an overview on local DG discretizations. The problem is considered in a two-dimensional polygonal region Ω which is a *geometrically nonconforming* union of disjoint polygonal substructures Ω_i, $i = 1, \ldots, N$. For simplicity of presentation we

assume that inside each substructure Ω_i the coefficient ρ_i is constant. The extension of the results to mildly variation of ρ_i inside Ω_i is straightforward. Large discontinuities of the coefficients are assumed to occur only across the interfaces of the substructures $\partial\Omega_i$. Inside each substructure Ω_i a conforming finite element method is introduced to discretize the local problem, and *nonmatching triangulations* are allowed to occur across the $\partial\Omega_i$. This kind of composite discretization is motivated by the location of the discontinuities of the coefficients and by the regularity of the solution of the problem. The discrete problem is formulated using a symmetric DG method with interior penalty (IPDG) terms on $\partial\Omega_i$. To deal with the discontinuities of the coefficients across the substructure interfaces, *harmonic averages* of the coefficients are considered on these interfaces; see [3].

The main goal of this paper is to design and analyze additive Neumann–Neumann algorithms for the resulting DG-discrete problem. This type of algorithms is well established for standard conforming and nonconforming discretizations; see [10] and references therein. We note that other two-level and multilevel preconditioners have been considered for solving discrete IPDG problems; see [2, 6, 8] and references therein. These papers focus on the scalability of the preconditioners with respect to the mesh parameters, however, little has been said about the robustness with respect to jumps of the coefficients and nonmatching grids across the substructuring interfaces. The notion of discrete harmonic extension in the DG sense was also introduced in [4] to achieve these desirable robustness for geometrically conforming substructures. In this paper we consider both the *geometrically nonconforming case* and *discontinuous coefficients*.

The problem is reduced to the Schur complement form with respect to unknowns on $\partial\Omega_i$, for $i = 1, \ldots, N$. Discrete harmonic functions defined in a special way, see Sect. 3.3, are used in this step. The methods are designed and analyzed for the Schur complement problem using the general theory of N–N methods; see [10]. The local problems are defined on Ω_i and edges or part of the edge of $\partial\Omega_j$ which are common to Ω_i. The coarse space is defined by using a special partitioning of unity with respect to the subdomains Ω_i and by introducing master and slave sides of the local interfaces between the substructures. Recall that we work with a geometrically nonconforming partition of Ω into substructures Ω_i, $i = 1, \ldots, N$. A (part of the) edge $E_{ij} = \partial\Omega_i \cap \partial\Omega_j$ is a master side when $\rho_i \geq C\rho_j$, otherwise it is a slave side. Hence, if $E_{ij} \subset \partial\Omega_i$ is a master side then $E_{ji} \subset \partial\Omega_j$, $E_{ij} = E_{ji}$, is a slave. The h_i-triangulation on E_{ij} and h_j-triangulation on E_{ji} are built in such a way that $h_i \geq Ch_j$ if $\rho_i \geq C\rho_j$. Here h_i and h_j are the parameters of the triangulation in Ω_i and Ω_j, respectively, and C is a generic $O(1)$ constant. We prove that the algorithms are almost optimal and their rates of convergence are independent of the mesh parameters, the number of subdomains Ω_i and the jumps of the coefficients. The algorithms are well suited for parallel computations and they can be straightforwardly extended to three-dimensional problems.

The paper is organized as follows. In Sects. 3.1 and 3.2 the differential problem and its DG discretization are formulated. In Sect. 3.3 the Schur complement problem is derived using discrete harmonic functions in a special way. Section 4 is dedicated to introducing notation and the *interface condition* on the coefficients and the mesh parameters. Two additive Neumann–Neumann Schwarz preconditioners, one based on a small coarse space and the other based on a larger coarse space, are defined and analyzed in Sect. 5.

3 Differential and Discrete Problems

In this section we formulate the discrete problem and its Schur complement problem.

3.1 Differential Problem

Consider the following problem: Find $u^* \in H_0^1(\Omega)$ such that

$$a(u^*, v) = f(v) \qquad \text{for all } v \in H_0^1(\Omega) \tag{1}$$

where

$$a(u,v) := \sum_{i=1}^{N} \int_{\Omega_i} \rho_i \nabla u \cdot \nabla v dx \ \text{ and } \ f(v) := \int_{\Omega} f v dx.$$

Here, $\overline{\Omega} = \cup_{i=1}^{N} \overline{\Omega}_i$ where the substructures Ω_i are disjoint regular polygonal subregions of diameter $O(H_i)$. We assume that the substructures Ω_i form a geometrically nonconforming partition of Ω, therefore, for all $i \neq j$ the intersection $\partial\Omega_i \cap \partial\Omega_j$ is empty, a vertex of Ω_i and/or Ω_j, or a common edge or part of an edge of $\partial\Omega_i$ and $\partial\Omega_j$. If the decomposition is geometrically conforming, then the intersection $\partial\Omega_i \cap \partial\Omega_j$ is empty or a common vertex of Ω_i and Ω_j, or a common edge of Ω_i and Ω_j. For simplicity of presentation we assume that the right-hand side $f \in L^2(\Omega)$ and the coefficients ρ_i are all positive constants.

3.2 Discrete Problem

In each Ω_i presentation, we introduce a shape regular triangulation $\mathcal{T}_{h_i}(\Omega_i)$ with triangular elements and the mesh parameter h_i. The resulting triangulation of Ω is in general nonmatching across $\partial\Omega_i$. We let $X_i(\Omega_i)$ be the regular finite element (FE) space of piecewise linear and continuous functions in $\mathcal{T}_{h_i}(\Omega_i)$. We do not assume that the functions in $X_i(\Omega_i)$ vanish on $\partial\Omega_i \cap \partial\Omega$. We define

$$X_h(\Omega) := X_1(\Omega_1) \times \cdots \times X_N(\Omega_N)$$

and represent functions v of $X_h(\Omega)$ as $v = \{v_i\}_{i=1}^{N}$ with $v_i \in X_i(\Omega_i)$.

The discrete problem obtained by the DG method, see [1, 3, 9], is of the form: Find $u_h^* \in X_h(\Omega)$ such that

$$\hat{a}_h(u_h^*, v_h) = f(v_h) \quad \text{for all } v_h \in X_h(\Omega) \tag{2}$$

where

$$\hat{a}_h(u,v) = \sum_{i=1}^{N} \hat{a}_i(u,v) \quad \text{and} \quad f(v) = \sum_{i=1}^{N} \int_{\Omega_i} f v_i dx. \tag{3}$$

Each bilinear form $\hat{a}_i$ is given as a sum of three bilinear forms:

$$\hat{a}_i(u,v) := a_i(u,v) + s_i(u,v) + p_i(u,v), \tag{4}$$

where

$$a_i(u,v) := \int_{\Omega_i} \rho_i \nabla u_i \cdot \nabla v_i dx, \tag{5}$$

$$s_i(u,v) := \sum_{E_{ij} \subset \partial\Omega_i} \int_{E_{ij}} \frac{\rho_{ij}}{l_{ij}} \left(\frac{\partial u_i}{\partial n_i}(v_j - v_i) + \frac{\partial v_i}{\partial n_i}(u_j - u_i) \right) ds$$

and

$$p_i(u,v) := \sum_{E_{ij} \subset \partial\Omega_i} \int_{E_{ij}} \frac{\rho_{ij}}{l_{ij}} \frac{\delta}{h_{ij}} (u_j - u_i)(v_j - v_i) ds. \tag{6}$$

Here, the bilinear form p_i is called the penalty term with a positive penalty parameter δ. In the above equations, we set $l_{ij} = 2$ when $E_{ij} = \partial\Omega_i \cap \partial\Omega_j$ is a common edge (or part of an edge) of $\partial\Omega_i$ and $\partial\Omega_j$, and define $\rho_{ij} := 2\rho_i\rho_j/(\rho_i + \rho_j)$ as the harmonic average of ρ_i and ρ_j, and $h_{ij} := 2h_ih_j/(h_i + h_j)$. In order to simplify notation we include the index $j = \partial$ when $E_{i\partial} := \partial\Omega_i \cap \partial\Omega$ is an edge of $\partial\Omega_i$ and set $l_{i\partial} := 1$ and let $v_\partial = 0$ for all $v \in X_h(\Omega)$, and define $\rho_{i\partial} := \rho_i$ and $h_{i\partial} := h_i$. The outward normal derivative on $\partial\Omega_i$ is denoted by $\frac{\partial}{\partial n_i}$. We note that when ρ_{ij} is given by the harmonic average then $\min\{\rho_i, \rho_j\} \leq \rho_{ij} \leq 2\min\{\rho_i, \rho_j\}$.

A priori error estimates for the method are optimal for constant coefficients, and also for the case where h_i and h_j are of the same order; see [1, 9]. For discontinuous coefficients ρ_i and/or for mesh sizes h_i and h_j are not on the same order, see Theorem 4.2 of [3] and Lemma 2.2 of [5].

3.3 Schur Complement Problem

In this subsection we derive the Schur complement bilinear form for the problem (2). We first introduce auxiliary notation.

Define $X_i^\circ(\Omega_i)$ as the subspace of $X_i(\Omega_i)$ of functions that vanish on $\partial\Omega_i$. A function $u_i \in X_i(\Omega)$ can be represented as

$$u_i = \mathcal{H}_i u_i + \mathcal{P}_i u_i \tag{7}$$

where $\mathcal{H}_i u_i$ is the discrete harmonic part of u_i in the sense of $a_i(.,.)$, see (5), i.e.,

$$\begin{cases} a_i(\mathcal{H}_i u_i, v_i) = 0 \quad \text{for all } v_i \in X_i^\circ(\Omega_i) \\ \mathcal{H}_i u_i = u_i \quad \text{on} \quad \partial\Omega_i, \end{cases} \tag{8}$$

while $\mathcal{P}_i u_i$ is the projection of u_i into $X_i^\circ(\Omega_i)$ in the sense of $a_i(.,.)$, i.e.,

$$a_i(\mathcal{P}_i u_i, v_i) = a_i(u_i, v_i) \quad \text{for all } v_i \in X_i^\circ(\Omega_i). \tag{9}$$

Note that $\mathcal{H}_i u_i$ is the classical discrete harmonic part of u_i. Let us denote by $X_h^\circ(\Omega)$ the subspace of $X_h(\Omega)$ defined by $X_h^\circ(\Omega) := X_1^\circ(\Omega_1) \times \cdots \times X_N^\circ(\Omega_N)$ and consider the global projections $\mathcal{H}u := \{\mathcal{H}_i u_i\}_{i=1}^N$ and $\mathcal{P}u := \{\mathcal{P}_i u_i\}_{i=1}^N : X_h(\Omega) \to X_h^\circ(\Omega)$ in the sense of $\sum_{i=1}^N a_i(\cdot,\cdot)$. Hence, a function $u \in X_h(\Omega)$ can then be decomposed as

$$u = \mathcal{H}u + \mathcal{P}u. \tag{10}$$

Alternatively to (10), a function $u \in X_h(\Omega)$ can be represented as

$$u = \hat{\mathcal{H}}u + \hat{\mathcal{P}}u, \tag{11}$$

where $\hat{\mathcal{P}}u = \{\hat{\mathcal{P}}_i u_i\}_{i=1}^N : X_h(\Omega) \to X_h^\circ(\Omega)$ is the projection in the sense of the original bilinear for $\hat{a}_h(\cdot,\cdot)$, see (3), and $\hat{\mathcal{H}}u = \{\hat{\mathcal{H}}_i u\}_{i=1}^N \in X_h(\Omega)$ where $\hat{\mathcal{H}}_i u$ is the discrete harmonic part of u in the sense of $\hat{a}_i(.,.)$ defined in (4), i.e., $\hat{\mathcal{H}}_i u \in X_i(\Omega_i)$ is the solution of

$$\begin{cases} \hat{a}_i(\hat{\mathcal{H}}_i u, v_i) = 0 \quad \text{for all } v_i \in X_i^\circ(\Omega_i), \\ \qquad \hat{\mathcal{H}}_i u = u_i \ \text{ on } \partial\Omega_i \\ \qquad \hat{\mathcal{H}}_i u = u_j \ \text{ on every (part of) edge } E_{ji} \subset \partial\Omega_j. \end{cases} \tag{12}$$

Here the index j in the last equation of (12) runs over all Ω_j and $j = \partial$ such that $\overline{\Omega}_i \cap \overline{\Omega}_j$ and $\overline{\Omega}_i \cap \partial\Omega$ has one-dimensional nonzero measure, respectively. In the latter case, recall that $u_\partial = 0$.

Observe that since $\hat{\mathcal{P}}_i u_i \in X_i^\circ(\Omega_i)$ we have that for all $v_i \in X_i^\circ(\Omega_i)$,

$$a_i(\hat{\mathcal{P}}_i u, v_i) = \hat{a}_h(u, R_i^T v_i),$$

where R_i^T is the standard discrete zero extension operator, i.e., $R_i^T v_i := \{v_j\}_{j=1}^N$, where v_j vanishes for $j \neq i$; see also Sect. 4 for the definition of other discrete zero extension operators I_i and $\tilde{I}_i$.

The discrete solution of (2) can be decomposed as $u_h^* = \hat{\mathcal{H}}u_h^* + \hat{\mathcal{P}}u_h^*$. To compute the projection $\hat{\mathcal{P}}u_h^*$ we need to solve the following set of standard discrete Dirichlet problems:

$$a_i(\hat{\mathcal{P}}_i u_h^*, v_i) = f(R_i^T v_i) \qquad \text{for all } v_i \in X_i^\circ(\Omega_i). \tag{13}$$

Note that these problems, for $i = 1, \dots N$, are local and independent, and so, they can be solved in parallel. This is a precomputational step.

We next formulate the problem for $\hat{\mathcal{H}} u_h^*$. We first point out that for $v_i \in X_i^\circ(\Omega_i)$ we have

$$\hat{a}_i(u_i, v_i) = (\rho_i \nabla u_i, \nabla v_i)_{L^2(\Omega_i)} + \sum_{E_{ij} \subset \partial\Omega_i} \frac{\rho_{ij}}{l_{ij}} (\frac{\partial v_i}{\partial n}, u_j - u_i)_{L^2(E_{ij})}. \tag{14}$$

For $u \in X_h(\Omega)$ observe that (12) is obtained from

$$\hat{a}_h(\hat{\mathcal{H}} u, v) = 0 \tag{15}$$

by taking $v = \{v_i\}_{i=1}^N \in X_h^\circ(\Omega)$. It is easy to see that $\hat{\mathcal{H}} u = \{\hat{\mathcal{H}}_i u\}_{i=1}^N$ and $\hat{\mathcal{P}} u = \{\hat{\mathcal{P}}_i u_i\}_{i=1}^N$ are orthogonal in the sense of $\hat{a}_h(.,.)$, i.e.,

$$\hat{a}_h(\hat{\mathcal{H}} u, \hat{\mathcal{P}} v) = 0, \qquad u, v \in X^h(\Omega). \tag{16}$$

In addition,

$$\mathcal{H}\hat{\mathcal{H}} u = \mathcal{H} u \quad \text{and} \quad \hat{\mathcal{H}}\mathcal{H} u = \hat{\mathcal{H}} u \tag{17}$$

since neither $\hat{\mathcal{H}} u$ nor $\mathcal{H} u$ changes the values of u at the nodes on the boundaries of the subdomains Ω_i; see (8) and (12).

Define

$$\Gamma_h := (\cup_i \partial\Omega_{ih_i}), \tag{18}$$

where $\partial\Omega_{ih_i}$ is the set of nodal points of $\partial\Omega_i$. *We note that the definition of Γ_h includes the nodes on both triangulations of $\cup_i \partial\Omega_i$.*

We are now in a position to derive the Schur complement problem for (2). Applying the decomposition (11) in (2) we obtain

$$\hat{a}_h(\hat{\mathcal{H}} u_h^* + \hat{\mathcal{P}} u_h^*, \hat{\mathcal{H}} v_h + \hat{\mathcal{P}} v_h) = f(\hat{\mathcal{H}} v_h + \hat{\mathcal{P}} v_h).$$

Using (13) and (15) we have

$$\hat{a}_h(\hat{\mathcal{H}} u_h^*, \hat{\mathcal{H}} v_h) = f(\hat{\mathcal{H}} v_h) \qquad \text{for all } v_h \in X_h(\Omega). \tag{19}$$

This is the Schur complement problem for (2). We denote by V the set of all functions v_h in $X_h(\Omega)$ such that $v_h \equiv \hat{\mathcal{H}} v_h$, i.e., the space of discrete harmonic functions in the sense of the $\hat{\mathcal{H}}$. We rewrite the Schur complement problem as follows: Find $u_h^* \in V$ such that

$$\mathcal{S}(u_h^*, v_h) = g(v_h) \qquad \text{for all } v_h \in V \tag{20}$$

where, here and below, $u_h^* \equiv \hat{\mathcal{H}} u_h^*$ and

$$\mathcal{S}(u_h, v_h) := \hat{a}_h(\hat{\mathcal{H}} u_h, \hat{\mathcal{H}} v_h) \ \text{ and } \ g(v_h) := f(\hat{\mathcal{H}} v_h). \tag{21}$$

The Schur complement problem (20) has a unique solution.

4 Notation and the Interface Condition

We first classify substructures according to their position with respect to the boundary $\partial\Omega$. We say that a substructure Ω_i is an *interior substructure* or *floating substructures* if Ω_i does not share an edge with the boundary of Ω. Otherwise, we say it is a *boundary substructure* or *nonfloating substructure*. We denote by $\mathcal{N}_I$ and $\mathcal{N}_B$ the sets of indices of interior and boundary substructures, respectively.

Let $\overset{\circ}{\Omega}_{ih_i}$ and $\partial\Omega_{ih_i}$ be the interior and boundary nodes of $\mathcal{T}_{h_i}(\overline{\Omega}_i)$ in Ω_i and on $\partial\Omega_i$, respectively. Define E_{ijh_i} as the set of nodes of $\partial\Omega_{ih_i}$ that are on E_{ij}. Recall that E_{ij} is a *closed* interval. We also define ∂E_{ijh_i} as the set of nodes on E_{ijh_i} that are closest to the boundary ∂E_{ij}. Let $\overset{\circ}{E}_{ijh_i} := E_{ijh_i} \setminus \partial E_{ijh_i}$ be the set of interior nodes in E_{ij}. Additionally, we define the extended boundary nodes $\partial^e E_{ijh_i}$ as the union of ∂E_{ijh_i} and the nodal points $y \in \partial\Omega_i \setminus E_{ij}$ closest to $x \in \partial E_{ij}$ when x is not a nodal point. Note that when E_{ij} is a full edge of $\partial\Omega_i$, then $\partial^e E_{ijh_i} = \partial E_{ij}$. Let $\overline{E}_{ijh_i} := \overset{\circ}{E}_{ijh_i} \cup \partial^e E_{ijh_i}$. We define

$$\Gamma_i := \partial\Omega_{ih_i} \cup \bigcup_{E_{ij}\subset\partial\Omega_i} \overline{E}_{jih_j}. \tag{22}$$

Note that Γ_i is defined to include the nodes on Γ_h necessary for computing $\hat{\mathcal{H}}_i$; see (12). Define W_i as the space of piecewise linear functions or its vector representation defined by the nodal values on Γ_i extended via $\hat{\mathcal{H}}_i$ (defined in (12)) inside Ω_i, i.e.,

$$W_i := \left\{ \text{nodal values of } v \text{ defined on } \overset{\circ}{\Omega}_{ih_i} \cup \Gamma_i : \ v \equiv \hat{\mathcal{H}}_i v \text{ in } \Omega_i \right\}. \tag{23}$$

Observe that a function $u^{(i)} \in W_i$ can be represented as

$$u^{(i)} = \{u_l^{(i)}\}_{l\in\#(i)} \quad \text{where} \quad \#(i) = \{i\} \cup \{j : E_{ij} \subset \partial\Omega_i\}.$$

Here $u_i^{(i)}$ and $u_j^{(i)}$ stand for the nodal values of $u^{(i)}$ on $\overline{\Omega}_i$ and on $\overline{E}_{jih_j}$, respectively. Recall also that sometimes we write $u = \{u_i\}_{i=1}^N \in V$ to refer to a function defined on all of Γ_h with each u_i defined (only) on $\partial\Omega_i$; see Sect. 3.2. We point out that E_{ij} and E_{ji} are geometrically the same even though the mesh on the side E_{ij} comes from the Ω_i triangulation while the mesh on the side E_{ji} corresponds from the Ω_j triangulation. Note also that, according to our conventions, if $i \in \mathcal{N}_B$ and $u^{(i)} \in W_i$ then $u_\partial^{(i)} = 0$ on the fictitious edge $E_{\partial i}$.

Define the extension operator $\tilde{I}_i : W_i \to V$ as follows: Given $u^{(i)} \in W_i$, let $\tilde{I}_i u^{(i)}$ be equal to $u^{(i)}$ at the nodes of Γ_i and $\overset{\circ}{\Omega}_{ih_i}$, equal to zero on $\Gamma_h \setminus \Gamma_i$, and extended by $\hat{\mathcal{H}}\tilde{I}_i u^{(i)}$ elsewhere and denoted also by $\tilde{I}_i$, i.e.,

$$\tilde{I}_i u(x) = \begin{cases} u(x) & \text{if } x \in \Gamma_i \\ 0 & \text{if } x \in \Gamma_h \backslash \Gamma_i \\ \hat{\mathcal{H}} \tilde{I}_i u & \text{elsewhere,} \end{cases} \tag{24}$$

where the last condition on (24) means that $\tilde{I}_i u$ is discrete harmonic in the sense of $\hat{\mathcal{H}}$.

To each pair $\{E_{ij}, E_{ji}\}$ we assign one master and one slave side. If E_{ij} is chosen to be the slave side then E_{ji} must be the master one. Note that since we are working with a geometrically nonconforming decomposition of Ω, a part of an edge can be labeled as master side while other part of the same edge can be marked as the slave side. The choice of slave-master sides is such that the *interface condition*, stated next in Assumption 1, can be satisfied. Under this assumption, Theorems 1 below hold with constants C_1 and C_2 independent of the ρ_i, h_i and H_i. This assumption says basically that the coarser meshes h_i should be chosen where the coefficients ρ_i are larger, and additionally, the master side should be chosen on the side where the coefficient is larger. In terms of accuracy, this condition is satisfied in practice since the solution u^* in general varies less where the coefficient is larger. We note that this condition is similar to the ones adopted in mortar studies for geometrical nonconforming cases; [7].

Assumption 1 (The interface condition) *We say that the coefficients $\{\rho_i\}$ and the local mesh sizes $\{h_i\}$ satisfy the interface condition if there exist constants β_1 and β_2, of order 1, such that for any (part of) edge E_{ij}, one of the following inequalities hold:*

$$\begin{cases} h_i \leq \beta_1 h_j \ \text{ and } \ \rho_i \leq \beta_2 \rho_j & \text{if } E_{ij} \text{ is a slave side, or} \\ h_j \leq \beta_1 h_i \ \text{ and } \ \rho_j \leq \beta_2 \rho_i & \text{if } E_{ij} \text{ is a master side.} \end{cases} \tag{25}$$

We associate to each Ω_i $1, \cdots, N$, a diagonal weighting matrix $D^{(i)} = \{D_l^{(i)}\}_{l \in \#(i)}$ on $\Gamma_i \cup \overset{\circ}{\Omega}_{ih_i}$. Let x be a nodal point of $\Gamma_i \cup \overset{\circ}{\Omega}_{ih_i}$. Then, the diagonal element of $D^{(i)}$ associated to x is defined by:

- On $\overset{\circ}{\Omega}_{ih_i} \cup \partial\Omega_{i,h_i}$ $\quad (l = i)$

$$D_i^{(i)}(x) = \begin{cases} 0 \text{ if } x \in \overset{\circ}{E}_{ijh_i} \text{ and } E_{ij} \text{ is a slave side} \\ 1 \text{ otherwise,} \end{cases} \tag{26}$$

- On $\overline{E}_{jih_j}$ $\quad (l = j)$

$$D_j^{(i)}(x) = \begin{cases} 0 \text{ if } x \in \partial^e E_{jih_j}, \\ 1 \text{ if } x \in \overset{\circ}{E}_{jih_j} \text{ and } E_{ij} \text{ is a master side} \\ 0 \text{ if } x \in \overset{\circ}{E}_{jih_j} \text{ and } E_{ij} \text{ is a slave side,} \end{cases} \tag{27}$$

- On $\overline{E}_{i\partial h_i}$

$$D_i^{(i)}(x) = 1 \text{ for all } x \in \overline{E}_{i\partial h_i}.$$

The prolongation operators $I_i : W_i \to V$, $i = 1, \cdots, N$, are defined as

$$I_i = \tilde{I}_i D^{(i)}. \tag{28}$$

It is easy to see that the image of I_i forms a decomposition of V since

$$\sum_{i=1}^{N} I_i \tilde{I}_i^T u = u, \tag{29}$$

where the $\tilde{I}_i^T$ stand for the restriction of V to W_i.

5 Additive Preconditioners

To design and analyze additive N–N type methods for solving (20) we use the general framework of ASMs; see Theorem 2.7 in [10]. We now consider an additive Schwarz method based on the coarse space $V_{0,I}$, i.e., a coarse space with one degree of freedom per interior substructure and no degrees of freedom for any boundary substructure; see (34). We now introduce the local and coarse problems to define the additive Schwarz method $T_{as,I}$.

5.1 Local Problems

Recall the definition of Γ_i in (22), the space W_i in (23) and the sets of $\mathcal{N}_B$ and $\mathcal{N}_I$ substructures, see Sect. 4. Define

$$\begin{cases} V_i = V_i(\Gamma_i) := \left\{ u^{(i)} \in W_i : \int_{\partial\Omega_i} u_i^{(i)} = 0 \right\}, \text{ if } i \in \mathcal{N}_I \\ V_i = V_i(\Gamma_i) := W_i, \qquad\qquad\qquad\qquad \text{if } i \in \mathcal{N}_B \end{cases} \tag{30}$$

i.e., for interior substructures Ω_i, V_i is the subspace of W_i consisting of functions with zero average value on $\partial\Omega_i$, while for boundary substructures, V_i is the whole space W_i. We recall that for $v^{(i)} \in W_i$ (or V_i) then $v^{(i)} \equiv \hat{\mathcal{H}}_i v^{(i)}$ and $v \in V$ we have $v^{(i)} = \hat{\mathcal{H}}_i v^{(i)}$ and $v = \hat{\mathcal{H}} v$.

For $u^{(i)}, v^{(i)} \in V_i$, $i = 1, \ldots, N$, we define the local bilinear form b_i as

$$b_i(u^{(i)}, v^{(i)}) := \hat{a}_i(u^{(i)}, v^{(i)}), \tag{31}$$

where the bilinear form $\hat{a}_i$ is defined in (4). We define the operators $T_i : V \to V$, $i = 1, \ldots, N$, by defining $\tilde{T}_i : V \to V_i$ as

$$b_i(\tilde{T}_i u, v^{(i)}) = \hat{a}_h(u, I_i v^{(i)}) \text{ for all } v^{(i)} \in V_i, \tag{32}$$

and then set $T_i = I_i \tilde{T}_i$. It is easy to see that these problems are well posed and that the T_i are symmetric with respect to the $\hat{a}_h$-inner product.

5.2 Coarse Problems

Let $e^{(i)} \in W_i$ be the vector with value one at the nodes of Γ_i and on $\overset{\circ}{\Omega}_{ih_i}$. Recall that the prolongation operators $\tilde{I}_i$ and I_i are defined in (24) and (28), respectively. Define $\Theta_i \in V$, for $i = 1, \ldots, N$, as $\Theta_i := \tilde{I}_i \Theta^{(i)}$ where $\Theta^{(i)} = D^{(i)} e^{(i)}$, hence, $\Theta_i = I_i e^{(i)}$. Note from (26) and (27) we have that

$$\sum_{i=1}^{N} \Theta_i = 1 \text{ on } \Gamma_h. \tag{33}$$

We consider the following coarse space:

$$V_{0,I} = \text{Span}\,\{\Theta_i\}_{i \in \mathcal{N}_I} \subset V. \tag{34}$$

The coarse bilinear form is defined according to

$$b_0(u, v) = \left(1 + \log \frac{H}{h}\right)^{-2} \hat{a}_h(u, v), \quad u, v \in V_{0,I}. \tag{35}$$

Next we define the projection-like operator $T_0 : V \to V_{0,I}$ as

$$b_0(T_0 u, v^{(0)}) = \hat{a}_h(u, v^{(0)}) \text{ for all } v^{(0)} \in V_{0,I}. \tag{36}$$

This problem is well posed and symmetric with respect to the $\hat{a}_h$-inner product.

The additive preconditioner is defined by

$$T_{as,I} = \sum_{i=0}^{N} T_i. \tag{37}$$

Note that $T_{as,I}$ is symmetric with respect to the inner product $\hat{a}_h(\cdot, \cdot)$.

5.3 Condition Number Estimate for $T_{as,I}$

In this section we state the main result concerning the preconditioner defined in (37) with $V_0 = V_{0,I}$.

Theorem 1. *Let Assumption 1 be satisfied. In addition, assume that for $i \in \mathcal{N}_B$, the size of $\partial\Omega_i \cap \partial\Omega$ is of the same order as the diameter of Ω_i. Then there exist positive constants C_1 and C_2 independent of $h_i, H_i, h_i/h_j$ and the jumps of ρ_i such that*

$$C_1 \hat{a}_h(u, u) \le \hat{a}_h(T_{as,I} u, u) \le C_2 \left(1 + \log \frac{H}{h}\right)^2 \hat{a}_h(u, u) \quad \textit{for all } u \in V. \tag{38}$$

Here $\log(H/h) = \max_i \log(H_i/h_i)$.

Proof. By the general theory of ASMs we need to check three key assumptions; see Theorem 2.7 [10]. The proof can be found in [5].

6 Final Remarks

The ASM considered can be generalized replacing the coarse space $V_{0,I}$, see (34), by adding boundary coarse basis functions, i.e.,

$$V_{0,I\cup B} = \text{Span}\,\{\Theta_i\}_{i\in\mathcal{N}_I\cup\mathcal{N}_B}\,. \tag{39}$$

The additive preconditioner is then defined by

$$T_{as,I\cup B} = \sum_{i=0}^{N} T_i, \tag{40}$$

where the T_0 is defined as in (36) except that now we replace $V_{0,I}$ by $V_{0,I\cup B}$. For this preconditioner, the Theorem 1 is also valid, moreover, it is valid without the assumption that the size of $\partial\Omega_i\cap\partial\Omega$ is of the same order as the diameter of $\partial\Omega_i$ when $i\in\mathcal{N}_B$.

The tools of the discussed methods can be used to design and analyze hybrid (BDD) methods for (20). We can also consider hybrid versions of $T_{as,I\cup B}$, see [5].

The numerical tests carried out for the above methods confirm the theoretical results, see [5]. In particular, Assumption 1 is necessary and sufficient.

The discussed methods can be straightforwardly extended to 3-D cases.

References

1. D.N. Arnold, F. Brezzi, B. Cockburn, and L. Donatella Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, 39(5):1749–1779 (electronic), 2001/02. ISSN 0036-1429.
2. B.A. de Dios and L. Zikatanov. Uniformly convergent iterative methods for discontinuous Galerkin discretizations. *J. Sci. Comput.*, 40(1–3):4–36, 2009. ISSN 0885-7474.
3. M. Dryja. On discontinuous Galerkin methods for elliptic problems with discontinuous coefficients. *Comput. Methods Appl. Math.*, 3(1):76–85 (electronic), 2003. ISSN 1609-4840. Dedicated to Raytcho Lazarov.
4. M. Dryja, J. Galvis, and M. Sarkis. BDDC methods for discontinuous Galerkin discretization of elliptic problems. *J. Complex.*, 23(4–6):715–739, 2007. ISSN 0885-064X.
5. M. Dryja, J. Galvis, and M. Sarkis. Neumann–Neumann methods for a DG discretization on geometrically nonconforming substructures. Technical Report 188, Department of Mathematics, Warsaw University, 2009.
6. X. Feng and O.A. Karakashian. Two-level additive Schwarz methods for a discontinuous Galerkin approximation of second order elliptic problems. *SIAM J. Numer. Anal.*, 39(4): 1343–1365 (electronic), 2001. ISSN 0036-1429.
7. H.H. Kim, M. Dryja, and O.B. Widlund. A BDDC method for mortar discretizations using a transformation of basis. *SIAM J. Numer. Anal.*, 47(1):136–157, 2008/09. ISSN 0036-1429. URL http://dx.doi.org/10.1137/070697859.

8. C. Lasser and A. Toselli. An overlapping domain decomposition preconditioner for a class of discontinuous Galerkin approximations of advection-diffusion problems. *Math. Comput.*, 72(243):1215–1238 (electronic), 2003. ISSN 0025-5718.
9. B. Rivière. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Applications*, volume 35 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008. ISBN 978-0-898716-56-6.
10. A. Toselli and O. Widlund. *Domain Decomposition Methods – Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2005. ISBN 3-540-20696-5.

On Adaptive-Multilevel BDDC

Bedřich Sousedík[1,2] * and Jan Mandel[1] †

[1] Department of Mathematical and Statistical Sciences,
University of Colorado Denver, Denver, CO 80217, USA
[2] Institute of Thermomechanics, Academy of Sciences of the Czech Republic, 182 00
Prague 8, Czech Republic
bedrich.sousedik@ucdenver.edu; jan.mandel@ucdenver.edu

1 Introduction

The BDDC method [2] is one of the most advanced methods of iterative substructuring. In the case of many substructures, solving the coarse problem exactly becomes a bottleneck. Since the coarse problem has the same structure as the original problem, it is straightforward to apply the method recursively to solve it only approximately. The two-level BDDC analysis has been extended into three-levels in a pioneering work in [14, 15], and into a general multilevel method in [11]. The methods for the adaptive selection of constraints for the two-level BDDC method have been studied in [9, 12]. Here we combine the two approaches into a new method preserving parallel scalability with increasing number of subdomains and excellent convergence properties.

The theoretical aspects of the design of the BDDC and a closely related FETI-DP on irregular subdomains in the plane has been studied in [6]. The authors in particular demonstrated that a proper choice of a certain scaling can significantly improve convergence of the methods. Our goal here is different. We consider only the standard stiffness scaling and we look for a space where the action of the BDDC preconditioner is defined. A combination of these two approaches, also with the proper choice of initial constraints [1], would be of independent interest. The presented algorithm has been recently extended into 3D in [13].

[4, 5] have recently successfully developed and extensively used several inexact solvers for the FETI-DP method, and [16] has extended the three-level BDDC methods to the saddle point problems.

All abstract spaces in this paper are finite dimensional. The dual space of a linear space U is denoted by U', and $\langle \cdot, \cdot \rangle$ is the duality pairing.

* Partially supported by National Science Foundation under grant DMS-0713876 and by the Grant Agency of the Czech Republic under grant 106/08/0403.
† Supported by National Science Foundation under grant DMS-0713876.

2 Abstract BDDC for a Model Problem

Let $\Omega \subset \mathbb{R}^2$ be a bounded polygonal domain, decomposed into N nonoverlapping polygonal substructures Ω_i, $i = 1, ..., N$, which form a conforming triangulation. That is, if two substructures have a nonempty intersection, then the intersection is a vertex, or a whole edge. Substructure vertices will also be called corners. Let W_i be the space of Lagrangean $P1$ or $Q1$ finite element functions with characteristic mesh size h on Ω_i, and which are zero on the boundary $\partial\Omega$. Suppose that the nodes of the finite elements coincide on edges common to two substructures. Let

$$W = W_1 \times \cdots \times W_N,$$

and let $U \subset W$ be the subspace of functions that are continuous across the substructure interfaces. We wish to solve the abstract linear problem

$$u \in U : a(u, v) = \langle f, v \rangle , \quad \forall v \in U, \tag{1}$$

for a given $f \in U'$, where a is a symmetric positive semidefinite bilinear form on some space $W \supset U$ and positive definite on U. The form $a(\cdot, \cdot)$ is called the energy inner product, the value of the quadratic form $a(u, u)$ is called the energy of u, and the norm $\|u\|_a = a(u, u)^{1/2}$ is called the energy norm. The operator $A : U \mapsto U'$ associated with a is defined by

$$a(u, v) = \langle Au, v \rangle , \quad \forall u, v \in U.$$

The values of functions from W at the corners and certain averages over the edges will be called the *coarse degrees of freedom*. Let $\widetilde{W} \subset W$ be the space of all functions such that the values of any coarse degrees of freedom have a common value for all relevant substructures and vanish on $\partial\Omega$. Define $U_I \subset U \subset \widetilde{W}$ as the subspace of all functions that are zero on all substructure boundaries $\partial\Omega_i$, $\widetilde{W}_\Delta \subset \widetilde{W}$ as the subspace of all function such that their coarse degrees of freedom vanish, $\widetilde{W}_\Pi$ as the subspace of all functions such that their coarse degrees of freedom between adjacent substructures coincide and such that their energy is minimal. Then

$$\widetilde{W} = \widetilde{W}_\Delta \oplus \widetilde{W}_\Pi, \quad \widetilde{W}_\Delta \perp_a \widetilde{W}_\Pi. \tag{2}$$

The component of the BDDC preconditioner computed in $\widetilde{W}_\Pi$ is called the coarse problem, cf. [11, Algorithm 11]. Functions that are a-orthogonal to U_I are called discrete harmonic. In [7, 9], the analysis was done in spaces of discrete harmonic functions after eliminating U_I; this is not the case here, so $\widetilde{W}$ does not consist of discrete harmonic functions only. Denote by P the energy orthogonal projection from U to U_I. Then $I - P$ is known as the projection onto the discrete harmonic functions. Finally, let E be a projection from $\widetilde{W}$ onto U defined by taking some weighted average over the substructure interfaces.

Let us briefly describe the construction of the space $\widetilde{W}$ using the coarse degrees of freedom. Suppose we are given a space X and a linear operator $C : W \mapsto X$ and define

$$\widetilde{W} = \{w \in W : C\,(I - E)\,w = 0\}\,. \tag{3}$$

The values Cw will be called the local coarse degrees of freedom. To represent their common values, i.e. the global coarse degrees of freedom, suppose there is a space U_c and linear operators

$$Q_P^T : U \to U_c \quad R_c : U_c \to X \quad R : U \to W,$$

such that

$$CR = R_c Q_P^T.$$

The operator Q_P^T selects global coarse degrees of freedom in U_c as linear combinations of global degrees of freedom; a global coarse degree of freedom is given by a row of Q_P. The operator R (resp. R_c) restricts a vector of global (coarse) degrees of freedom into a vector of local (coarse) degrees of freedom. See [9] for more details.

2.1 Multilevel BDDC

The substructuring components (the domains, spaces and operators) from the previous section will be denoted by an additional subscript $_1$, as Ω_1^i, $i = 1, \ldots N_1$, etc., and called level 1. We will call the coarse problem in $\widetilde{W}_{\Pi 1}$ the level 2 problem. It has the same finite element structure as the original problem (1) on level 1, so we have $U_2 = \widetilde{W}_{\Pi 1}$. Level 1 substructures are level 2 elements, level 1 coarse degrees of freedom are level 2 degrees of freedom. The shape functions on level 2 are the coarse basis functions in $\widetilde{W}_{\Pi 1}$, which are given by the conditions that the value of exactly one coarse degree of freedom is one and the others are zero, and that they are energy minimal in $\widetilde{W}_1$. Note that the resulting shape functions on level 2 are in general discontinuous between level 2 elements. Level 2 elements are then agglomerated into nonoverlapping level 2 substructures, etc. Level ℓ elements are level $\ell-1$ substructures, and the level ℓ substructures are agglomerates of level ℓ elements. Level ℓ substructures are denoted by Ω_ℓ^i, and they are assumed to form a quasiuniform conforming triangulation with characteristic substructure size H_ℓ. The degrees of freedom of level ℓ elements are given by level $\ell - 1$ coarse degrees of freedom, and shape functions on level ℓ are determined by minimization of energy on each level $\ell - 1$ substructure separately, so $U_\ell = \widetilde{W}_{\Pi,\ell-1}$. The averaging operators on level ℓ, $E_\ell : \widetilde{W}_\ell \to U_\ell$, are defined by averaging of the values of level ℓ degrees of freedom between level ℓ substructures Ω_ℓ^i. The space $U_{I\ell}$ consists of functions in U_ℓ that are zero on the boundaries of all level ℓ substructures, and $P_\ell : U_\ell \to U_{I\ell}$ is the a-orthogonal projection in U_ℓ onto $U_{I\ell}$. For convenience, let Ω_0^i be the original finite elements, $H_0 = h$.

Algorithm 1 (Multilevel BDDC, [11], Algorithm 17) *Define the preconditioner* $r_1 \in U_1' \longmapsto u_1 \in U_1$ *as follows:*
for $\ell = 1, \ldots, L-1$**,**

Compute interior pre-correction on level ℓ*,*

$$u_{I\ell} \in U_{I\ell} : a\,(u_{I\ell}, z_{I\ell}) = \langle r_\ell, z_{I\ell} \rangle\,, \quad \forall z_{I\ell} \in U_{I\ell}. \tag{4}$$

Get updated residual on level ℓ,

$$r_{B\ell} \in U_\ell, \quad \langle r_{B\ell}, v_\ell \rangle = \langle r_\ell, v_\ell \rangle - a\left(u_{I\ell}, v_\ell\right), \quad \forall v_\ell \in U_\ell. \tag{5}$$

Find the substructure correction on level ℓ,

$$w_{\Delta\ell} \in W_{\Delta\ell} : a\left(w_{\Delta\ell}, z_{\Delta\ell}\right) = \langle r_{B\ell}, E_\ell z_{\Delta\ell} \rangle, \quad \forall z_{\Delta\ell} \in W_{\Delta\ell}. \tag{6}$$

Formulate the coarse problem on level ℓ,

$$w_{\Pi\ell} \in W_{\Pi\ell} : a\left(w_{\Pi\ell}, z_{\Pi\ell}\right) = \langle r_{B\ell}, E_\ell z_{\Pi\ell} \rangle, \quad \forall z_{\Pi\ell} \in W_{\Pi\ell}, \tag{7}$$

If $\ell = L-1$, solve the coarse problem directly and set $u_L = w_{\Pi L-1}$, otherwise set up the right-hand side for level $\ell+1$,

$$r_{\ell+1} \in \widetilde{W}'_{\Pi\ell}, \quad \langle r_{\ell+1}, z_{\ell+1} \rangle = \langle r_{B\ell}, E_\ell z_{\ell+1} \rangle, \quad \forall z_{\ell+1} \in \widetilde{W}_{\Pi\ell} = U_{\ell+1}, \tag{8}$$

end.
for $\ell = L-1, \ldots, 1,$

Average the approximate corrections on substructure interfaces on level ℓ,

$$u_{B\ell} = E_\ell \left(w_{\Delta\ell} + u_{\ell+1}\right). \tag{9}$$

Compute the interior post-correction on level ℓ,

$$v_{I\ell} \in U_{I\ell} : a\left(v_{I\ell}, z_{I\ell}\right) = a\left(u_{B\ell}, z_{I\ell}\right), \quad \forall z_{I\ell} \in U_{I\ell}. \tag{10}$$

Apply the combined corrections,

$$u_\ell = u_{I\ell} + u_{B\ell} - v_{I\ell}. \tag{11}$$

end.

A condition number bound follows, cf. [[11], Lemma 20].

Lemma 1. *If for some $\omega_\ell \geq 1$, for all $\ell = 1, \ldots, L-1$,*

$$\omega_\ell = \sup_{w_\ell \in (I-P_\ell)\widetilde{W}_\ell} J_\ell(w_\ell), \qquad J_\ell(w_\ell) = \frac{\|(I-E_\ell)w_\ell\|_a^2}{\|w_\ell\|_a^2}, \tag{12}$$

then the multilevel BDDC preconditioner satisfies $\kappa \leq \omega = \prod_{k=1}^{L-1} \omega_\ell$.

3 Indicator of the Condition Number Bound

As in [9], we propose as an indicator of the condition number the maximum of the bounds from Lemma 1 computed by considering on each level ℓ only one pair of adjacent substructures s and t at a time:

$$\omega \approx \widetilde{\omega} = \Pi_{\ell=1}^{L-1} \max_{st} \omega_\ell^{st}, \quad \omega_\ell^{st} = \sup_{w_\ell^{st} \in (I-P_\ell^{st})\widetilde{W}_\ell^{st}} J_\ell^{st}\left(w_\ell^{st}\right), \tag{13}$$

where a pair of substructures is called adjacent if they share an edge, and the quantities with the superscript st are defined using the domain consisting of the level ℓ substructures s and t only.

The quantity $\widetilde{\omega}$ is called an *indicator* of the condition number bound.

Let S_ℓ^{st} be the Schur complement operator associated with the bilinear form $a(\cdot,\cdot)$ on the space $(I - P_\ell^{st})\widetilde{W}_\ell^{st}$. The next theorem is [8, Theorem 2] written in a way suitable for our purposes.

Theorem 1 *Let for $a > 0$, Π_ℓ^{st} be the orthogonal projection onto $(I - P_\ell^{st})\widetilde{W}_\ell^{st}$, and $I - \overline{\Pi}_\ell^{st}$ the orthogonal projection onto*

$$\operatorname{null}\left(\Pi_\ell^{st} S_\ell^{st} \Pi_\ell^{st} + a\left(I - \Pi_\ell^{st}\right)\right).$$

Then the stationary values $\omega_{\ell,1}^{st} \geq \omega_{\ell,2}^{st} \geq \ldots$ and the corresponding stationary vectors $w_{\ell,k}^{st}$ of the Rayleigh quotient J_ℓ^{st} in (13) satisfy

$$X_\ell^{st} w_{\ell,k}^{st} = \omega_{\ell,k}^{st} Y_\ell^{st} w_{\ell,k}^{st} \tag{14}$$

with Y_ℓ^{st} positive definite, where

$$\begin{aligned} X_\ell^{st} &= \Pi_\ell^{st}\left(I - E_\ell^{st}\right)^T S_\ell^{st}\left(I - E_\ell^{st}\right)\Pi_\ell^{st}, \\ Y_\ell^{st} &= \left(\overline{\Pi}_\ell^{st}\left(\Pi_\ell^{st} S_\ell^{st} \Pi_\ell^{st} + a\left(I - \Pi_\ell^{st}\right)\right)\overline{\Pi}_\ell^{st} + a\left(I - \overline{\Pi}_\ell^{st}\right)\right). \end{aligned}$$

The eigenvalue problem (14) is obtained by projecting the gradient of the Rayleigh quotient $J_\ell^{st}(w_\ell^{st})$ onto the complement in $(I - P_\ell^{st})\widetilde{W}_\ell^{st}$ of the subspace, where its denominator is zero, in two steps. Both projections Π_ℓ^{st} and $\overline{\Pi}_\ell^{st}$ are computed by matrix algebra, which is straightforward to implement numerically. The projection Π_ℓ^{st} projects onto $\operatorname{null} C_\ell^{st}\left(I - E_\ell^{st}\right)$, and $I - \overline{\Pi}_\ell^{st}$ projects onto a subspace of $\operatorname{null} S_\ell^{st}$, which can be easily constructed computationally if a matrix Z_ℓ^{st} is given such that $\operatorname{null} S_\ell^{st} \subset \operatorname{range} Z_\ell^{st}$. For this purpose, the rigid body modes are often available directly or they can be computed from the geometry of the finite element mesh. For levels $\ell > 1$, we can use the matrix Z_ℓ^{st} with columns consisting of coarse basis functions, because the span of the coarse basis functions contains the rigid body modes. In this way, we can reduce (14) to a symmetric eigenvalue problem, which is easier and more efficient to solve numerically.

4 Optimal Coarse Degrees of Freedom

Writing $\widetilde{W}_\ell^{st} = \operatorname{null} C_\ell^{st} \left(I - E_\ell^{st}\right)$ suggests how to add coarse degrees of freedom to decrease the value of indicator $\widetilde{\omega}$. The following theorem is an analogy of [8, Theorem 3]. It follows immediately from the standard characterization of eigenvalues as minima and maxima of the Rayleigh quotient on subspaces spanned by eigenvectors, applied to (14).

Theorem 2. *Suppose $n_\ell^{st} \geq 0$ and the coarse dof selection matrix $C_\ell^{st}\left(I - E_\ell^{st}\right)$ is augmented by the rows $w_{\ell,k}^{stT}\left(I - E_\ell^{st}\right)^T S_\ell^{st}\left(I - E_\ell^{st}\right)$, where $w_{\ell,k}^{st}$ are the eigenvectors from (14). Then $\omega_\ell^{st} = \omega_{\ell,n_\ell^{st}+1}^{st}$, and $\omega_\ell^{st} \geq \omega_{\ell,n_\ell^{st}+1}^{st}$ for any other augmentation by at most n_ℓ^{st} columns.*

In particular, if $\omega_{\ell,n_\ell^{st}+1}^{st} \leq \tau$ for all pairs of adjacent substructures s, t and for all levels $\ell = 1, \ldots, L-1$, then $\widetilde{\omega} \leq \tau^{L-1}$.

Theorem 2 allows us to guarantee that the condition number indicator $\widetilde{\omega} \leq \tau^{L-1}$ for a given target value τ, by adding the smallest possible number of coarse degrees of freedom.

The primal coarse space selection mechanism that corresponds to this augmentation can be explained as follows. Let us write the augmentation as

$$c_{\ell,k}^{st} = \left[\, c_{\ell,k}^{s} \; c_{\ell,k}^{t} \,\right] = w_{\ell,k}^{stT}\left(I - E_\ell^{st}\right)^T S_\ell^{st}\left(I - E_\ell^{st}\right),$$

where $c_{\ell,k}^{s}$ and $c_{\ell,k}^{t}$ are blocks corresponding to substructures s and t. It should be noted that the matrix E_ℓ^{st} is constructed for a pair of substructures s, t in such a way that, cf., e.g. [10, Eq. (7)],

$$B_{D,\ell}^{stT} B_\ell^{st} = I - E_\ell^{st},$$

where $B_{D,\ell}^{st}$ and B_ℓ^{st} are matrices known from the FETI-DP method. In particular, the entries of B_ℓ^{st} are $+1$ for substructure s and -1 for substructure t. This also relates our algorithm to the one from [8, 9]. Next, let us observe that, due to the application of $I - E_\ell^{st}$, for the two blocks of $c_{\ell,k}^{st}$ it holds that $c_{\ell,k}^{s} = -c_{\ell,k}^{t}$, i.e., for the two substructures the constraint weights have the same absolute values and opposite sign. Hence it is sufficient to consider only one of the two blocks, e.g., $c_{\ell,k}^{s}$. The augmentation of the global coarse degrees of freedom selection matrix $[Q_{P,\ell}, q_{k,\ell}]$ is constructed by adding a block of k columns computed as

$$q_{k,\ell} = R_\ell^{sT} c_k^{sT}.$$

Each column of $q_{k,\ell}$ defines a coarse degree of freedom associated with the interface of level ℓ substructures s and t. Because R_ℓ^s is a $0-1$ matrix, it means that columns in $q_{k,\ell}$ are formed by a scattering of the entries in c_k^{sT}.

5 Adaptive-Multilevel BDDC in 2D

We describe in more detail the implementation of the algorithm. It consists of two main steps: (i) setup, and (ii) the loop of preconditioned conjugate gradients with the Adaptive-Multilevel BDDC as a preconditioner. The setup was outlined in the previous section, and it can be summarized as follows:

Algorithm 2 *Adding of coarse degrees of freedom in order to guarantee that the condition number indicator $\widetilde{\omega} \leq \tau^{L-1}$, for a given a target value τ:*

for levels $\ell = 1 : L-1$,

Create substructures with roughly the same numbers of degrees of freedom, minimizing the number of "cuts"(use a graph partitioner, e.g., METIS 4.0 ([3]) with weights on both, vertices and edges).

Find a suitable set of initial constraints (corners in 2D), and set up the BDDC structures for the adaptive algorithm.

for all edges $\mathcal{E}_\ell$ ***on level*** ℓ,

Compute the largest local eigenvalues and corresponding eigenvectors, until the first m^{st} is found such that $\lambda^{st}_{m^{st}} \leq \tau$, put $k = 1, \ldots, m^{st}$.

Compute the constraint weights $c^{st}_k = [c^s_k \; c^t_k]$ as

$$c^{st}_k = w^{stT}_k \Pi^{Ist}_\ell \left(I - E^{st}_\ell\right)^T S^{st}_\ell \left(I - E^{st}_\ell\right) \Pi^{Ist}_\ell, \tag{15}$$

where Π^{Ist}_ℓ is a projection constructed using the set of initial constraints.

Take one block, e.g., c^s_k and keep nonzero weights for the edge $\mathcal{E}_\ell$.

Add to the global coarse dofs selection matrix $Q_{P,\ell}$ the k columns $q_{k,\ell}$ as

$$q_{k,\ell} = R^{sT}_\ell c^{sT}_k. \tag{16}$$

end.

Setup the BDDC structures for level ℓ and check size of the coarse problem: if small enough, call this the level L problem, factor it directly, and exit from the loop.

end.

We remark that the adaptive algorithm is significantly simpler and easier to implement compared to our previous algorithm from [8, 9]. The constraints in (15) are generated from the eigenvectors by the same function that evaluates the left hand side in (14). Then they are "torn" into two blocks, and entries of one of them, that correspond to a particular edge shared by substructures s and t on the level ℓ, are scattered into additional columns of the matrix $Q_{P,\ell}$.

The adaptive algorithm uses matrices and operators that are readily available in our implementation of the standard BDDC method (unlike in [12] this time with an explicit coarse space solve) with one exception: in order to satisfy the local partition of unity property, cf. [10, Eq. (9)],

$$E_\ell^{st} R_\ell^{st} = I,$$

we need to generate locally the weight matrices E_ℓ^{st}.

The substructures on higher levels are then treated as (coarse) elements with energy minimal basis functions. However, the number of added constraints is a-priori unknown. For this reason, the coarse elements must allow for variable number of nodes per element, and also for variable number of degrees of freedom per node. It is also essential to generate a sufficient number of corners as initial constraints, in particular to prevent rigid body motions between any pair of adjacent substructures. This topic has been addressed several times cf., e.g., a recent contribution in [1].

Finally, we remark that instead of performing interior pre-correction and post-correction on level $\ell = 1$, cf. Eqs. (4)–(5) and (10)–(11), we can benefit from reducing the problem to interfaces in the pre-processing step.

6 Numerical Examples and Conclusion

The adaptive-multilevel BDDC preconditioner was implemented in Matlab for the 2D linear elasticity problem (with $\lambda = 1$, and $\mu = 2$) on a square domain discretized by finite elements with $1{,}182{,}722$ degrees of freedom. The domain was decomposed into $2{,}304$ subdomains on the second level and into 9 subdomains on the third-level. Such decomposition leads to the coarsening ratio $H_\ell/H_{\ell-1} = 16$, with $\ell = 1, 2$. In order to test the adaptive selection of constraints, one single edge has been jagged on both decomposition levels, see Fig. 1. We have computed the eigenvalues and eigenvectors of (14) by setting up the matrices and using standard methods for the symmetric eigenvalue problem in Matlab, version 7.8.0.347 (R2009a).

In the first set of experiments, we have compared performance of the non-adaptive BDDC method with 2 and 3 decomposition levels. The results are presented in Tables 1 and 2. As expected from the theory the convergence of the algorithm deteriorates when additional levels are introduced.

In the next set of experiments, we have tested the adaptive algorithm for the two-level BDDC. The results are summarized in Table 6. The algorithm performs consistently with our previous formulation in [9]. The eigenvalues associated with edges between substructures clearly distinguish between the single problematic edge and the others (Table 3). Adding the coarse dofs created from the associated eigenvectors according to Theorem 2 decreases the value of the condition number indicator $\widetilde{\omega}$ and improves convergence at the cost of increasing the number of coarse dofs.

Finally, we have tested the performance of the Adaptive-Multilevel BDDC for the model problem with three-level decomposition (Fig. 1). Because the number of coarse degrees of freedom depends on an a-priori chosen value of τ and the coarse

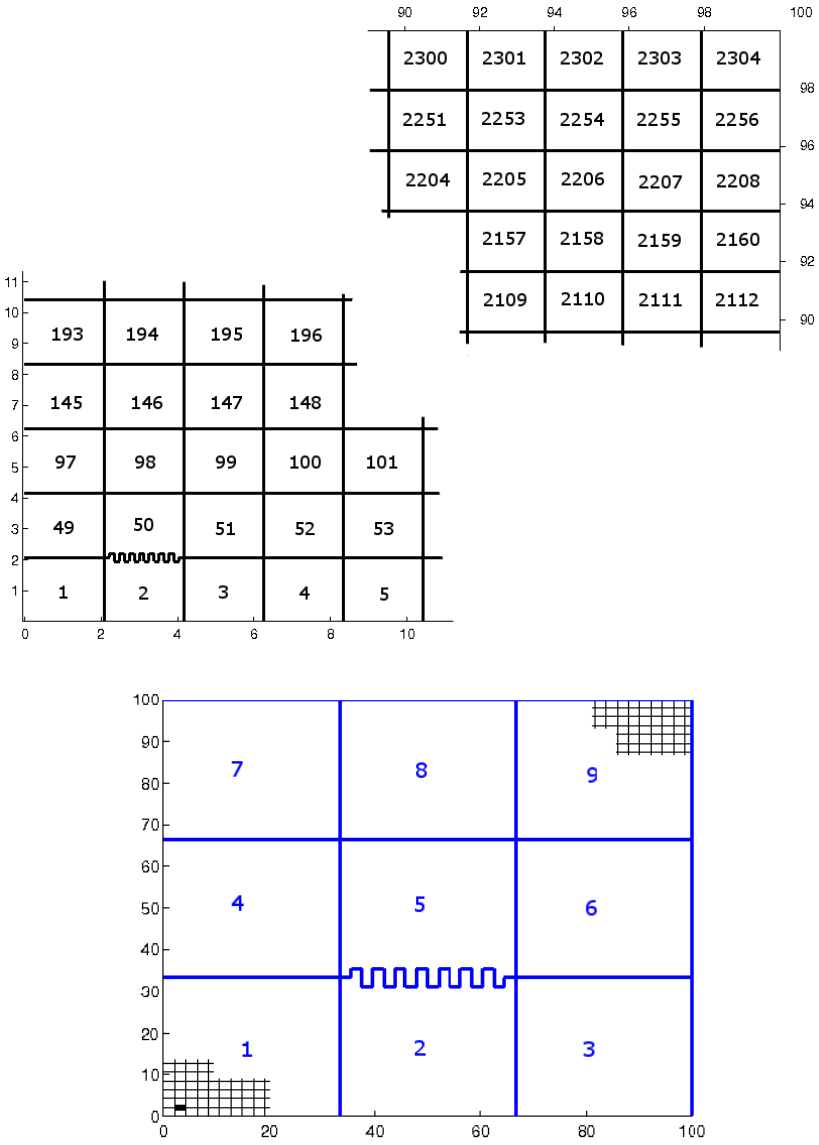


Fig. 1. The two-level decomposition into $48 \times 48\,(= 2{,}304)$ subdomains (*top*), and the decomposition into 9 subdomains for the three-level method (*bottom*); the jagged edge from the lower decomposition level is indicated here by a *thick line*.

Table 1. Results for non-adaptive 2-level method. Constraints are corners, or corners and arithmetic averages over edges, denoted as c, c+e, resp. Nc is number of constraints, $\mathcal{C}$ is size of the coarse problem related to size of a subdomain problem, κ is the condition number estimate, it is number of iterations (tol. 10^{-8}).

Constraint	Nc	$\mathcal{C}$	κ	it
c	4794	9.3	18.41	43
c+e	13818	26.9	18.43	32

Table 2. Results for non-adaptive 3-level method. Headings are as in Table 1.

Constraint	Nc	$\mathcal{C}$	κ	it
c	$4794 + 24$	1.0	67.5	74
c+e	$13818 + 48$	3.0	97.7	70

Table 3. Eigenvalues of the local problems for several pairs of subdomains s and t on the decomposition level $\ell = 1$ (the jagged edge is between subdomains 2 and 50).

s	t	$\lambda_{st,1}$	$\lambda_{st,2}$	$\lambda_{st,3}$	$\lambda_{st,4}$	$\lambda_{st,5}$	$\lambda_{st,6}$	$\lambda_{st,7}$	$\lambda_{st,8}$
1	2	3.8	2.4	1.4	1.3	1.2	1.1	1.1	1.1
1	49	6.0	3.5	2.7	1.4	1.3	1.1	1.1	1.1
2	3	5.4	2.6	1.6	1.3	1.2	1.1	1.1	1.1
2	50	24.3	18.4	18.3	16.7	16.7	14.7	13.5	13.1
3	4	3.4	2.4	1.4	1.3	1.1	1.1	1.1	1.1
3	51	7.4	4.6	3.7	1.7	1.4	1.3	1.2	1.1
49	50	12.6	5.1	4.3	1.9	1.6	1.3	1.2	1.2
50	51	8.7	4.8	3.9	1.8	1.5	1.3	1.2	1.2
50	98	7.5	4.6	3.7	1.7	1.4	1.3	1.2	1.1

basis functions on level ℓ become shape basis functions on level $\ell + 1$, the solutions of local eigenvalue problems will depend on τ as well. This fact is illustrated by Table 4 for $\tau = 2$, and by Table 5 for $\tau = 10$ (the local eigenvalues for $\tau = 3$ were essentially same as for $\tau = 2$). Comparing the values in these two tables, we see that lower values of τ result in worse conditioning of the local eigenvalue problems on higher decomposition level. This immediately gives rise to a conjecture that it might not be desirable to decrease the values of τ arbitrarily low in order to achieve a better convergence of the method. On the other hand, for the model problem, comparing the convergence results for the two-level method (Table 6) with the three-level method (Table 7), we see that with the adaptive constraints we were able to achieve nearly the same convergence properties for the two methods.

Table 4. Eigenvalues of the local problems for several pairs of subdomains s, t on level $\ell = 2$ with $\tau = 2$ (the jagged edge is between subdomains 2 and 5).

s	t	$\lambda_{st,1}$	$\lambda_{st,2}$	$\lambda_{st,3}$	$\lambda_{st,4}$	$\lambda_{st,5}$	$\lambda_{st,6}$	$\lambda_{st,7}$	$\lambda_{st,8}$
1	2	16.5	9.0	5.4	2.6	2.1	1.4	1.3	1.3
1	4	6.5	4.7	1.9	1.7	1.3	1.2	1.2	1.1
2	3	23.1	9.4	4.6	3.2	2.1	1.6	1.4	1.3
2	5	84.3	61.4	61.4	55.9	55.8	49.3	48.0	46.9
3	6	13.7	8.8	4.4	2.2	1.9	1.4	1.3	1.2
4	7	6.5	4.7	1.9	1.7	1.3	1.2	1.2	1.1
5	6	18.9	13.1	11.3	3.8	2.6	2.1	1.9	1.5
5	8	17.3	12.9	10.8	3.6	2.3	2.0	1.8	1.4
8	9	13.7	8.8	4.4	2.2	1.9	1.4	1.3	1.2

Table 5. Eigenvalues of the local problems for several pairs of subdomains s, t on level $\ell = 2$ with $\tau = 10$ (the jagged edge is between subdomains 2 and 5).

s	t	$\lambda_{st,1}$	$\lambda_{st,2}$	$\lambda_{st,3}$	$\lambda_{st,4}$	$\lambda_{st,5}$	$\lambda_{st,6}$	$\lambda_{st,7}$	$\lambda_{st,8}$
1	2	7.7	4.5	2.7	1.6	1.4	1.2	1.2	1.1
1	4	3.6	3.0	1.5	1.5	1.2	1.2	1.1	1.1
2	3	10.9	4.8	2.7	1.7	1.5	1.2	1.2	1.1
2	5	23.2	17.2	13.7	13.7	12.7	12.4	11.0	10.9
3	6	6.1	4.2	2.5	1.5	1.3	1.2	1.1	1.1
4	7	3.6	3.0	1.5	1.5	1.2	1.2	1.1	1.1
5	6	9.8	6.2	4.1	2.1	1.6	1.5	1.3	1.2
5	8	8.6	5.9	3.9	2.0	1.5	1.4	1.2	1.2
8	9	6.1	4.2	2.5	1.5	1.3	1.2	1.1	1.1

Table 6. Results for the adaptive 2-level method. Headings are same as in Table 1, and τ is the condition number target, $\widetilde{\omega}$ is the condition number indicator.

τ	Nc	$\mathcal{C}$	$\widetilde{\omega}$	κ	it
∞(=c)	4,794	9.3	–	18.41	43
10	4,805	9.4	8.67	8.34	34
3	18,110	35.3	2.67	2.44	15
2	18,305	35.7	1.97	1.97	13

Table 7. Results for the adaptive 3-level method. Headings are as in Table 6, but the threshold τ is now used on each of the two decomposition levels and so $\widetilde{\omega} \leq \tau^2$.

τ	Nc	$\mathcal{C}$	$\widetilde{\omega}$	κ	it
∞(=c)	4,794 + 24	1.0	–	67.5	74
10	4,805 + 34	1.0	84.97	37.42	60
3	18,110 + 93	3.9	7.88	3.11	19
2	18,305 + 117	4.0	3.84	2.28	15

References

1. P. Burda, M. Čertíková, J. Damašek, A. Novotný, and J. Šístek. Selection of corners for the BDDC method. Submitted to Math. Comput. Simulation, 2009.
2. C.R. Dohrmann. A preconditioner for substructuring based on constrained energy minimization. *SIAM J. Sci. Comput.*, 25(1):246–258, 2003.
3. G. Karypis and V. Kumar. METIS: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices, version 4.0. Technical Report, Department of Computer Science, University of Minnesota, 1998.
4. A. Klawonn and O. Rheinbach. Inexact FETI-DP methods. *Int. J. Numer. Methods Eng.*, 69(2):284–307, 2007.
5. A. Klawonn and O. Rheinbach. A hybrid approach to 3-level FETI. *PAMM*, 8(1):10841–10843, 2008.
6. A. Klawonn, O. Rheinbach, and O.B. Widlund. An analysis of a FETI-DP algorithm on irregular subdomains in the plane. *SIAM J. Numer. Anal.*, 46(5):2484–2504, 2008.
7. J. Mandel, C.R. Dohrmann, and R. Tezaur. An algebraic theory for primal and dual substructuring methods by constraints. *Appl. Numer. Math.*, 54(2):167–193, 2005.
8. J. Mandel and B. Sousedík. Adaptive coarse space selection in the BDDC and the FETI-DP iterative substructuring methods: Optimal face degrees of freedom. In Olof B. Widlund and David E. Keyes, editors, *Domain Decomposition Methods in Science and Engineering XVI, Volume 55 of Lecture Notes in Computational Science and Engineering*, pp. 421–428. Springer Heidelberg, Berlin, New York, 2006.
9. J. Mandel and B. Sousedík. Adaptive selection of face coarse degrees of freedom in the BDDC and the FETI-DP iterative substructuring methods. *Comput. Methods Appl. Mech. Eng.*, 196(8):1389–1399, 2007.
10. J. Mandel and B. Sousedík. BDDC and FETI-DP under minimalist assumptions. *Computing*, 81:269–280, 2007.
11. J. Mandel, B. Sousedík, and C.R. Dohrmann. Multispace and Multilevel BDDC. *Computing*, 83(2–3):55–85, 2008.
12. J. Mandel, B. Sousedík, and J. Šístek. Adaptive BDDC in three dimensions. Submitted to Math. Comput. Simulation, 2009.
13. B. Sousedík. *Adaptive-Multilevel BDDC*. PhD thesis, University of Colorado Denver, Department of Mathematical and Statistical Sciences, 2010.
14. X. Tu. Three-level BDDC in three dimensions. *SIAM J. Sci. Comput.*, 29(4):1759–1780, 2007.
15. X. Tu. Three-level BDDC in two dimensions. *Int. J. Numer. Methods Eng.*, 69(1):33–59, 2007.
16. X. Tu. A three-level BDDC algorithm for saddle point problems. Submitted to Numer. Math., 2008.

Interpolation Based Local Postprocessing for Adaptive Finite Element Approximations in Electronic Structure Calculations

Jun Fang[1], Xingyu Gao[2], Xingao Gong[3], and Aihui Zhou[1]

[1] LSEC, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China,
fangjun@lsec.cc.ac.cn; azhou@lsec.cc.ac.cn
[2] HPCC, Institute of Applied Physics and Computational Mathematics, Beijing 100094, China,
gao_xingyu@iapcm.ac.cn
[3] Department of Physics, Fudan University, Shanghai 200433, China,
xggong@fudan.edu.cn

Summary. In this paper, we propose an interpolation based local postprocessing approach for finite element electronic structure calculations over locally refined hexahedral finite element meshes. It is shown that our approach is very efficient in finite element approximations of ground state energies.
Key words: adaptive finite element, eigenvalue, electronic structure, interpolation, Kohn–Sham equation, local postprocessing

1 Introduction

It is significant to obtain the ground state energy in the electronic structure study. In modern electronic structure calculations, the ground state energy is usually obtained from solving the Kohn–Sham equation [4, 17]. A general concern is the Kohn–Sham equation of a confined system posed on a bounded domain $\Omega \subset \mathbb{R}^3$:

$$\begin{cases} (-\frac{1}{2}\Delta + V_{eff}(\rho))\psi_i = \epsilon_i \psi_i, & \text{in } \Omega, \\ \psi_i = 0, & \text{on } \partial\Omega, \quad i = 1, \ldots, N_s, \end{cases} \tag{1}$$

where $\rho(\mathbf{r}) \equiv \sum_{i=1}^{N_s} f_i |\psi_i(\mathbf{r})|^2$ is the electron density, N_s the number of electron orbitals ψ_i with associated occupancy number f_i, and $V_{eff}(\rho)$ the so-called effective potential that is a nonlinear functional of ρ.

To solve nonlinear eigenvalue problem (1), a self-consistent approach such as DIIS (Direct Inversion Iterative Subspace) or Pulay's method in [22, 23] is required. As a result, the central computation in solving the Kohn–Sham equation is the repeated solution of the following type of linear eigenvalue problem:

$$\begin{cases} -\Delta u + Vu = \lambda u, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases} \tag{2}$$

where V is some effective potential. Since the electron density at the ground state decays exponentially [2, 13, 27], we may set Ω to be a cube in the computation. Note that even though V in (2) was relatively smooth in the pseudopotential setting, the eigenfunctions of (2) would vary rapidly in the neighborhood of the nuclei but be diffuse further away. Thus some efficient multi-resolution is significant for approximating eigenfunctions in the real space [4, 6, 20].

The multi-resolution can be achieved by adaptive finite element discretizations. Indeed, the preponderant strength of the finite element method lies in its ability to arrange local refinements in the regions where there are strong variations and high resolution is needed while treating the distant zones from nuclei at a coarser scale. We refer the reader to [4, 8, 9, 10, 21, 28, 33] and the references cited therein for the applications of finite element methods to electronic structure calculations. In this work, adaptive hexahedral finite elements will be studied for a better accuracy and efficiency on such a cubic domain [5, 11].

Once finite element eigenfunctions reach the self-consistent convergence, some postprocessing techniques are worth while to enhance the approximations when the extra cost is low. Indeed, the effectiveness of finite element postprocessing has been already shown in [8, 9, 10, 26]. In this paper we propose an interpolation based local postprocessing scheme for finite element quantum eigenvalue approximations and apply the approach to improve the ground state energy approximation. This scheme is derived from our understanding of the behavior of wavefunctions. For a quantum many-particle system, there is a general principle of locality or "nearsightedness" that the properties at one point may be considered independent of what happens at distant points [14, 16, 20]. And wavefunctions of a quantum many-particle system are somehow smooth and oscillate in the region where the system is located only [2, 15, 32]. Thus local higher order finite elements should be used (c.f., e.g., [8, 9, 10]). The computational complexity of higher order finite element discretizations, however, is larger than that of lower order finite element discretizations. To reduce the complexity, in this paper, we will propose some higher order interpolation approach for fast higher order finite element eigenvalue approximations. This approach is a local postprocessing on the lower order finite element approximations with little extra cost.

Now let us give some more details for an illustrative exhibition of the main idea in this paper. The trilinear finite element eigenfunctions are expanded by the basis of trilinear finite elements distributed on the locally refined mesh. In the case of the self-consistent convergence, we locate the father cell with eight children lying at the finest level of the hierarchy of grids. Based on trilinear finite element solutions on the children, a new eigenfunction approximation can be easily constructed as a

triquadratic polynomial on this father cell. And the local accuracy enhancement of eigenfunctions in the high-resolution regions will effectively improve the approximations of the associated eigenvalues as well as ground state energies by Rayleigh quotients.

Our interpolation approach may be viewed as an averaging technique over adaptive finite element meshes while the existing averaging technique for quantum eigenvalue approximations in [26] is set for the gradient of eigenfunctions, in particular, employs some global projection. It is significant that our interpolation based local postprocessing is carried out only over the local domain where the molecular system is located. More notably, there are no auxiliary degrees of freedom needed by our postprocessing since the high-order interpolation is locally constructed over the selected father cells at the coarser level next to the finest. So, our approach is good at memory requirement and computation complexity. The theoretical tool for motivating this idea is the local error estimates for finite element approximations developed in [29, 31] (see also Sect. 2.1). We should mention that an interpolation global postprocessing is first introduced in [18] for finite element eigenvalue approximations over uniformly finite element meshes.

It is shown numerically that our scheme is a potentially efficient postprocessing technique for computing quantum eigenvalues (see Sect. 3.2). In fact, the computed electron density in the region of the system can be improved by the local high-order interpolation postprocessing. So it is expected that our approach would also benefit calculations of other quantum quantities.

The rest of this paper is organized as follows. In the next section, we first introduce our hexahedral finite element discretizations and then illustrate the local interpolation postprocessing theoretically and numerically. We present some applications to electronic structure calculations in Sect. 3 and finally we provide some concluding remarks.

2 Interpolation Based Finite Element Postprocessing

In this section, we shall first describe some basic notation and a finite element discretization for eigenvalue problem (2) and then introduce our local interpolation postprocessing, which will be supported by numerical experiments for a model example.

We shall use the standard notation for Sobolev spaces $W^{s,p}(\Omega)$ and their associated norms and seminorms (see, e.g., [1]). For $p = 2$, we denote $H^s(\Omega) = W^{s,2}(\Omega)$ and $H_0^1(\Omega) = \{v \in H^1(\Omega) : v|_{\partial\Omega} = 0\}$, where $v|_{\partial\Omega} = 0$ is in the sense of trace, $\|\cdot\|_{s,\Omega} = \|\cdot\|_{s,2,\Omega}$ and $\|\cdot\|_{\Omega} = \|\cdot\|_{0,2,\Omega}$. (In some places in this paper, $\|\cdot\|_{s,2,\Omega}$ should be viewed as piecewise defined if necessary.)

Throughout this paper, we shall use the letter C (with or without subscripts) to denote a generic positive constant which may stand for different values at its different occurrences. One basic assumption on the mesh is that the level difference of two adjacent cells cannot be more than one. For $D \subset \Omega_0 \subset \Omega$, we use the notation $D \subset\subset \Omega_0$ to mean that $\text{dist}(\partial D \setminus \partial\Omega, \partial\Omega_0 \setminus \partial\Omega) > 0$.

Let $\Omega = (a,b)^3$ with $a,b \in \mathbb{R}$. Let $T^h(\Omega)$ consist of hexahedra whith edges parallel to x-axis, y-axis and z-axis respectively, where h is the mesh size. Define

$$S^h(\Omega) = \{v \in C(\bar{\Omega}) : v|_\tau \in Q_\tau \ \forall \tau \in T^h(\Omega)\}, \tag{3}$$

where $Q_\tau = \text{span}\{x^i y^j z^k : 0 \le i,j,k \le 1\}$. Set $S_0^h(\Omega) = S^h(\Omega) \cap H_0^1(\Omega)$. These are Lagrange finite element spaces. We refer the reader to [7, 29] (see also [24, 25]) for their basic properties that will be used in our analysis.

If $I_h : C(\bar{\Omega}) \longrightarrow S^h(\Omega)$ is the trilinear Lagrange finite element interpolation operator associated with $T^h(\Omega)$, then we derive from integration by parts that (see, e.g., [12, 19])

$$\left| \int_\tau \nabla(w - I_h w)\nabla v \right| \le C h_\tau^2 |w|_{3,\tau} |\nabla v|_{0,\tau} \ \ \forall v \in S^h(\Omega), \ \forall \tau \in T^h(\Omega), \tag{4}$$

where h_τ is the diameter of τ.

2.1 Finite Element Discretizations

A standard finite element discretization for (2) is: Find a pair of $(\lambda_h, u_h) \in \mathbb{R} \times S_0^h(\Omega)$ satisfying $\|u_h\|_{0,\Omega} = 1$ and

$$a(u_h, v) = \lambda_h (u_h, v) \ \ \forall v \in S_0^h(\Omega). \tag{5}$$

We use (λ_h, u_h) as an approximation to $(\lambda, u) \in \mathbb{R} \times H_0^1(\Omega)$, where (λ, u) is a solution of

$$a(u, v) = \lambda (u, v) \ \ \forall v \in H_0^1(\Omega) \tag{6}$$

with $\|u\|_{0,\Omega} = 1$ and

$$a(w, v) = \int_\Omega \frac{1}{2} \nabla w \nabla v + V w v \ \ \forall w, v \in H_0^1(\Omega).$$

If $V \in L^\infty(\Omega)$, then the associated exact eigenfunction $u \in H_0^1(\Omega) \cap H^2(\Omega)$. Thus we may assume that (see, e.g., [3])

$$|\lambda - \lambda_h| + h\|\nabla(u - u_h)\|_{0,\Omega} + \|u - u_h\|_{0,\Omega} \le C h^2. \tag{7}$$

Let $P_h : H_0^1(\Omega) \longrightarrow S_0^h(\Omega)$ be the Galerkin projection defined by

$$a(w - P_h w, v) = 0 \ \ \forall v \in H_0^1(\Omega). \tag{8}$$

Then we have (see [30])

Proposition 1. *There holds*

$$\|P_h u - u_h\|_{1,\Omega} \le C h^2. \tag{9}$$

2.2 Interpolation Based Local Postprocessing

Let Ω_0 be a subdomain of Ω. The following local superclose result can be derived from (4) and the local error estimation of finite element Galerkin approximations (see, e.g., [19])

Proposition 2. *Let $D \subset\subset \Omega_0$. If $u \in H_0^1(\Omega) \cap H^3(\Omega_0)$, then*

$$\|P_h u - I_h u\|_{1,D} \leq Ch^2. \tag{10}$$

It is seen that we can define a triquadratic Lagrange interpolation Π_{2h} on any father cell $\square$ that consists of 27 children elements in $T^h(\Omega)$. Let Ω_0 be covered by a group of father cells and aligned with $T^h(\Omega)$. Note that

$$\begin{aligned} \Pi_{2h} I_h &= \Pi_{2h}, \\ \|\nabla \Pi_{2h} v\|_{0,\square} &\leq \|\nabla v\|_{0,\square} \ \ \forall v \in S^h(\Omega), \\ \|\Pi_{2h} w - w\|_{1,\square} &\leq Ch^2 \|w\|_{3,\square}. \end{aligned}$$

We obtain

$$\|\Pi_{2h} u_h - u\|_{1,D} \leq Ch^2 \tag{11}$$

from Proposition 1, Proposition 2, and the identity

$$\Pi_{2h} u_h - u = \Pi_{2h}(u_h - P_h u) + \Pi_{2h}(P_h u - I_h u) + \Pi_{2h} u - u.$$

We may use some a Rayleigh quotient to get a new eigenvalue approximation λ^h as follows

$$\lambda^h = \frac{a(u^h, u^h)}{\| u^h \|_{0,\Omega}^2},$$

where

$$u^h = \begin{cases} \Pi_{2h} u_h & , \text{in } \bar{\Omega}_0, \\ u_h & , \text{in } \Omega \setminus \bar{\Omega}_0. \end{cases} \tag{12}$$

Indeed, our numerical experiments show that λ^h is much more accurate than λ_h even if Ω_0 is a part of Ω where local quadratic interpolation Π_{2h} can be carried out.

2.3 Quantum Harmonic Oscillator

For illustration, we consider an oscillator model, which is a simple problem in quantum mechanics:

$$-\frac{1}{2}\Delta u + \frac{1}{2}|x|^2 u = \lambda u, \qquad \text{in } \mathbb{R}^3. \tag{13}$$

The first eigenvalue of (13) is $\lambda = 1.5$ and is associated with the eigenfunction $u = \gamma e^{-\frac{|x|^2}{2}}$, where γ is a nonzero constant so that $\|u\|_{0,\mathbb{R}^3} = 1$.

In our experiments, we choose $\Omega = (-5.0, 5.0)^3$ as the computational domain, on which the zero Dirichlet boundary condition is imposed. We use a uniform mesh as the initial mesh. We carry out local refinements on subdomain $\Omega_0 = (-2.5, 2.5)^3$ by uniformly refining once and consider $D = (1.25, 1.25)^3$. Let(λ_h, u_h), (λ^h, u^h) be the trilinear finite element approximation to (λ, u) and the interpolation postprocessing eigenpair, respectively. Define

$$e_h = |\lambda_h - \lambda|\,,\ e^h = \left|\lambda^h - \lambda\right|.$$

$$\eta_h = \|\nabla(u_h - u)\|_{0,D}\,,\ \eta^h = \|\nabla(\Pi_{2h}u_h - u)\|_{0,D}\,.$$

Numerical results in Table 1 show the errors of the first eigenpair which supports our theory.

Table 1. Oscillator: interpolation on Ω_0.

Initial mesh size	η_h	Order(η_h)	e_h	η^h	Order(η^h)	e^h
$1/2^{-3} \times 10.0$	0.20743		0.03846	0.14364		0.01407
$1/2^{-4} \times 10.0$	0.10508	0.98114	0.00975	0.03265	2.13730	0.00141
$1/2^{-5} \times 10.0$	0.05269	0.99589	0.00244	0.00817	1.99868	0.00024
$1/2^{-6} \times 10.0$	0.02636	0.99918	0.00061	0.00205	1.99471	0.00005

3 Applications to Electronic Structure Calculations

Now we apply the interpolation based local postprocessing approach to solving Kohn–Sham equation (1), from which we see that highly accurate finite element approximations can be obtained over adaptive finite element meshes by using triquadratic interpolation postprocessing on each father cell of the coarser level next to the finest of the grid hierarchy.

3.1 Linearization of Kohn–Sham Equation

Since Kohn–Sham equation (1) is a nonlinear eigenvalue system, we need to linearize and solve it iteratively, which is called self-consistent field iteration (SCF). The SCF iteration is described as follows:

1. Given an initial electron density $\rho_{in}(\mathbf{r})$.
2. Compute $V_{eff}(\rho_{in})$ and solve

$$\begin{cases} (-\frac{1}{2}\Delta + V_{eff}(\rho_{in}))\psi_i = \epsilon_i\psi_i, & \text{in } \Omega, \\ \psi_i = 0, & \text{on } \partial\Omega, \\ \int_\Omega \psi_i\psi_j = \delta_{ij}, & i,j = 1,\ldots,N_s. \end{cases} \tag{14}$$

3. Set $\rho_{out} = \sum_{i=1}^{N_s} f_i|\psi_i(\mathbf{r})|^2$.

4. Compute the difference between ρ_{in} and ρ_{out}. If the difference is not small enough, "mix" density using Pulay's method to obtain the new ρ_{in}, repeat from Step 2. Otherwise stop.

In our computation, Pulay's method [22, 23] will be used. After self-consistent convergence is reached, we (carry out the postprocessing and) compute the total energy of the ground state [20]:

$$E_{tot} = \sum_{i=1}^{N_s} f_i \epsilon_i - \int_\Omega d\mathbf{r} V_{xc}(\mathbf{r})\rho(\mathbf{r}) - \frac{1}{2}\int_\Omega \int_\Omega \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} + E_{xc}(\rho) + \frac{1}{2}\sum_{I,J=1,I\neq J}^{N_{nuclei}} \frac{Z_I Z_J}{|R_I - R_J|}, \tag{15}$$

where V_{xc} is the exchange-correlation potential, E_{xc} the exchange-correlation energy, $\epsilon_i (i = 1, \ldots, N_s)$ the eigenvalues, and R_I and Z_I^{ion} represent position and valence of the I-th atom, respectively.

3.2 Experiments

The initial electron density in our computation is constructed by some combination of the pseudo atomic obitals [11] and the adaptive refinement is done through the following a posteriori error estimators [8]:

$$h_\tau \|\nabla\rho\|_{0,\tau} \ \ \forall \tau \in T^h(\Omega). \tag{16}$$

The mesh should be locally refined so as to meet the multi-resolution requirements (see Sect. 1). We locate father cells on the coarser level next to the finest of the grid hierarchy and carry out the triquadratic interpolation on these father cells. Our hexahedral mesh is well suited for this local interpolation: values of the trilinear finite element solutions on the 27 nodes are employed to determine the 27 coefficients of the required triquadratic Lagrange interpolating functions.

Figures 1, 2 and 3 are schematic figures illustrating the hexahedral discretizations before and after a local refinement around nuclei and the way to do the triquadratic interpolation. Figure 1 shows the standard hexahedral finite element discretizations and there is a nucleus within the dashed cell, for instance. Figure 2 gives the grid after refinement. The dashed subdomain has been divided into eight child cells. Figure 3 emphasizes those 27 nodes for the triquadratic Lagrange interpolation on the father cell.

After this "smoothing" of the eigenfunctions, we construct new eigenvalue approximations by the following Rayleigh quotients:

$$\epsilon_i = \frac{\frac{1}{2}\int_\Omega \left|\nabla u_i^h\right|^2 + \int_\Omega V_{eff}(\rho_{in})(u_i^h)^2}{\int_\Omega (u_i^h)^2}. \tag{17}$$

Consequently, the ground state total energy can be improved by these updated eigenvalues (c.f. (15)).

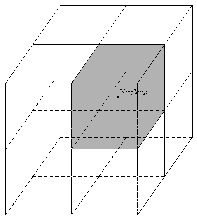


Fig. 1. Standard hexahedral FE discretizations with a nucleus in the dashed cell.

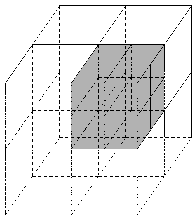


Fig. 2. Hexahedral mesh after local refinement on the dashed cell.

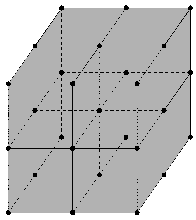


Fig. 3. 27 marked points for triquadratic interpolation on the father cell.

Our computing platform is a Dell Optiplex 755 (Intel Core Duo 2.6 GHz, 4 MB L2 cache, 2 GB memory), provided by the State Key Laboratory of Scientific and Engineering Computing (LSEC) of Chinese Academy of Sciences. Our programs are compiled with "g++ -O3" and run on a single core. The package ARPACK is employed as the eigensolver. The hexahedral grids are visualized by JaVis-1.2.3 developed by HPCC of Institute of Applied Physics and Computational Mathematics.

Benzene

Our computational domain for molecule benzene is $[-20.0, 20.0]^3$ and the adaptive finite element grids are generated on the basis of initial density and the a posteriori error estimators mentioned above. We see that the total energy decreases significantly after interpolation postprocessing. Note that the time of postprocessing is 5 s out of a total time of about 1 min.

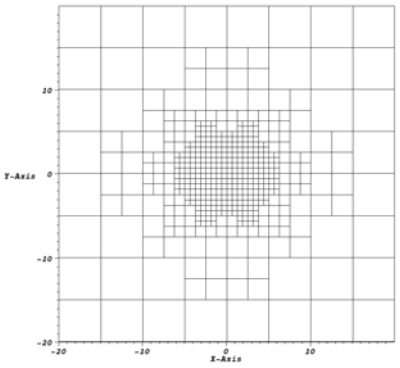


Fig. 4. A coarser mesh of C_6H_6 next to the finest mesh at $z = 0.0$ au.

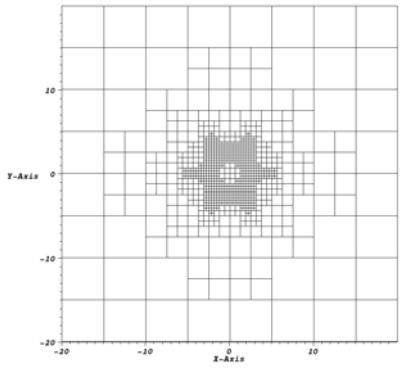


Fig. 5. The finest mesh of C_6H_6 at $z = 0.0$ au.

Table 2. Benzene: interpolation on a part of the father cells.

E_{tot}	Err. w.r.t SIESTA's	E_{tot}^{post}	Err. w.r.t SIESTA's
-37.03 au	1.5%	-37.58 au	0.053%

Table 3. Benzene: total CPU time and time for postprocessing.

Total CPU time	CPU time for postprocessing	Percentage
66.74 s	5.08 s	7.61%

Fullerene

To simulate the molecule C_{60}, we use $[-30.0, 16.0]\times[-23.0, 22.0]\times[-24.0, 21.0]$ as the computational domain. Table 4 shows that, after interpolation postprocessing on the father cells, we obtain a satisfactory approximation of the total energy. Besides, the computational cost is small compared to solving the linear eigenvalue problems. In this example, based on our choice of initial density, we achieve convergence after four self-consistent steps, and the time of postprocessing is 5 min out of a total time of about 80 min.

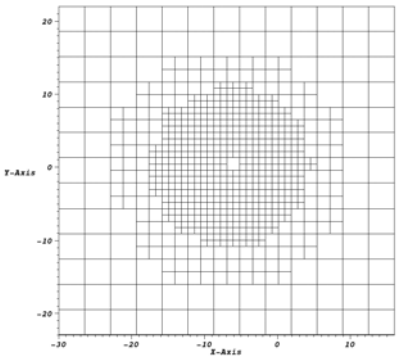


Fig. 6. A coarser mesh of C_{60} next to the finest mesh at $z = 0.0$ au.

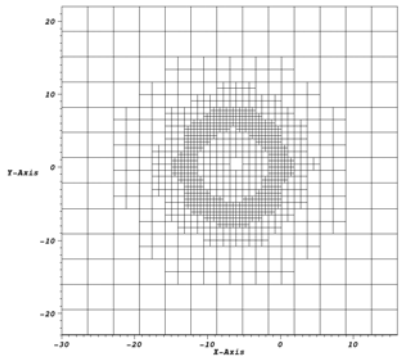


Fig. 7. The finest mesh of C_{60} at $z = 0.0$ au.

Table 4. Fullerene: interpolation on a part of the father cells.

E_{tot}	Err. w.r.t SIESTA's	E_{tot}^{post}	Err. w.r.t SIESTA's
-328.78 au	3.67%	-335.78 au	1.62%

Table 5. Fullerene: total CPU time and time for postprocessing.

Total CPU time	CPU time for postprocessing	Percentage
81 m 7.65 s	5 m 9.67 s	6.37%

4 Concluding Remarks

In this paper, we have proposed an interpolation based local postprocessing approach to adaptive finite element approximations in electronic structure calculations. It is shown by the theoretical analysis for linear eigenvalue problems and particularly successful applications to ground state energy calculations that this is a simple but powerful approach for highly accurate approximations. In our ongoing work, we apply this approach to computations of other quantum quantities of complex molecular systems.

Acknowledgement. This work was partially supported by the National Science Foundation of China under grants 10425105 and 10871198 and the National Basic Research Program under grant 2005CB321704. The authors would like to thank Dr. Xiaoying Dai, Prof. Lihua Shen, Mr. Zhang Yang, and Dr. Dier Zhang for their stimulating discussions and fruitful cooperations on electronic structure computations that have motivated this work.

References

1. R.A. Adams. *Sobolev Spaces*. Acadamic Press, New York, NY, 1975.
2. S. Agmon. *Lectures on the Exponential Decay of Solutions of Second-Order Elliptic Operators*. Princeton University Press, Princeton, NJ, 1981.
3. I. Babuska and J.E. Osborn. Finite element-Galerkin approximation of the eigenvalues and eigenvectors of self-adjoint problems. *Math. Comput.*, 52(186):275–297, 1989.
4. T.L. Beck. Real-space mesh techniques in density-functional theory. *Rev. Mod. Phys.*, 72:1041–1080, 2000.
5. J.R. Brauer. *What Every Engineer Should Know About Finite Element Analysis*. Marcel Decker Inc., New York, NY, 1993.
6. E.L. Briggs, D.J. Sullivan, and J. Bernholc. Real-space multigrid-based approach to large-scale electronic structure calculations. *Phys. Rev. B*, 54:14362–14375, 1996.
7. P.G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland, Amsterdam, 1978.
8. X. Dai. *Adaptive and Localization Based Finite Element Discretizations for the First-Principles Electronic Structure Calculations*. PhD thesis, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, 2008.
9. X. Dai, L. Shen, and A. Zhou. A local computational scheme for higher order finite element eigenvalue approximations. *Int. J Numer. Anal. Model.*, 5:570–589, 2008.
10. X. Dai and A. Zhou. Three-scale finite element discretizations for quantum eigenvalue problems. *SIAM J. Numer. Anal.*, 46:295–324, 2008.
11. X. Gao. *Hexahedral Finite Element Methods for the First-Principles Electronic Structure Calculations*. PhD thesis, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, 2009.

12. X. Gao, F. Liu, and A. Zhou. Three-scale finite element eigenvalue discretizations. *BIT*, 48(3):533–562, 2008.
13. D. Gilbarg and N.S. Trudinger. *Elliptic Partial Differential Equations of Second Order*. Springer, Berlin, Heidelberg, third edition, 2001.
14. S. Goedecker. Linear scaling methods for the solution of Schrödinger's equation. In C. Le Bris, editor, Handbook of Numerical Analysis, volume X of Computational Chemistry. Elsevier, Amsterdam, 2003.
15. X. Gong, L. Shen, D. Zhang, and A. Zhou. Finite element approximations for Schrödinger equations with applications to electronic structure computations. *J. Comput. Math.*, 26: 310–323, 2008.
16. W. Kohn. Density functional and density matrix method scaling linearly with the number of atoms. *Phys. Rev. Lett.*, 76:3168–3171, 1996.
17. W. Kohn and L.J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140(4A):A1133–A1138, 1965.
18. Q. Lin and Y. Yang. Interpolation and correction of finite elements (in Chinese). *Math. Pract. Theory*, (3):29–35, 1991.
19. Q. Lin and Q. Zhu. *The Preprocessing and Postprocessing for the Finite Element Method (in Chinese)*. Shanghai Scientific & Technical Publishers, Shanghai, 1994.
20. R.M. Martin. *Electronic Structure: Basic Theory and Practical Methods*. Cambridge University Press, Cambridge, 2004.
21. J.E. Pask and P.A. Sterne. Finite element methods in *ab initio* electronic structure calculations. *Model. Simul. Mater. Sci. Eng.*, 13:71–96, 2005.
22. P. Pulay. Convergence acceleration of iterative sequences in the case of scf iteration. *Chem. Phys. Lett.*, 73:393–398, 1980.
23. P. Pulay. Improved scf convergence acceleration. *J. Comput. Chem.*, 3:556–560, 1982.
24. A.H. Schatz and L.B. Wahlbin. Interior maximum-norm estimates for finite element methods. *Math. Comput.*, 31:414–442, 1977.
25. A.H. Schatz and L.B. Wahlbin. Interior maximum-norm estimates for finite element methods, Part II. *Math. Comput.*, 64:907–928, 1995.
26. L. Shen and A. Zhou. A defect correction scheme for finite element eigenvalues with applications to quantumn chemistry. *SIAM J. Sci. Comput.*, 28:321–338, 2006.
27. B. Simon. Schrödinger operators in the twentieth century. *J. Math. Phys.*, 41:3523–3555, 2000.
28. E. Tsuchida and M. Tsukada. Electronic-structure calculations based on the finite-element method. *Phys. Rev. B*, 52:5573–5578, 1995.
29. J. Xu and A. Zhou. Local and parallel finite element algorithms based on two-grid discretizations. *Math. Comput.*, 69:881–909, 2000.
30. J. Xu and A. Zhou. A two-grid discretization scheme for eigenvalue problems. *Math. Comput.*, 70:17–25, 2001.
31. J. Xu and A. Zhou. Local and parallel element algorithms for eigenvalue problems. *Acta. Math. Appl. Sin. Engl. Ser.*, 18:185–200, 2002.
32. H. Yserentant. On the regularity of the electronic Schrödinger equation in Hilbert space of mixed derivatives. *Numer. Math.*, 98:731–759, 2004.
33. D. Zhang, L. Shen, A. Zhou, and X. Gong. Finite element method for solving Kohn–Sham equations based on self-adaptive tetrahedral mesh. *Phys. Lett. A*, 372:5071–5076, 2008.

A New a Posteriori Error Estimate for Adaptive Finite Element Methods

Yunqing Huang, Huayi Wei, Wei Yang, and Nianyu Yi

Hunan Key Laboratory for Computation and Simulation in Science and Engineering, School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, Hunan, P.R.China, huangyq@xtu.edu.cn; huayiwei1984@gmail.com; yangweixtu@126.com; yinianyu365109@126.com

1 Introduction

In many scientific problems, adaptive finite element methods has been widely used to improve the accuracy of numerical solutions. The general idea is to refine or adjust the mesh such that the errors are "equally" distributed over the computational mesh, with the aim of achieving a better accurate solution using an optimal number of degrees of freedom. By using the information from the approximated solution and the known data, the a posteriori error estimator provides the information about the size and the distribution of the error of the finite element approximation. There is a large numerical analysis literature on adaptive finite element methods, and various types of a posteriori estimates have been proposed for different problems, see e.g. [1]. The a posterior error estimate and adaptive finite element method were first introduced by [2]. Since the later 1980s, much research work on a posteriori error estimate has been developed including the residual type a posteriori error estimate [8], recovery type a posteriori error estimate [16], a posteriori error estimate based on hierarchic basis [4, 5], and so on. For the literature, the readers are referred to the books [1, 3, 12, 14], the papers [6, 13, 15], and the references cited therein.

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain with Lipschitz boundary $\partial\Omega$. We assume that $\mathcal{T}_h$ is a shape regular triangulation of Ω. Let $V_h \subset H^1(\Omega)$ be the corresponding continuous piecewise linear finite element space associated with $\mathcal{T}_h$, and $u_h \in V_h$ be a finite element approximation to a second order elliptic boundary value problem.

In this paper, we consider the adaptive finite element methods for a second order elliptic boundary value problem. We propose a new a posteriori error estimate which is motivated from the smoothing iteration of the multilevel iterative methods. In particular, on current mesh $\mathcal{T}_h$, we solve the equation to obtain the finite element solution u_h, then global refine the mesh $\mathcal{T}_h$ to obtain the auxiliary mesh $\mathcal{T}_{h/2}$. On the fine mesh, we use a simple smoother such as Gauss–Seidel iteration with u_h as

the initial value. After m iterations, we obtain an approximation solution $u_{h/2,m}$ of finite element solution $u_{h/2}$ on fine mesh $\mathcal{T}_{h/2}$. Then take $\|\nabla(u_h - u_{h/2,m})\|$ as the a posteriori estimate to guide the mesh refinement on $\mathcal{T}_h$. In practice, it only need small number of smoothing steps to obtain an efficient a posteriori error estimator $\|\nabla(u_h - u_{h/2,m})\|$, the computational cost is relatively small.

The rest of the paper is organized as follows: In Sect. 2 we propose the new a posteriori error estimate and investigate its properties. And we describe adaptive finite element algorithm with our new a posteriori error estimator for a second order elliptic boundary value problem. We present some numerical investigations in the efficiency of the new a posteriori error estimate and the performance of the corresponding adaptive finite element algorithm in Sect. 3.

2 A Posteriori Error Estimate

We consider the boundary value problem

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ u = g & \text{on } \partial\Omega, \end{cases} \tag{1}$$

where $\Omega \in \mathbb{R}^2$ is a bounded domain with Lipschitz boundary $\partial\Omega$, for simplicity, Ω is assumed to be a polygonal domain.

In weak form, this problem reads: Find $u \in V = \{v \in H^1(\Omega) : v|_{\partial\Omega} = g\}$ such that

$$a(u, v) = f(v) \quad \forall v \in H_0^1(\Omega), \tag{2}$$

where

$$a(u, v) = \int_\Omega \nabla u \nabla v dx,$$

and

$$f(v) = \int_\Omega f v dx.$$

Let $\mathcal{T}_h$ be a shape regular triangulation of Ω. Consider the C^0 linear finite element space V_h associated with $\mathcal{T}_h$ and defined by

$$V_h = \{v \in H^1(\Omega) : v \in P_1(\tau), \forall \tau \in \mathcal{T}_h\},$$

where $P_l(D)$ denotes the set of all polynomials defined of $D \subseteq \mathbb{R}^2$ of total degree $\leq l$. The discrete approximation to (1) is obtained in the standard way: Find $u_h \in V_h \cap V$ such that

$$a(u_h, v) = f(v) \quad \forall v \in V_h \cap H_0^1(\Omega). \tag{3}$$

Suppose that $\{\psi_i : i = 1, 2, \cdots, N\}$ are the basis for V_h, and define the matrix A^h, and a vector, F^h, via

$$A_{ij}^h := a(\psi_i, \psi_j) \qquad \text{and} \qquad F_i^h := f(\psi_i) \qquad \forall i, j = 1, 2, \cdots, N.$$

Then (3) is equivalent to solving $A^h U = F^h$ with $u_h = \sum_{i=1}^{N} u_i \psi_i$ and $U = (u_i)$. Clearly, the matrix A^h is a symmetric positive definite (SPD) matrix as $a(\cdot, \cdot)$ is SPD.

Let $\mathcal{T}_{h/2}$ be a global refinement of the triangulation $\mathcal{T}_h$ and $V_h \subset V_{h/2}$, suppose $u_h, u_{h/2}$ are then the discrete finite element solutions over $\mathcal{T}_h$ and $\mathcal{T}_{h/2}$, respectively. We have the following orthogonality relation between $u - u_{h/2}$ and $u_h - u_{h/2}$, which follows immediately from the Galerkin orthogonality.

$$\|\nabla(u - u_{h/2})\|_{0,\Omega}^2 = \|\nabla(u - u_h)\|_{0,\Omega}^2 - \|\nabla(u_h - u_{h/2})\|_{0,\Omega}^2. \tag{4}$$

Using the orthogonality (4), we have

$$\begin{aligned} \frac{\|\nabla u_{h/2} - \nabla u_h\|_{0,\Omega}^2}{\|\nabla u - \nabla u_h\|_{0,\Omega}^2} &= \frac{\|\nabla u - \nabla u_h\|_{0,\Omega}^2 - \|\nabla u - \nabla u_{h/2}\|_{0,\Omega}^2}{\|\nabla u - \nabla u_h\|_{0,\Omega}^2} \\ &= 1 - \frac{\|\nabla u - \nabla u_{h/2}\|_{0,\Omega}^2}{\|\nabla u - \nabla u_h\|_{0,\Omega}^2}. \end{aligned}$$

With the saturation assumption:

$$\|\nabla u - \nabla u_{h/2}\|_{0,\Omega} \leq \beta \|\nabla u - \nabla u_h\|_{0,\Omega}, \qquad \beta \in [0, 1),$$

we have

$$\sqrt{1 - \beta^2} \leq \frac{\|\nabla u_{h/2} - \nabla u_h\|_{0,\Omega}}{\|\nabla u - \nabla u_h\|_{0,\Omega}} \leq 1. \tag{5}$$

Numerical examples show that

$$\frac{\|\nabla u_{h/2} - \nabla u_h\|_{0,\Omega}}{\|\nabla u - \nabla u_h\|_{0,\Omega}} \to \frac{\sqrt{3}}{2}. \tag{6}$$

So $\|\nabla(u_{h/2} - u_h)\|_{0,\Omega}$ can be used as a posteriori error estimate if $u_{h/2}$ is at hand. Notice that $u_{h/2} - u_h$ is of high frequency which can be easily obtained by a few smoothing iterations. So we can use the $\|\nabla(u_{h/2,m} - u_h)\|_{0,\Omega}$ instead of $\|\nabla(u_{h/2} - u_h)\|_{0,\Omega}$ after m steps of the a posteriori error estimate, where $u_{h/2,m}$ is an approximation of $u_{h/2}$ by the smoothing iterations, and the computational cost is much cheaper. From (6), it is possible that

$$\frac{\|\nabla u_{h/2,m} - \nabla u_h\|_{0,\Omega}}{\|\nabla u - \nabla u_h\|_{0,\Omega}} \to \frac{\sqrt{3}}{2}. \tag{7}$$

Note that if we have the approximation $u_{h/2,m}$ on $\mathcal{T}_{h/2}$, we then could obtain $I_2 u_{h/2,m}$ by interpolating $u_{h/2,m}$ into the piecewise quadratic finite element spaces on $\mathcal{T}_h$. In Sect. 3, the numerical examples show

$$\frac{\|\nabla I_2 u_{h/2,m} - \nabla u_h\|_{0,\Omega}}{\|\nabla u - \nabla u_h\|_{0,\Omega}} \to 1, \tag{8}$$

it means that the error estimate $\|\nabla I_2 u_{h/2,m} - \nabla u_h\|_{0,\Omega}$ is an asymptotically exact a posteriori error estimate for adaptive finite element methods.

For our error estimator, we find a better approximation $u_{h/2,m}$ in a bigger space, which shares the same principle as the hierarchical basis error estimator of [4]. Comparing with the hierarchal basis error estimator, we obtain the error estimator by solving the problem on the finer mesh, and Bank and Smith solve an approximation problem on the enriched subspace to estimate the error.

We now describe an algorithm to obtain our new a posteriori error estimate for mesh $\mathcal{T}_h$ in detail. Given the finite element solution u_h, the number of smoothing iterations m, we carry out the following steps to obtain the new a posteriori error estimate.

1. Global refine $\mathcal{T}_h$ to obtain an auxiliary fine mesh $\mathcal{T}_{h/2}$.
2. Build the finite element space $V_{h/2}$ on the fine mesh $\mathcal{T}_{h/2}$, and the corresponding stiffness matrix $A^{h/2}$ and load vector $F^{h/2}$.
3. Obtain $I_h^{h/2} u_h$ by interpolating u_h from V_h to $V_{h/2}$, taking $I_h^{h/2} u_h$ as the initial value $u_{h/2,0}$ and solving the linear equations

$$A^{h/2} U = F^{h/2} \tag{9}$$

in m smoothing iterations to obtain $U^m = (u_i^m)$. We then obtain an approximation of $u_{h/2}$

$$u_{h/2,m} = \sum_{i=1}^{N_{h/2}} u_i^m \psi_i,$$

where $N_{h/2}$ is the number of basis function of $V_{h/2}$.

4. For each $\tau \in \mathcal{T}_h$, we calculate

$$\eta_{\tau,m} = \|\nabla(u_h - u_{h/2,m})\|_{0,\tau}$$

as the error estimator on τ, and take

$$\eta_{h,m}^2 = \sum_{\tau \in \mathcal{T}_h} \eta_{\tau,m}^2$$

as the a posteriori error estimate.

For the condition number of the finite element equations on adaptively refined meshes $\{\mathcal{T}_l : l \in \mathcal{N}\}$, a mesh family $\{\mathcal{T}_l : l \in \mathcal{N}\}$ is said to be nondegenerate if there exists a constant $\rho > 0$ such that for all $l \in \mathcal{N}$ and for all $\tau \in \mathcal{T}_l$ there is a ball of radius $\rho \cdot diam(\tau)$ contained in τ, where $diam(\tau)$ denotes the diameter of τ.

Following [7], we assume that the basis $\{\psi_i : i = 1, 2, \cdots, N\}$ of V_h is a local basis:

$$\max_{1 \le i \le N} cardinality\{\tau \in \mathcal{T}_h, supp(\psi_i) \cap \tau \neq \emptyset\} \le C. \tag{10}$$

We have the following estimates:

Lemma 1. *Suppose that the mesh $\mathcal{T}_h$ is nondegenerate. Let A^h denote the matrix corresponding to the inner product $a(\cdot,\cdot)$, i.e., $A^h_{ij} = a(\psi_i, \psi_j)$ where $\{\psi_i : i = 1, 2, \cdots, N\}$ are the standard linear Lagrange basis. Then the maximum eigenvalue λ_{max} of A^h is bounded by*

$$\lambda_{max} \le C. \tag{11}$$

Proof. First note that if we set $v = \sum_{i=1}^{N} v_i \psi_i$ then

$$a(v, v) = V^t A^h V,$$

where $V = (v_i)$, because $a(\cdot,\cdot)$ is bilinear. From the inverse estimate and (10), we have

$$\begin{aligned} a(v, v) &\le C\|v\|_1^2 = C \sum_{\tau\in\mathcal{T}_h} \|v\|_{1,\tau}^2 \le C \sum_{\tau\in\mathcal{T}_h} \|v\|_{0,\infty,\tau}^2 \\ &\le C \sum_{\tau\in\mathcal{T}_h} \sum_{supp(\psi_i)\cap\tau\neq\emptyset} v_i^2 \le CV^tV. \end{aligned}$$

Then we obtain (11).

For solving the linear equations $AU = F$, a basic linear iterative method can be written in the following form:

$$U^{k+1} = U^k + B(F - AU^k), \quad k = 0, 1, 2, \cdots, \tag{12}$$

starting from an initial guess $U^0 \in R^n$.

The Richardson iterative scheme corresponds to (12) with $B = \frac{\omega}{\rho(A)} I$. Namely,

$$U^{k+1} = U^k + \frac{\omega}{\rho(A)}(F - AU^k), \quad k = 0, 1, 2, \cdots. \tag{13}$$

We first discuss its "smoothing property". Set $\omega = 1$ in (13) and define

$$S = I - \frac{1}{\rho(A)} A.$$

Theorem 1. *For the smoother S, we have*

$$\|S^m V\|_A \le Cm^{-1/2}\|V\|_0, \qquad \forall V \in R^n, \tag{14}$$

where $\|V\|_0 = (V, V)^{1/2}$ is the l^2-norm in R^n and

$$\|V\|_A = (AV, V)^{1/2}, \tag{15}$$

is the A-norm corresponding to the linear system we wish to solve.

Proof. Since A is an symmetric positive define matrix, then we have $A\phi_i = \lambda_i\phi_i$ with $\lambda_{min} = \lambda_1 \le \lambda_2 \le \cdots \le \lambda_n = \lambda_{max}$, $(\phi_i, \phi_j) = \delta_{ij}$, and $\forall v \in \mathbb{R}^n$,

$$V = \sum_{i=1}^{n} v_i\phi_i.$$

Then

$$S^m V = \left(I - \frac{1}{\rho(A)}A\right)^m V = \sum_{i=1}^{n}\left(1 - \frac{\lambda_i}{\lambda_{max}}\right)^m v_i\phi_i.$$

And

$$\begin{aligned}\|S^m V\|_A^2 &= \sum_{i=1}^{n}\left(1 - \frac{\lambda_i}{\lambda_{max}}\right)^{2m} v_i^2\lambda_i \\ &= \lambda_{max}\left(\sum_{i=1}^{n}\left(1 - \frac{\lambda_i}{\lambda_{max}}\right)^{2m}\frac{\lambda_i}{\lambda_{max}}v_i^2\right) \\ &\le \lambda_{max}\left\{\sup_{0\le x\le 1}(1-x)^{2m}x\right\}\sum_{i=1}^{n} v_i^2.\end{aligned}$$

Clearly,

$$\sup_{0\le x\le 1}(1-x)^{2m}x \le \frac{1}{2m+1}.$$

From (11), we have

$$\lambda_{max} \le C.$$

Then, from the above inequalities, we obtain

$$\|S^m V\|_A^2 \le Cm^{-1}\|V\|_0^2.$$

On the quasi-uniformly meshes, the smoother operator S have the well known smoothing property

$$\|S^m v_h\|_A \le C\frac{h^{-1}}{m^{1/2}}\|v_h\|_{0,\Omega}, \qquad \forall v_h \in V_h.$$

In the following, from a numerical example, we investigate the smoothing property of Gauss–Seidel smoother on locally refined meshes. We solve the Laplace equation with the exact solution $u = r^{\frac{2}{3}}\sin(\frac{2}{3}\theta), r = \sqrt{x^2+y^2}$ on a L-Shape domain by the adaptive algorithm. We consider one of the adaptive level, we obtain the finite element solution u_h on $\mathcal{T}_h$, then we get $\mathcal{T}_{h/2}$ (see Fig. 1 (Left)) by globally refining $\mathcal{T}_h$. Set u_h as the initial value, and solve the Eq. (16) by executing m smoothing steps on the $\mathcal{T}_{h/2}$, the results are plotted in Fig. 1 (Right), we see that the smoother operator S admits the similar property on the locally refined meshes.

It is obviously that we can obtain an approximation $u_{h/2,m}$ for $u_{h/2}$ at any accuracy with a larger m. And we known that the error between $u_{h/2}$ and $u_{h/2,m}$is reduced quickly at the beginning of several iterative steps, then we need to do only

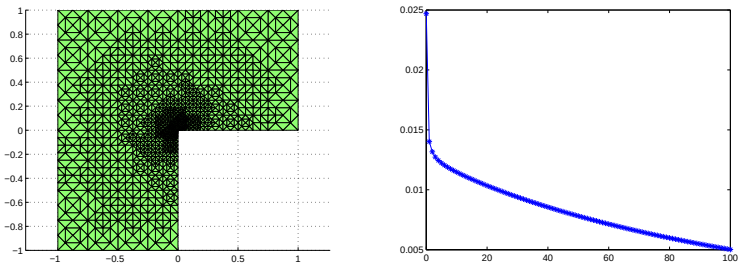


Fig. 1. *Left*: Refined mesh. *Right*: Gauss–Seidel convergence history.

a few smoothing steps to obtain an approximation $u_{h/2,m}$ for our a posteriori error estimator. From our numerical examples in Sect. 3, $m = 3$ performs well.

The standard adaptive finite element methods through local refinement can be written in the following loop

$$\text{SOLVE} \rightarrow \text{ESTIMATE} \rightarrow \text{MARK} \rightarrow \text{REFINE}.$$

Using the above new a posteriori error estimator, the adaptive algorithm has the following general steps:

1. Construct an initial coarse mesh $\mathcal{T}_0$ representing sufficiently well the geometry of the problem. Put $k := 0$.
2. Solve the discrete problem on $\mathcal{T}_k$ to obtain the solution u_k.
3. For each element $\tau \in \mathcal{T}_k$ compute the a posteriori error estimate. In detail, first globally refine $\mathcal{T}_k$ to obtain the fine mesh $\mathcal{T}_k^{'}$, then take u_k as the initial value, use the Gauss–Seidel iteration in m steps, solve the discrete problem on $\mathcal{T}_k^{'}$ to obtain the approximation $u_{k,m}$. Then we get the error estimator $\|\nabla u_k - \nabla u_{k,m}\|_{0,\tau}$ on each $\tau \in \mathcal{T}_k$.
4. If the estimated global error $\|\nabla u_k - \nabla u_{k,m}\|_{o,\Omega}$ is sufficiently small then **stop**. Otherwise, using a suitable marking strategy, decide which elements have to be refined and construct the next mesh $\mathcal{T}_{k+1}$ through local refinement. Replace k by $k + 1$ and return to step 2.

One drawback of hierarchical type error estimators is the computational cost to refine the mesh and assemble the matrix equation on the finer mesh. For our error estimator, in step 3, we can assemble the matrix equation in the finer mesh $\mathcal{T}_{h/2}$ by using the element stiffness matrix in $\mathcal{T}_h$, as the finer mesh $\mathcal{T}_{h/2}$ is the global refinement of $\mathcal{T}_h$, each element are refined into four children elements, the children's element stiffness matrix is the same as its farther's element stiffness matrix for constant coefficients. For smoothing coefficient we can also use the element stiffness matrix on $\mathcal{T}_h$ to assemble the stiffness matrix on $\mathcal{T}_{h/2}$. Then we obtain the a posteriori error estimator at a relatively small computational cost. Thus the adaptive algorithm with our new a posteriori error estimate is efficient and simple in practice. We present

some numerical examples in the following section to investigate in the performance of the adaptive finite element algorithm.

3 Numerical Validation and Applications

In this section, we present some numerical examples to verify the results in Sect. 2 with the model problem

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ u = g & \text{on } \partial\Omega, \end{cases} \tag{16}$$

where $\Omega \in \mathbb{R}^2$ is a bounded domain with Lipschitz boundary $\partial\Omega$.

For a $\tau \in \mathcal{T}_h$,

$$\eta_\tau = \|\nabla u_{h/2} - \nabla u_h\|_{0,\tau}, \quad \text{and} \quad \eta_h = \|\nabla u_{h/2} - \nabla u_h\|_{0,\Omega},$$

the new a posteriori error estimator in τ is

$$\eta_{\tau,m} = \|\nabla u_{h/2,m} - \nabla u_h\|_{0,\tau}, \quad \text{and} \quad \eta_{h,m} = \|\nabla u_{h/2,m} - \nabla u_h\|_{0,\Omega}.$$

To measure the accuracy of η_m, we use the index θ_τ, θ_h defined by

$$\theta_\tau = \frac{\eta_{\tau,m}}{\|\nabla u - \nabla u_h\|_{0,\tau}}, \quad \text{and} \quad \theta_h = \frac{\eta_{h,m}}{\|\nabla u - \nabla u_h\|_{0,\Omega}}.$$

Accordingly, for the error estimator $\eta'_{\tau,m} = \|\nabla I_2 u_{h/2,m} - \nabla u_h\|_{0,\tau}$ and $\eta'_{h,m} = \|\nabla I_2 u_{h/2,m} - \nabla u_h\|_{0,\Omega}$, where $I_2 u_{h/2,m}$ is a piecewise quadratic polynomial which obtained by the interpolation postprocessing. We define

$$\theta'_\tau = \frac{\eta'_{\tau,m}}{\|\nabla u - \nabla u_h\|_{0,\tau}}, \qquad \theta'_h = \frac{\eta'_{h,m}}{\|\nabla u - \nabla u_h\|_{0,\Omega}}.$$

In the following examples, we investigate the performance of our new a posteriori error estimator. In detail, we consider two types of methods for local mesh refinement, one based on Centroidal Voronoi Delaunay Triangulation(CVDT) [10, 11], the other on bisection, 3 Gauss–Seidel iterations are used to obtain the approximation $u_{h/2,m}$, and then $\eta_{h,\tau}$ is used as the error estimator. We implement our numerical tests with the Matlab package iFEM [9].

Example 1 In this example, we solve (16) with $f = 0$ and the exact solution $u = r^{\frac{2}{3}} \sin(\frac{2}{3}\theta), r = \sqrt{x^2 + y^2}$ on the L-Shape domain $\Omega = \{-1 \le x, y \le 1\} \setminus \{0 \le x \le 1, -1 \le y \le 0\}$. The mesh refinement is based on CVDT. The results are shown in Fig. 2. We see that

$$\|\nabla u - \nabla u_h\|_0 = O(N^{-1/2}),,$$

$$\|\nabla u - \nabla I_2 u_{h/2,3}\|_0 = O(N^{-0.7}), \qquad \|\nabla u_{h/2} - \nabla u_{h/2,3}\|_0 = O(N^{-0.67}).$$

For the efficient index, it shows that

$$\theta_h \to \frac{\sqrt{3}}{2}, \qquad \theta_h' \to 1.$$

Notice that the decay of $\|\nabla u - \nabla u_h\|_0$ is quasi-optimal.

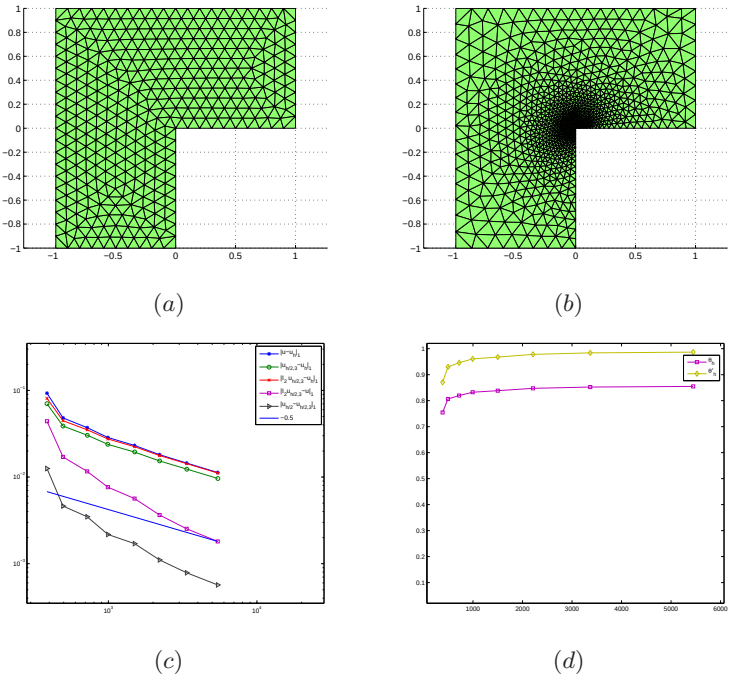


Fig. 2. Results of example 1. (**a**): initial mesh; (**b**): refined mesh after 4 refinements; (**c**): errors; (**d**): effectivity index.

Example 2 In this example, as in Example 1, we solve (16) with the exact solution $u = r^{\frac{2}{3}} \sin(\frac{2}{3}\theta)$ on the L-Shape domain. But, we use the bisection for local mesh refinement. We obtain similar results; Fig. 3 plots the initial mesh and the adaptively refined mesh after 8 adaptive iterations. From Fig. 3, we see that

$$\|\nabla u - \nabla u_h\|_0 = O(N^{-1/2}),$$

$$\|\nabla u - \nabla I_2 u_{h/2,3}\|_0 = O(N^{-0.85}), \qquad \|\nabla u_{h/2} - \nabla u_{h/2,3}\|_0 = O(N^{-3/4}).$$

For the efficient index, it shows that

$$\theta_h \to \frac{\sqrt{3}}{2}, \qquad \theta_h' \to 1.$$

Notice that the decay of $\|\nabla u - \nabla u_h\|_0$ is also quasi-optimal.

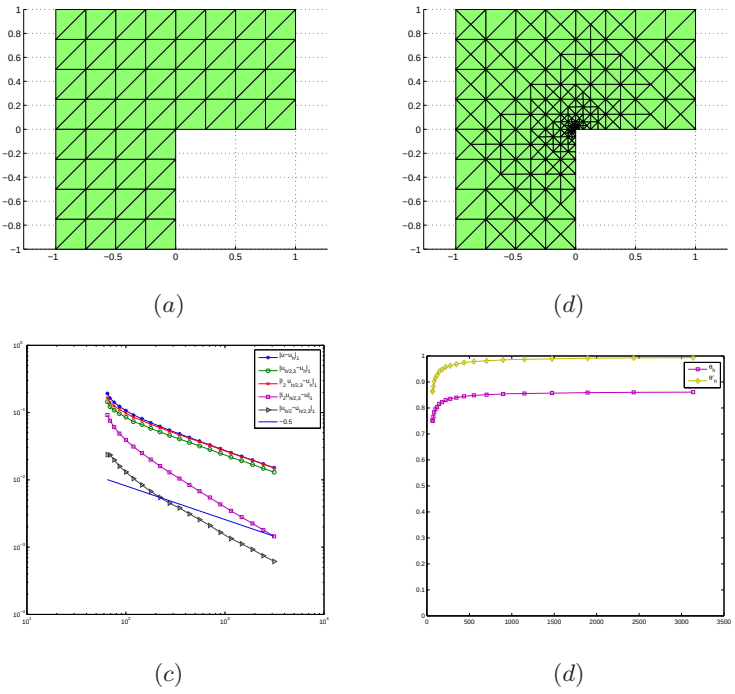


Fig. 3. Results of example 2. (**a**): initial mesh; (**b**): refined mesh after 8 refinements; (**c**): errors; (**d**): effectivity index.

Example 3 In this example, we solve (16) with $f = 1$ and the exact solution $u = \sqrt{\frac{1}{2}(r-x)} - \frac{1}{4}r^2, r = \sqrt{x^2+y^2}$ on a crack domain $\Omega = \{|x| + |y| < 1\} \setminus \{0 \le x \le 1, y = 0\}$. Figure 4 plots the initial mesh and the adaptively refined mesh after 8 adaptive iterations, and shows the performance of the error estimator. We see that

$$\|\nabla u - \nabla u_h\|_0 = O(N^{-1/2}),$$

$$\|\nabla u - \nabla I_2 u_{h/2,3}\|_0 = O(N^{-0.65}), \qquad \|\nabla u_{h/2} - \nabla u_{h/2,3}\|_0 = O(N^{-0.65}).$$

For the efficient index, it shows that

$$\theta_h \to \frac{\sqrt{3}}{2}, \qquad \theta_h' \to 1.$$

The decay of $\|\nabla u - \nabla u_h\|_0$ is also quasi-optimal.

Finally, based on the numerical observation and rough analysis, we may propose a conjecture on the convergence property of the finite element method.

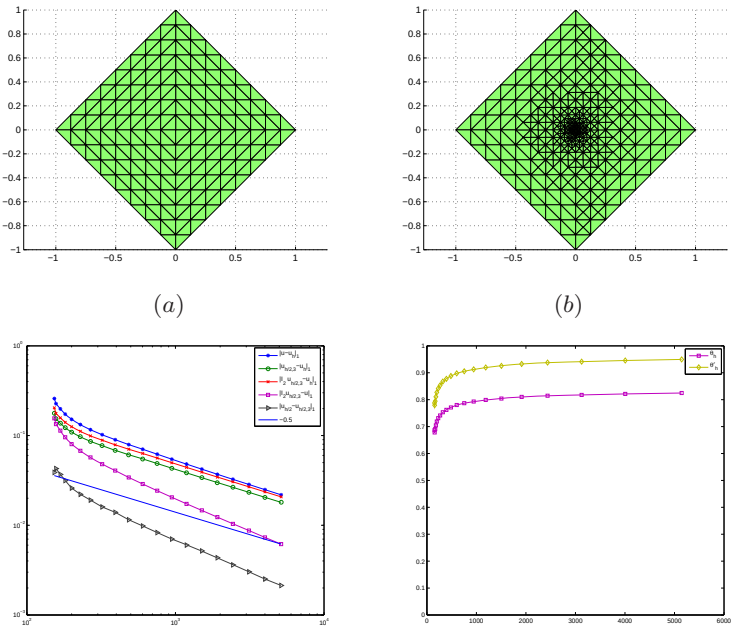


Fig. 4. Results of example 3. (**a**): initial mesh; (**b**): refined mesh after 8 refinements; (**c**): errors; (**d**): efficient index.

Conjecture For linear triangular element approximation on a sequence of triangulations $\mathcal{T}_h$, if the convergence rate is optimal in the sense of

$$\|u - u_h\|_1 \leq CN^{-1/2},$$

where N is the total number of unknowns. Then there holds

$$\frac{\|u_h - u_{h/2}\|_1}{\|u - u_h\|_1} \to \frac{\sqrt{3}}{2}\ (N \to \infty) \quad \text{and} \quad \frac{\|u_h - I_2 u_{h/2}\|_1}{\|u - u_h\|_1} \to 1\ (N \to \infty).$$

References

1. M. Ainsworth and J.T. Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Wiley Interscience, New York, NY, 2000.
2. I. Babuška and W.C. Rheinboldt. A posteriori error estimates for the finite element method. *Int. J. Numer. Methods Eng.*, 12:1597–1615, 1978.
3. I. Babuška and T. Strouboulis. *The Finite Element Method and Its Reliability*. Oxford University Press, Oxford, 2001.

4. R.E. Bank and R.K. Smith. A posteriori estimates based on hierarchical basis. *SIAM J. Numer. Anal.*, 30:921–935, 1993.
5. R.E. Bank. Hierarchical bases and the finite element method. *Acta Numer.*, 5:1–43, 1996.
6. R.E. Bank and J. Xu. Asymptotically exact a posteriori error estimators, part i: grids with superconvergence. *SIAM J. Numer. Anal.*, 41:2294–2312, 2003.
7. R.E. Bank and L.R. Scott. On the conditioning of finite element equations with highly refined meshes. *SIAM J. Numer. Anal.*, 26:1383–1394, 1989.
8. R.E. Bank and A. Weiser. Some a posteriori error estimators for elliptic partial differential equations. *Math. Comput.*, 44:283–301, 1985.
9. L. Chen. *iFEM: An innovative finite element method package in Matlab*. http://math.uci.edu/ chenlong/iFEM.html, 2008.
10. Y.Q. Huang, D.S. Wang H.F. Qin, and Q. Du. Convergent adaptive finite element method based on centroidal Voronoi tessellations and superconvergence. Submitted.
11. Y.Q. Huang, H.F. Qin, and D.S. Wang. Centroidal Voronoi tessellation-based finite element superconvergence. *Int. J. Numer. Methods Eng.*, 76:1819–1839, 2008.
12. R. Verfürth. *A Review of a Posteriori Error Estimation and Adaptive Mesh-refinment Techniques*. Wiley/Teubner, Stuttgart, 1996.
13. J. Xu and Z.M. Zhang. Analysis of recovery type a posteriori error estimators for mildly structured grids. *Math. Comput.*, 73:1139–1152, 2004.
14. N.N. Yan. *Superconvergence Analysis and a Posteriori Error Estimation in Finite Element Methods*. Science Press, Beijing, 2008.
15. J.Z. Zhu and Z.M. Zhang. The relationship of some a posteriori estimators. *Comput. Methods Appl. Mech. Eng.*, 176:463–475, 1999.
16. O.C. Zienkiewicz and J.Z. Zhu. The supercovergent patch recovery and a posteriori error estimates. *Int. J. Numer. Methods Eng.*, 33:1331–1382, 1992.

Space-Time Nonconforming Optimized Schwarz Waveform Relaxation for Heterogeneous Problems and General Geometries

Laurence Halpern[1], Caroline Japhet[2], and Jérémie Szeftel[3]

[1] LAGA, Université Paris XIII, Villetaneuse 93430, France, halpern@math.univ-paris13.fr
[2] LAGA, Université Paris XIII, Villetaneuse 93430, France; CSCAMM, University of Maryland College Park, MD 20742 USA, japhet@cscamm.umd.edu, the first two authors are partially supported by french ANR (COMMA) and GdR MoMaS.
[3] Département de mathématiques et applications, Ecole Normale supérieure, 45 rue d'Ulm, 75230 Paris cedex 05 France, Jeremie.Szeftel@ens.fr

1 Introduction

In many fields of applications it is necessary to couple models with very different spatial and time scales and complex geometries. Amongst them are ocean-atmosphere coupling and far field simulations of underground nuclear waste disposal. For such problems with long time computations, a splitting of the time interval into windows is essential. This allows for robust and fast solvers in each time window, with the possibility of nonconforming space-time grids, general geometries, and ultimately adaptive solvers.

Optimized Schwarz Waveform Relaxation (OSWR) methods were introduced and analyzed for linear advection-reaction-diffusion problems with constant coefficients in [1, 3, 9]. All these methods rely on an algorithm that computes independently in each subdomain over the whole time interval, exchanging space-time boundary data through optimized transmission operators. They can apply to different space-time discretization in subdomains, possibly nonconforming and need a very small number of iterations to converge. Numerical evidences of the performance of the method with variable smooth coefficients were given in [9].

An extension to discontinuous coefficients was introduced in [4], with asymptotically optimized Robin transmission conditions in some particular cases. In [2, 6], semi-discretization in time in one dimension was performed using discontinuous Galerkin, see [8, 10]. In [7], we extended the analysis to the two dimensional case. We obtained convergence results and error estimates for rectangular or strip-subdomains.

For the space discretization, we extended numerically the nonconforming approach in [5] to advection-diffusion problems and optimized order 2 transmission

conditions, to allow for non-matching grids in time and space on the boundary. The space-time projections between subdomains were computed with an optimal projection algorithm without any additional grid, as in [5]. In [7], two dimensional simulations with continuous coefficients were presented.

We present here new results in two directions: we extend the proof of convergence of the OSWR algorithm to nonoverlapping subdomains with curved interfaces. We also present simulations for two subdomains, with piecewise smooth coefficients and a curved interface, for which no error estimates are available. We finally present an application to the porous media equation.

We consider the advection-diffusion-reaction equation,

$$\partial_t u + \nabla \cdot (\boldsymbol{b} u - \nu \nabla u) + cu = f \text{ in } \mathbb{R}^N \times (0,T), \tag{1}$$

with initial condition u_0, and $N = 2$. The advection, diffusion and reaction coefficients $\boldsymbol{b}$, ν and c, are piecewise smooth, we suppose $\nu \geq \nu_0 > 0$ *a.e.*.

2 The Continuous OSWR Algorithm

We consider a decomposition into nonoverlapping subdomains $\Omega_i, i \in \{1,...,I\}$, organized as depicted in Fig. 1. The interfaces between the subdomains are supposed to be flat at infinity. For any $i \in \{1,...,I\}$, $\partial\Omega_i$ is the boundary of Ω_i, $\boldsymbol{n}_i$ the unit exterior normal vector to $\partial\Omega_i$, $\mathcal{N}_i$ is the set of indices of the neighbors of Ω_i. For $j \in \mathcal{N}_i$, $\Gamma_{i,j}$ is the common interface.

Fig. 1. Decomposition in subdomains. *Left*: Robin transmission conditions, *right*: second order transmission conditions.

Following [1, 2, 3, 4], we introduce the boundary operators $\mathcal{S}_{i,j}$ acting on functions defined on $\Gamma_{i,j}$:

$$\mathcal{S}_{i,j}\varphi = p_{i,j}\varphi + q_{i,j}(\partial_t\varphi + \nabla_{\Gamma_{i,j}} \cdot (\boldsymbol{r}_{i,j}\varphi - s_{i,j}\nabla_{\Gamma_{i,j}}\varphi)),$$

with respectively ∇_Γ and $\nabla_\Gamma\cdot$ the gradient and divergence operators on Γ. $p_{i,j}$, $q_{i,j}$, $\boldsymbol{r}_{i,j}$, $s_{i,j}$ are real parameters. $q_{i,j} = 0$, will be referred to as a Robin operator. We introduce the coupled problems

$$\begin{aligned} &\partial_t u_i + \nabla \cdot (\boldsymbol{b}_i u_i - \nu_i \nabla u_i) + c_i u_i = f \text{ in } \Omega_i \times (0,T) \\ &\left(\nu_i \partial_{\boldsymbol{n}_i} - \boldsymbol{b}_i \cdot \boldsymbol{n}_i\right) u_i + \mathcal{S}_{i,j} u_i = \\ &\qquad \left(\nu_j \partial_{\boldsymbol{n}_i} - \boldsymbol{b}_j \cdot \boldsymbol{n}_i\right) u_j + \mathcal{S}_{i,j} u_j \text{ on } \Gamma_{i,j} \times (0,T), j \in \mathcal{N}_i. \end{aligned} \tag{2}$$

As coefficients ν and $\boldsymbol{b}$ are possibly discontinuous on the interface, we note, for $s \in \Gamma_{i,j}$, $\nu_i(s) = \lim_{\varepsilon \to 0} \nu(s - \varepsilon \boldsymbol{n}_i)$. The same notation holds for $\boldsymbol{b}$. Under regularity assumptions, solving (1) is equivalent to solving (2) for $i \in \{1, ..., I\}$ with u_i the restriction of u to Ω_i. We now introduce an algorithm to solve (2). An initial guess $(g_{i,j})$ is given in $L^2((0,T) \times \Gamma_{i,j})$ for $i \in \{1, ..., I\}, j \in \mathcal{N}_i$. We solve iteratively

$$\begin{aligned} &\partial_t u_i^k + \nabla \cdot (\boldsymbol{b}_i u_i^k - \nu_i \nabla u_i^k) + c_i u_i^k = f \text{ in } \Omega_i \times (0,T), \\ &\left(\nu_i \partial_{\boldsymbol{n}_i} - \boldsymbol{b}_i \cdot \boldsymbol{n}_i\right) u_i^k + \mathcal{S}_{i,j} u_i^k = \\ &\qquad \left(\nu_j \partial_{\boldsymbol{n}_i} - \boldsymbol{b}_j \cdot \boldsymbol{n}_i\right) u_j^{k-1} + \mathcal{S}_{i,j} u_j^{k-1} \text{ on } \Gamma_{i,j} \times (0,T), j \in \mathcal{N}_i. \end{aligned} \tag{3}$$

with the convention $\left(\nu_i \partial_{\boldsymbol{n}_i} - \boldsymbol{b}_i \cdot \boldsymbol{n}_i\right) u_i^1 + \mathcal{S}_{i,j} u_i^1 = g_{i,j},\ j \in \mathcal{N}_i$.

Theorem 1. *Assume $\boldsymbol{b}_i \in (W^{1,\infty}(\Omega_i))^N$, $\nu_i \in W^{1,\infty}(\Omega_i)$, $p_{i,j} \in W^{1,\infty}(\Gamma_{i,j})$ with $p_{i,j} > 0$ a.e.. If $q_{i,j} = 0$, or if $q_{i,j} = q > 0$ with $\boldsymbol{r}_{i,j} \in (W^{1,\infty}(\Gamma_{i,j}))^{N-1}$, $\boldsymbol{r}_{i,j} = \boldsymbol{r}_{j,i}$ on $\Gamma_{i,j}$, $s_{i,j} \in W^{1,\infty}(\Gamma_{i,j})$, $s_{i,j} > 0$, $s_{i,j} = s_{j,i}$ on $\Gamma_{i,j}$, the algorithm (3) converges in each subdomain to the solution of problem (2).*

Proof. We first need some results in differential geometry. For every $j \in \mathcal{N}_i$, the normal vector $\boldsymbol{n}_i$ can be extended in a neighbourhood of $\Gamma_{i,j}$ as a smooth function $\tilde{\boldsymbol{n}}_i$ with length one. Let $\psi_{i,j} \in \mathcal{C}^\infty(\overline{\Omega_i})$, such that $\psi_{i,j} \equiv 1$ in a neighbourhood of $\Gamma_{i,j}$, $\psi_{i,j} \equiv 0$ in a neighbourhood of $\Gamma_{i,k}$ for $k \in \mathcal{N}_i$, $k \neq j$ and $\sum_{j \in \mathcal{N}_i} \psi_{i,j} > 0$ on Ω_i. Let $\tilde{\boldsymbol{n}}_i$ be defined on a neighbourhood of the support of $\psi_{i,j}$. We can extend the tangential gradient and divergence operators in the support of $\psi_{i,j}$ by:

$$\widetilde{\nabla}_{\Gamma_{i,j}} \varphi := \nabla \varphi - (\partial_{\tilde{\boldsymbol{n}}_i} \varphi) \tilde{\boldsymbol{n}}_i, \quad \widetilde{\nabla}_{\Gamma_{i,j}} \cdot \boldsymbol{\varphi} := \nabla \cdot (\boldsymbol{\varphi} - (\boldsymbol{\varphi} \cdot \tilde{\boldsymbol{n}}_i) \tilde{\boldsymbol{n}}_i).$$

It is easy to see that $(\widetilde{\nabla}_{\Gamma_{i,j}} \varphi)_{|\Gamma_{i,j}} = \nabla_{\Gamma_{i,j}} \varphi$, $(\widetilde{\nabla}_{\Gamma_{i,j}} \cdot \boldsymbol{\varphi})_{|\Gamma_{i,j}} = \nabla_{\Gamma_{i,j}} \cdot \boldsymbol{\varphi}$ and for $\boldsymbol{\varphi}$ and χ with support in $supp(\psi_{i,j})$, we have

$$\int_{\Omega_i} (\widetilde{\nabla}_{\Gamma_{i,j}} \cdot \boldsymbol{\varphi})\, \chi \, dx = - \int_{\Omega_i} \boldsymbol{\varphi} \cdot \widetilde{\nabla}_{\Gamma_{i,j}} \chi \, dx. \tag{4}$$

Now we prove Theorem 1. The key point is to obtain energy estimates for the homogeneous problem (2), i.e. for $f = u_0 = 0$. We sketch the proof in the most difficult case $q_{i,j} = q > 0$. For the geometry, we consider the case depicted in the right part of Fig. 1. In that case Ω_i has at most two neighbours with

We set $\|\varphi\|_i = \|\varphi\|_{L^2(\Omega_i)}$, $|||\varphi|||_i^2 = \|\sqrt{\nu_i}\, \nabla \varphi\|_{L^2(\Omega_i)}^2$, $\|\varphi\|_{i,\infty} = \|\varphi\|_{L^\infty(\Omega_i)}$, $\|\varphi\|_{i,1,\infty} = \|\varphi\|_{W^{1,\infty}(\Omega_i)}$ and $\beta_i = \sum_{j \in \mathcal{N}_i} \psi_{i,j} \beta_{i,j}$ with $\beta_{i,j} = \sqrt{\frac{p_{i,j} + p_{j,i}}{2}}$.

1. We multiply the first equation of (3) by $\beta_i^2\, u_i^k$, integrate on $\Omega_i \times (0,t)$ then integrate by parts in space,

$$\frac{1}{2}\|\beta_i u_i^k(t)\|_i^2 + \int_0^t |||\beta_i u_i^k(\tau,\cdot)|||_i^2 \, d\tau - \int_0^t \int_{\Omega_i} \beta_i(\boldsymbol{b}_i \cdot \nabla\beta_i)(u_i^k)^2 \, dx\, d\tau$$
$$+ \int_0^t \int_{\Omega_i} (c_i + \frac{1}{2}\nabla\cdot\boldsymbol{b}_i)\beta_i^2 (u_i^k)^2 \, dx\, d\tau - \int_0^t \int_{\Omega_i} \nu_i |\nabla\beta_i|^2 (u_i^k)^2 \, dx\, d\tau$$
$$- \int_0^t \int_{\Gamma_{i,j}} \beta_{i,j}^2 (\nu_i \partial_{\boldsymbol{n}_i} u_i^k - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2} u_i^k)\, u_i^k \, d\sigma\, d\tau = 0. \quad (5)$$

2. We multiply the first equation of (3) by $\partial_t u_i^k$, integrate on $\Omega_i \times (0,t)$ and integrate by parts in space,

$$\int_0^t \|\partial_t u_i^k\|_i^2 \, d\tau + \frac{1}{2} |||\, u_i^k(t)\,|||_i^2 + \int_0^t \int_{\Omega_i} (c_i u_i^k + \nabla\cdot(\boldsymbol{b}_i u_i^k))\, \partial_t u_i^k \, dx\, d\tau$$
$$- \int_0^t \int_{\Gamma_{i,j}} \nu_i \partial_{\boldsymbol{n}_i} u_i^k \, \partial_t u_i^k \, d\sigma\, d\tau = 0. \quad (6)$$

3. We multiply the first equation of (3) by $\widetilde{\nabla}_{\Gamma_{i,j}}\cdot(\psi_{i,j}^2 \boldsymbol{r}_{i,j} u_i^k)$ integrate on $\Omega_i\times(0,t)$ integrate by parts in space to obtain

$$\int_0^t \int_{\Omega_i} \partial_t u_i^k \widetilde{\nabla}_{\Gamma_{i,j}}\cdot(\psi_{i,j}^2 \boldsymbol{r}_{i,j} u_i^k)\, dx\, d\tau + \int_0^t \int_{\Omega_i} \nabla\cdot(\boldsymbol{b}_i u_i^k)\, \widetilde{\nabla}_{\Gamma_{i,j}}\cdot(\psi_{i,j}^2 \boldsymbol{r}_{i,j} u_i^k)\, dx\, d\tau$$
$$+ \int_0^t \int_{\Omega_i} c_i u_i^k \widetilde{\nabla}_{\Gamma_{i,j}}\cdot(\psi_{i,j}^2 \boldsymbol{r}_{i,j} u_i^k)\, dx\, d\tau - \int_0^t \int_{\Gamma_{i,j}} \nu_i \partial_{\boldsymbol{n}_i} u_i^k \, \nabla_{\Gamma_{i,j}}\cdot(\boldsymbol{r}_{i,j} u_i^k)\, d\sigma\, d\tau$$
$$- \frac{1}{4}\int_0^t \|\psi_{i,j}\sqrt{\nu_i\, s_{i,j}}\, \nabla\widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k\|_i^2 \, d\tau \le C \int_0^t (\|\sqrt{\nu_i}\nabla u_i^k\|_i^2 + \|\beta_i u_i^k\|_i^2)\, d\tau. \quad (7)$$

4. We multiply the first equation of (3) by $-\widetilde{\nabla}_{\Gamma_{i,j}}\cdot(\psi_{i,j}^2 s_{i,j}\, \widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k)$ integrate on $\Omega_i\times(0,t)$, integrate by parts in space using (4). Using that

$$-\int_0^t \int_{\Omega_i} \nu_i \nabla u_i^k \cdot \nabla(\widetilde{\nabla}_{\Gamma_{i,j}}\cdot(\psi_{i,j}^2 s_{i,j}\, \widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k))\, dx\, d\tau$$
$$\ge \frac{1}{2}\int_0^t \|\psi_{i,j}\sqrt{\nu_i\, s_{i,j}}\, \nabla\widetilde{\nabla}_{\Gamma_{i,j}}, u_i^k)\|_i^2 \, d\tau - C\int_0^t \|\sqrt{\nu_i}\nabla u_i^k\|_i^2 \, d\tau,$$

we obtain

$$\frac{1}{2}\|\psi_{i,j}\sqrt{s_{i,j}}\, \widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k(t)\|_i^2 + \frac{1}{2}\int_0^t \|\psi_{i,j}\sqrt{\nu_i\, s_{i,j}}\, \nabla\widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k)\|_i^2 \, d\tau$$
$$+ \int_0^t \int_{\Omega_i} \psi_{i,j}^2 s_{i,j}\, c_i |\widetilde{\nabla}_{\Gamma_{i,j}} u_i^k|^2 \, dx\, d\tau + \int_0^t \int_{\Gamma_{i,j}} \nu_i \partial_{\boldsymbol{n}_i} u_i^k \, \nabla_{\Gamma_{i,j}}\cdot(s_{i,j}\, \nabla_{\Gamma_{i,j}}\, u_i^k)\, d\sigma\, d\tau$$
$$\le \int_0^t \int_{\Omega_i} \nabla\cdot(\boldsymbol{b}_i u_i^k)\, \widetilde{\nabla}_{\Gamma_{i,j}}\cdot(\psi_{i,j}^2 s_{i,j}\, \widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k)\, dx\, d\tau + C\int_0^t \|\sqrt{\nu_i}\nabla u_i^k\|_i^2 \, d\tau. \quad (8)$$

We add (6), (7) and (8), multiply the result by q, and add it to (5). We use $ab \le \frac{a^2}{2\varepsilon} + \frac{\varepsilon}{2}b^2$ in the integral terms in the right-hand side, simplify with the left-hand side, and obtain

$$\frac{1}{2}\left(\|\beta_i u_i^k(t)\|_i^2 + q\,|||\, u_i^k(t)\,|||_i^2 + q\|\psi_{i,j}\sqrt{s_{i,j}}\,\widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k(t)\|_i^2\right) + \int_0^t |||\beta_i u_i^k(\tau,\cdot)|||_i^2\, d\tau$$
$$+ \frac{q}{2}\int_0^t \|\partial_t u_i^k\|_i^2\, d\tau + \frac{q}{8}\int_0^t \|\psi_{i,j}\sqrt{\nu_i\, s_{i,j}}\,\nabla\widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k\|_i^2\, d\tau$$
$$- q\int_0^t\int_{\Gamma_{i,j}} \nu_i\partial_{\boldsymbol{n}_i} u_i^k\,\left(\partial_t u_i^k + \nabla_{\Gamma_{i,j}}\cdot(\boldsymbol{r}_{i,j}u_i^k) - \nabla_{\Gamma_{i,j}}\cdot(s_{i,j}\,\nabla_{\Gamma_{i,j}}\, u_i^k)\right)\, d\sigma\, d\tau$$
$$- \int_0^t\int_{\Gamma_{i,j}} \beta_{i,j}^2(\nu_i\partial_{\boldsymbol{n}_i}u_i^k - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}u_i^k)\, u_i^k\, d\sigma\, d\tau \le \frac{q}{2}(\|\boldsymbol{b}_i\|_{i,1,\infty} + \|c_i\|_{i,\infty})\|u_i^k(t)\|_i^2$$
$$+ C\left(\int_0^t \|\beta_i u_i^k\|_i^2\, d\tau + q\int_0^t \|\sqrt{\nu_i}\nabla u_i^k\|_i^2\, d\tau\right). \quad (9)$$

Recalling that $s_{i,j} = s_{j,i}$ on $\Gamma_{i,j}$ and $\boldsymbol{r}_{i,j} = \boldsymbol{r}_{j,i}$ on $\Gamma_{i,j}$, we use now the identity:

$$\left(\nu_i\partial_{\boldsymbol{n}_i}u_i^k - \boldsymbol{b}_i\cdot\boldsymbol{n}_i u_i^k + \mathcal{S}_{i,j}u_i^k\right)^2 - \left(\nu_i\partial_{\boldsymbol{n}_i}u_i^k - \boldsymbol{b}_i\cdot\boldsymbol{n}_i u_i^k - \mathcal{S}_{j,i}u_i^k\right)^2$$
$$= 4\big(\beta_{i,j}^2(\nu_i\partial_{\boldsymbol{n}_i}u_i^k - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}u_i^k)u_i^k + q\nu_i\partial_{\boldsymbol{n}_i}u_i^k(\partial_t u_i^k + \nabla_{\Gamma_{i,j}}\cdot(\boldsymbol{r}_{i,j}u_i^k))\big)$$
$$- 4\nabla_{\Gamma_{i,j}}\cdot(s_{i,j}\,\nabla_{\Gamma_{i,j}}\, u_i^k) + 2q(p_{i,j} - p_{j,i} - 2\boldsymbol{b}_i\cdot\boldsymbol{n}_i)(\partial_t u_i^k + \nabla_{\Gamma_{i,j}}\cdot(\boldsymbol{r}_{i,j}u_i^k)$$
$$- \nabla_{\Gamma_{i,j}}\cdot(s_{i,j}\,\nabla_{\Gamma_{i,j}}\, u_i^k))u_i^k + (p_{i,j} + p_{j,i})(p_{i,j} - p_{j,i} - \boldsymbol{b}_i\cdot\boldsymbol{n}_i)(u_i^k)^2. \quad (10)$$

Replacing (10) into (9), we obtain

$$\frac{1}{2}\left(\|\beta_i u_i^k(t)\|_i^2 + q\,|||\, u_i^k(t)\,|||_i^2 + q\|\psi_{i,j}\sqrt{s_{i,j}}\,\widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k(t)\|_i^2\right) + \int_0^t |||\beta_i u_i^k(\tau,\cdot)|||_i^2\, d\tau$$
$$+ \frac{q}{2}\int_0^t \|\partial_t u_i^k\|_i^2\, d\tau + \frac{1}{4}\int_0^t\int_{\Gamma_{i,j}} \left(\nu_i\partial_{\boldsymbol{n}_i}u_i^k - \boldsymbol{b}_i\cdot\boldsymbol{n}_i\, u_i^k - \mathcal{S}_{j,i}u_i^k\right)^2\, d\sigma\, d\tau$$
$$+ \frac{q}{8}\int_0^t \|\psi_{i,j}\sqrt{\nu_i\, s_{i,j}}\,\nabla\widetilde{\nabla}_{\Gamma_{i,j}}\, u_i^k\|_i^2\, d\tau \le \frac{1}{4}\int_0^t\int_{\Gamma_{i,j}} \left(\nu_i\partial_{\boldsymbol{n}_i}u_i^k - \boldsymbol{b}_i\cdot\boldsymbol{n}_i\, u_i^k + \mathcal{S}_{i,j}u_i^k\right)^2\, d\sigma\, d\tau$$
$$+ \int_0^t\int_{\Gamma_{i,j}} (p_{i,j}+p_{j,i})(-p_{i,j}+p_{j,i}+\boldsymbol{b}_i\cdot\boldsymbol{n}_i)(u_i^k)^2\, d\sigma\, d\tau + \frac{q}{2}(\|\boldsymbol{b}_i\|_{i,1,\infty}+\|c_i\|_{i,\infty})\|u_i^k(t)\|_i^2$$
$$+ \frac{q}{2}\int_0^t\int_{\Gamma_{i,j}} (-p_{i,j}+p_{j,i}+2\boldsymbol{b}_i\cdot\boldsymbol{n}_i)(\partial_t u_i^k + \nabla_{\Gamma_{i,j}}\cdot(\boldsymbol{r}_{i,j}u_i^k) - \nabla_{\Gamma_{i,j}}\cdot(s_{i,j}\,\nabla_{\Gamma_{i,j}}\, u_i^k))\, u_i^k\, d\sigma\, d\tau$$
$$+ C\left(\int_0^t \|\beta_i u_i^k\|_i^2\, d\tau + q\int_0^t \|\sqrt{\nu_i}\nabla u_i^k\|_i^2\, d\tau\right). \quad (11)$$

In order to estimate the fourth term in the right-hand side of (11), we observe that

$$\int_0^t\int_{\Gamma_{i,j}} (-p_{i,j}+p_{j,i}+2\boldsymbol{b}_i\cdot\boldsymbol{n}_i)u_i^k\,\partial_t u_i^k\, d\sigma\, d\tau = \frac{1}{2}\int_{\Gamma_{i,j}} (-p_{i,j}+p_{j,i}+2\boldsymbol{b}_i\cdot\boldsymbol{n}_i)u_i^k(t)^2\, d\sigma.$$

By the trace theorem in the right-hand side, we write:

$$\int_0^t\int_{\Gamma_{i,j}} (-p_{i,j} + p_{j,i} + 2\boldsymbol{b}_i\cdot\boldsymbol{n}_i)u_i^k\,\partial_t u_i^k\, d\sigma\, d\tau \le C\|u_i^k(t)\|_i\|\sqrt{\nu_i}\nabla u_i^k(t)\|_i,$$

and

$$\|u_i^k(t)\|_i^2 = 2\int_0^t \int_{\Omega_i} (\partial_t u_i^k) u_i^k \le 2\left(\int_0^t \|\partial_t u_i^k\|_i^2\right)^{\frac{1}{2}} \left(\int_0^t \|u_i^k\|_i^2\right)^{\frac{1}{2}}, \quad (12)$$

we obtain

$$\frac{q}{2}\int_0^t \int_{\Gamma_{i,j}} (-p_{i,j} + p_{j,i} + 2\boldsymbol{b}_i \cdot \boldsymbol{n}_i) u_i^k \, \partial_t u_i^k \, d\sigma \, d\tau \le \frac{q}{8}\int_0^t \|\partial_t u_i^k\|_i^2 \, d\tau + \frac{q}{4} \|\!| u_i^k(t) |\!\|_i^2 + C\left(\int_0^t \|\beta_i u_i^k\|_i^2 \, d\tau\right). \quad (13)$$

Moreover, integrating by parts and using the trace theorem, we have:

$$-\frac{q}{2}\int_0^t \int_{\Gamma_{i,j}} \nabla_{\Gamma_{i,j}} \cdot (s_{i,j} \nabla_{\Gamma_{i,j}} u_i^k)(-p_{i,j} + p_{j,i} + 2\boldsymbol{b}_i \cdot \boldsymbol{n}_i) u_i^k \, d\sigma \, d\tau \le \frac{q}{16}\int_0^t \|\psi_{i,j}\sqrt{\nu_i \, s_{i,j}} \, \nabla \widetilde{\nabla}_{\Gamma_{i,j}} u_i^k\|_i^2 \, d\tau + C\Big(\int_0^t \|\widetilde{\nabla}_{\Gamma_{i,j}} u_i^k\|_i^2 \, d\tau + \int_0^t \|\beta_i u_i^k\|_i^2 \, d\tau\Big). \quad (14)$$

Using (12), we estimate the third term in the right-hand side of (11) by:

$$\frac{q}{2}(\|\boldsymbol{b}_i\|_{i,1,\infty} + \|c_i\|_{i,\infty})\|u_i^k(t)\|_i^2 \le \frac{q}{8}\int_0^t \|\partial_t u_i^k\|_i^2 \, d\tau + C\int_0^t \|\beta_i u_i^k\|_i^2 \, d\tau. \quad (15)$$

Replacing (14), (13) and (15) in (11), then using the transmission conditions, we have:

$$\begin{aligned}
&\frac{1}{2}\left(\|\beta_i u_i^k(t)\|_i^2 + \frac{q}{2} \|\!| u_i^k(t) |\!\|_i^2 + q\|\psi_{i,j}\sqrt{s_{i,j}}\, \widetilde{\nabla}_{\Gamma_{i,j}} u_i^k(t)\|_i^2\right) \\
&+ \int_0^t \|\!|\beta_i u_i^k(\tau,\cdot)|\!\|_i^2 \, d\tau + \frac{q}{4}\int_0^t \|\partial_t u_i^k\|_i^2 \, d\tau + \frac{q}{16}\int_0^t \|\psi_{i,j}\sqrt{\nu_i \, s_{i,j}} \, \nabla \widetilde{\nabla}_{\Gamma_{i,j}} u_i^k\|_i^2 \, d\tau \\
&\qquad + \frac{1}{4}\int_0^t \int_{\Gamma_{i,j}} \left(\nu_i \partial_{\boldsymbol{n}_i} u_i^k - \boldsymbol{b}_i \cdot \boldsymbol{n}_i \, u_i^k - \mathcal{S}_{j,i} u_i^k\right)^2 d\sigma \, d\tau \\
&\qquad \le \frac{1}{4}\int_0^t \int_{\Gamma_{i,j}} \left(\nu_j \partial_{\boldsymbol{n}_i} u_j^{k-1} - \boldsymbol{b}_j \cdot \boldsymbol{n}_i \, u_j^{k-1} + \mathcal{S}_{i,j} u_j^{k-1}\right)^2 d\sigma \, d\tau \\
&\qquad\qquad + C\left(\int_0^t \|\beta_i u_i^k\|_i^2 \, d\tau + \frac{q}{2}\int_0^t \|\sqrt{\nu_i} \, \nabla u_i^k\|_i^2 \, d\tau\right).
\end{aligned}$$

We now sum up over the interfaces $j \in \mathcal{N}_i$, then over the subdomains for $1 \le i \le I$, and over the iterations for $1 \le k \le K$, the boundary terms cancel out, and with $\alpha(t) = \frac{1}{4}\sum_{i\in\{1,\dots,I\}} \sum_{j\in\mathcal{N}_i} \int_0^t \int_{\Gamma_{i,j}} \left(\nu_j \partial_{\boldsymbol{n}_i} u_i^0 - \boldsymbol{b}_j \cdot \boldsymbol{n}_i \, u_j^0 + \mathcal{S}_{i,j} u_j^0\right)^2 d\sigma \, d\tau$, we obtain for any $t \in (0, T)$,

$$\begin{aligned}
&\sum_{k\in\{1,\dots,K\}} \sum_{i\in\{1,\dots,I\}} \left(\|\beta_i u_i^k(t)\|_i^2 + \frac{q}{2}\|\sqrt{\nu_i}\nabla u_i^k(t)\|_i^2 + \nu_0 \int_0^t \|\nabla(\beta_i u_i^k)\|_i^2 \, d\tau\right) \\
&\le \alpha(t) + C \sum_{k\in\{1,\dots,K\}} \sum_{i\in\{1,\dots,I\}} \left(\int_0^t \|\beta_i u_i^k\|_i^2 \, d\tau + \frac{q}{2}\int_0^t \|\sqrt{\nu_i}\nabla u_i^k\|_i^2 \, d\tau\right).
\end{aligned}$$

We conclude with Gronwall's lemma that the sequence converges in $L^2(0,T;H^1(\Omega_i))$.

3 Numerical Results

We recall the discrete time nonconforming Schwarz waveform relaxation method developed in [7].

Let $\mathcal{T}_i$ be the time partition in subdomain Ω_i, with N_i+1 intervals I_n^i, and time step k_n^i. We define interpolation operators $\mathcal{I}^i$ and projection operators $\mathcal{P}^i$ in each subdomain as in [7], and we solve

$$\begin{aligned}
&\partial_t(\mathcal{I}^i U_i^k) + \nabla\cdot(\boldsymbol{b}U_i^k - \nu_i\nabla U_i^k) + c_i\,U_i^k = \mathcal{P}^i f \text{ in } \Omega_i\times(0,T),\\
&\big(\nu_i\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\big)\,U_i^k + S_{i,j}U_i^k =\\
&\qquad \mathcal{P}^i\big((\nu_j\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_j\cdot\boldsymbol{n}_i}{2})\,U_j^{k-1} + \widetilde{S}_{i,j}U_j^{k-1}\big) \text{ on } \Gamma_{i,j}\times(0,T),
\end{aligned}$$

with $S_{i,j}\,U = p_{i,j}\,U + q_{i,j}\,(\partial_t(\mathcal{I}^iU) + \nabla_{\Gamma_{i,j}}\cdot(\boldsymbol{r}_{i,j}U - s_{i,j}\nabla_{\Gamma_{i,j}}U))$, and $\widetilde{S}_{i,j}\,U = p_{i,j}\,U + q_{i,j}\,(\partial_t(\mathcal{I}^jU) + \nabla_{\Gamma_{i,j}}\cdot(\boldsymbol{r}_{i,j}U - s_{i,j}\nabla_{\Gamma_{i,j}}U))$.

The coefficients $p_{i,j}$ and $q_{i,j}$ are defined through an optimization procedure, see [1], restricted to values such that the subdomain problems are well-posed. The time semi-discrete analysis was performed in [7] in the case $\nabla\cdot\boldsymbol{b} = 0$. For the space discretization, we use the nonconforming approach in [5] extended to problem (1) and order 2 transmission conditions, to allow non-matching grids in time and space on the boundary. We have implemented the algorithm with $\mathbf{P}_1$ finite elements in space in each subdomain. Time windows are used in order to reduce the number of iterations of the algorithm. To reduce the number of parameters and following [1], we choose $\boldsymbol{r}_{i,j} = \boldsymbol{\Pi}_{\Gamma_{i,j}}\boldsymbol{b}_j$ with $\boldsymbol{\Pi}_{\Gamma_{i,j}}$ the tangential trace on $\Gamma_{i,j}$, and $s_{i,j} = \nu_j$ (even though the present analysis does not cover this case). The optimized parameters are constant along the interface. They correspond to a mean value of the parameters obtained by a numerical optimization of the convergence factor.

We first give an example of a multidomain solution with one time window. The physical domain is $\Omega = (0,1)\times(0,2)$, the final time is $T = 1$. The initial value is $u_0 = 0.25e^{-100((x-0.55)^2+(y-1.7)^2)}$ and the right-hand side is $f = 0$. The domain Ω is split into two subdomains $\Omega_1 = (0,0.5)\times(0,2)$ and $\Omega_2 = (0.5,1)\times(0,2)$. The reaction factor c is zero, the advection and diffusion coefficients are $\boldsymbol{b}_1 = (0,-1)$, $\nu_1 = 0.001\sqrt{y}$, and $\boldsymbol{b}_2 = (-0.1,0)$, $\nu_2 = 0.1\sin(xy)$. The mesh size over the interface and time step in Ω_1 are $h_1 = 1/32$ and $k_1 = 1/128$, while in Ω_2, $h_2 = 1/24$ and $k_2 = 1/94$. On Fig. 2, we observe, at final time $T = 1$, a very good behavior of the multidomain solution after 5 iterations. The relative error with the one domain solution is of the same order as the error of the scheme.

We analyze now the precision in time. The space mesh is conforming and the converged solution is such that the residual is smaller than 10^{-8}. We compute a variational reference solution on a time grid with 4,096 time steps. The nonconforming solutions are interpolated on the previous grid to compute the error. We start with a time grid with 128 time steps for the left domain and 94 time steps for the right domain. Thereafter the time steps are divided by 2 several times. Figure 3 (left) shows

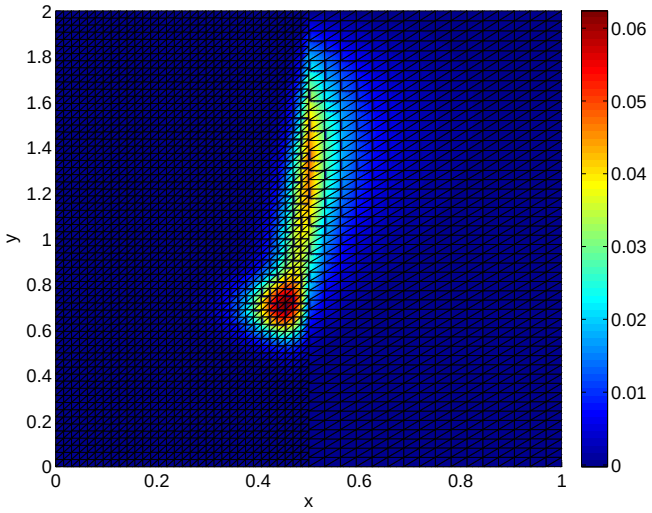


Fig. 2. Nonconforming DG-OSWR solution after 5 iterations.

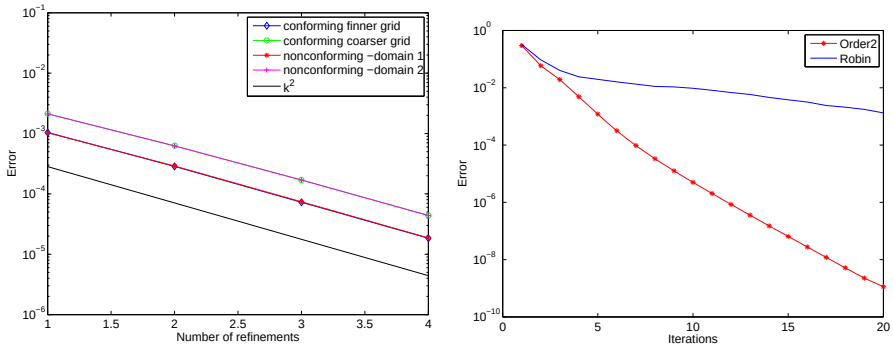


Fig. 3. Error between variational and DG-OSWR solutions versus the refinement in time (*left*), and versus the iterations (*right*).

the norms of the error in $L^\infty(I; L^2(\Omega_i))$ versus the number of refinements, for both subdomains. First we observe the order 2 in time for the nonconforming case. This fits the theoretical estimates, even though we have theoretical results only for Robin transmission conditions. Moreover, the error obtained in the nonconforming case, in the subdomain where the grid is finer, is nearly the same as the error obtained in the conforming finer case.

The computations are done using Order 2 transmissions. Indeed, the error between the multidomain and variational solutions decreases much faster with the Order 2 transmissions conditions than with the Robin transmissions conditions as we can see on Fig. 3 (right), in the conforming case.

We now consider advection-diffusion equations with discontinuous porosity:

$$\omega \partial_t u + \nabla \cdot (\boldsymbol{b} u - \nu \nabla u) = 0.$$

The physical domain is $\Omega = (0,1) \times (0,2)$, the final time is $T = 1.5$. The initial value is $u_0 = 0.5e^{-100((x-0.7)^2+(y-1.5)^2)}$. Domain Ω is split into two subdomains $\Omega_1 \times (0,1.5)$ and $\Omega_2 \times (0,1.5)$ with $(\frac{1}{2} - \frac{sin(2\pi s)}{8}, 2s)$, $0 < s < 1$ a parametrization of the interface, as in Fig. 4. The advection and diffusion coeffi-

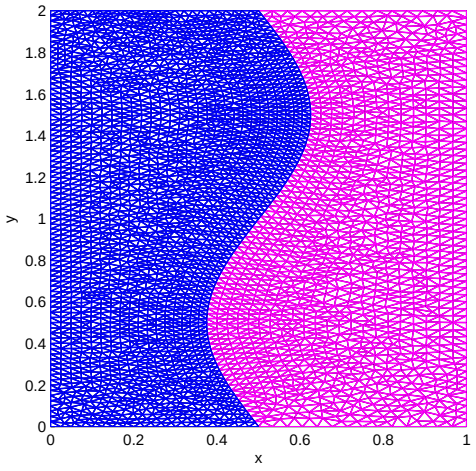


Fig. 4. Domain decomposition with Ω_1 (*left*) and Ω_2 (*right*).

cients are $\boldsymbol{b}_1 = (-sin(\frac{\pi}{2}(y-1))cos(\pi(x-\frac{1}{2})), 3cos(\frac{\pi}{2}(y-1))sin(\pi(x-\frac{1}{2})))$, $\nu_1 = 0.003$, $\omega_1 = 0.1$, and $\boldsymbol{b}_2 = \boldsymbol{b}_1$, $\nu_2 = 0.01$, $\omega_2 = 1$. We consider first a conforming grid in space. The mesh size over the interface is $h = 1/104$ and time step in Ω_1 is $k_1 = 1/128$, while in Ω_2, $k_2 = 1/94$. On Fig. 5, we observe, at final time $T = 1.5$, that the approximate solution computed using ten time windows and 3 iterations in each time window is close to the variational solution computed in one time window on the conforming finer space-time grid as shown on the error. We now consider nonconforming grids in space as shown on Fig. 4. The mesh size over the interface and time step in Ω_1 are $h_1 = 1/104$ and $k_1 = 1/128$, while in Ω_2, $h_2 = 1/81$ and $k_2 = 1/94$. On Fig. 6, we observe, at final time $T = 1.5$, that the approximate solution computed using 5 iterations in one time window is close to the variational solution computed on the conforming finer space-time grid. On Fig. 7 we observe the precision versus the mesh size and time step. The converged solution is such that the residual is smaller than 10^{-8}. A variational reference solution is

computed on a time grid with 2,048 time steps and 384 mesh grid. The space-time nonconforming solutions are interpolated on the previous grid to compute the error. We start with a time grid with 32 time steps and 24 mesh size for the left domain and time steps 12 and 12 mesh size for the right domain and divide by 2 the time step and mesh size several times. Figure 7 shows the norms of the error in $L^2(I; L^2(\Omega_i))$ versus the time steps, for both subdomains. We observe the order 2 for the nonconforming space-time case, even though we have theoretical results only for the time semi-discretized case in [7].

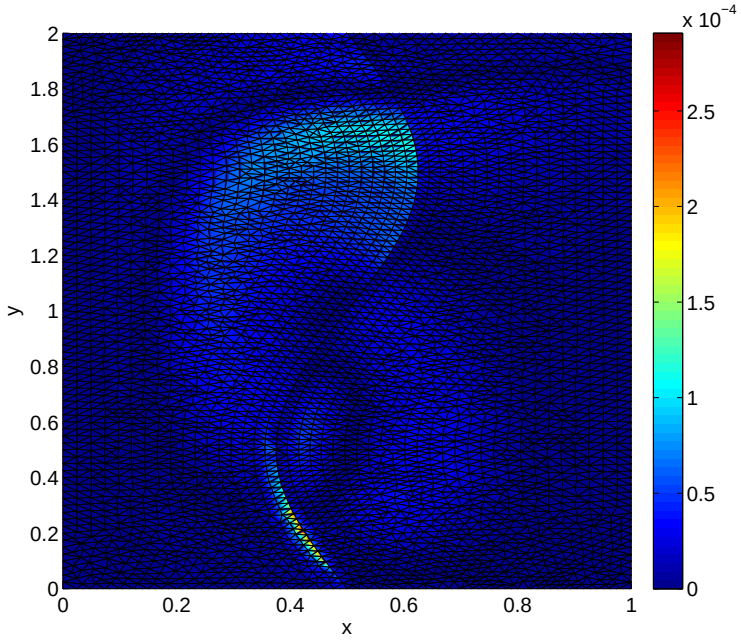


Fig. 5. Error between variational and DG-OSWR solutions, at final time, after 10 time windows and 3 iterations per window.

4 Conclusions

We have analyzed the continuous algorithm for variable discontinuous coefficients and general decompositions. We have shown numerically that the method preserves the order of the one domain scheme in the case of discontinuous variable coefficients, nonconforming grids in space and time and a curved interface. An analysis of the influence of the decomposition in time windows is in progress.

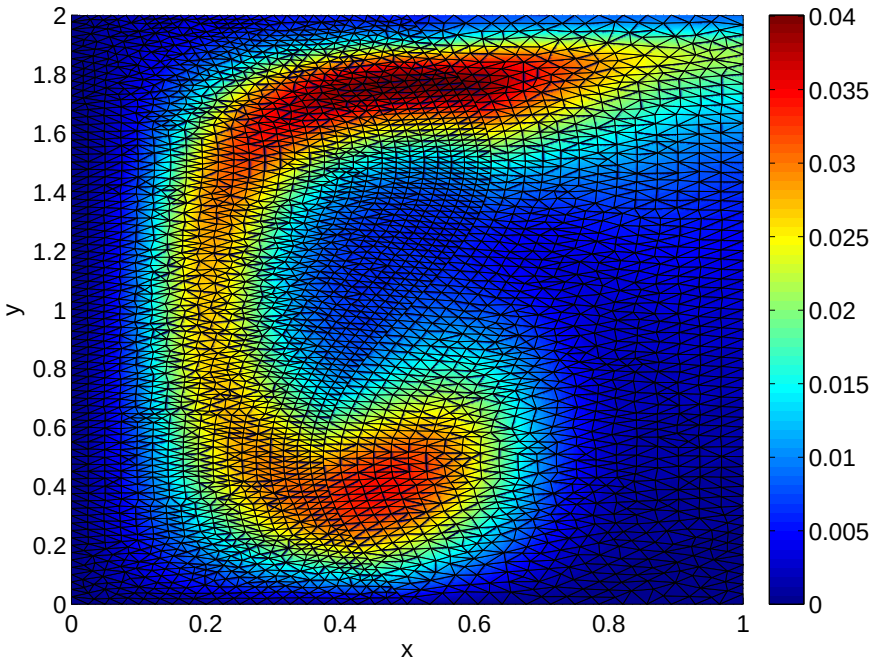


Fig. 6. DG-OSWR solution at final time, after 5 iterations.

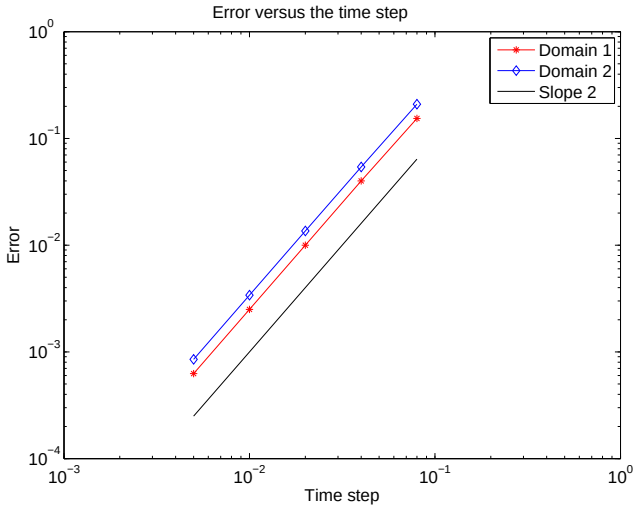


Fig. 7. Error curves versus the time step.

References

1. D. Bennequin, M.J. Gander, and L. Halpern. A homographic best approximation problem with application to optimized Schwarz waveform relaxation. *Math. Comput.*, 78:185–223, 2009.
2. E. Blayo, L. Halpern, and C. Japhet. Optimized Schwarz waveform relaxation algorithms with nonconforming time discretization for coupling convection-diffusion problems with discontinuous coefficients. In O.B. Widlund and D.E. Keyes, editors, *Decomposition Methods in Science and Engineering XVI*, volume 55 of *Lecture Notes in Computational Science and Engineering*, pp. 267–274. Springer Berlin, Heidelberg, New York, 2007.
3. M.J. Gander and L. Halpern. Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems. *SIAM J. Numer. Anal.*, 45(2):666–697, 2007.
4. M.J. Gander, L. Halpern, and M. Kern. Schwarz waveform relaxation method for advection–diffusion–reaction problems with discontinuous coefficients and non-matching grids. In O.B. Widlund and D.E. Keyes, editors, *Decomposition Methods in Science and Engineering XVI*, volume 55 of *Lecture Notes in Computational Science and Engineering*, pp. 916–920. Springer Berlin, Heidelberg, New York, 2007.
5. M.J. Gander, C. Japhet, Y. Maday, and F. Nataf. A new cement to Glue nonconforming grids with Robin interface conditions : The finite element case. In R. Kornhuber, R.H.W. Hoppe, J. Périaux, O. Pironneau, O.B. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering*, volume 40 of *Lecture Notes in Computational Science and Engineering*, pp. 259–266. Springer Berlin, Heidelberg, New York, 2005.
6. L. Halpern and C. Japhet. Discontinuous Galerkin and nonconforming in time optimized Schwarz waveform relaxation for heterogeneous problems. In U. Langer, M. Discacciati, D.E. Keyes, O.B. Widlund, and W. Zulehner, editors, *Decomposition Methods in Science and Engineering XVII*, volume 60 of *Lecture Notes in Computational Science and Engineering*, pp. 211–219. Springer Berlin, Heidelberg, New York, 2008.
7. L. Halpern, C. Japhet, and J. Szeftel. Discontinuous Galerkin and nonconforming in time optimized Schwarz waveform relaxation. In *Proceedings of the Eighteenth International Conference on Domain Decomposition Methods*, 2009. `http://numerik.mi.fu-berlin.de/DDM/DD18/` in electronic form. These proceedings in printed form.
8. C. Johnson, K. Eriksson, and V. Thomée. Time discretization of parabolic problems by the discontinuous Galerkin method. *RAIRO Modél. Math. Anal. Numér.*, 19, 1985.
9. V. Martin. An optimized Schwarz waveform relaxation method for the unsteady convection diffusion equation in two dimensions. *Appl. Numer. Math.*, 52:401–428, 2005.
10. V. Thomée. *Galerkin Finite Element Methods for Parabolic Problems*. Springer Berlin, Heidelberg, New York, 1997.

Convergence Behaviour of Dirichlet–Neumann and Robin Methods for a Nonlinear Transmission Problem

Heiko Berninger, Ralf Kornhuber, and Oliver Sander *

Fachbereich Mathematik und Informatik, Freie Universität Berlin, Berlin, Germany

Summary. We investigate Dirichlet–Neumann and Robin methods for a quasilinear elliptic transmission problem in which the nonlinearity changes discontinuously across two subdomains. In one space dimension, we obtain convergence theorems by extending known results from the linear case. They hold both on the continuous and on the discrete level. From the proofs one can infer mesh-independence of the convergence rates for the Dirichlet–Neumann method, but not for the Robin method. In two space dimensions, we consider numerical examples which demonstrate that the theoretical results might be extended to higher dimensions. Moreover, we investigate the asymptotic convergence behaviour for fine mesh sizes quantitatively. We observe a good agreement with many known linear results, which is remarkable in view of the nonlinear character of the problem.

1 Introduction

We consider the following setting. Let $\Omega \subset \mathbb{R}^n$ be a bounded Lipschitz domain divided into two non-overlapping subdomains Ω_1, Ω_2 with the interface $\Gamma = \overline{\Omega}_1 \cap \overline{\Omega}_2$. The outer normal of Ω_1 is denoted by $\mathbf{n}$. Furthermore, let $f \in L^2(\Omega)$ and k_1, $k_2 \in L^\infty(\mathbb{R})$ with $k_i \geq \alpha > 0$ for $i = 1, 2$. In strong form the domain decomposition problem that we aim at reads:

Find a function p in Ω, $p_i := p_{|\Omega_i} \in H^1(\Omega_i)$, $i = 1, 2$, $p_{|\partial\Omega} = 0$, such that

$$-\operatorname{div}(k_i(p_i)\nabla p_i) = f \qquad \text{on } \Omega_i, \ \ i = 1, 2 \tag{1}$$

$$p_1 = p_2 \qquad \text{on } \Gamma \tag{2}$$

$$k_1(p_1)\nabla p_1 \cdot \mathbf{n} = k_2(p_2)\nabla p_2 \cdot \mathbf{n} \qquad \text{on } \Gamma\,. \tag{3}$$

A powerful tool to treat problems of this kind is to introduce new variables u_i, $i = 1, 2$, by Kirchhoff transformations κ_i, defined by

$$u_i(x) := \kappa_i(p_i(x)) = \int_0^{p_i(x)} k_i(q)\,dq \quad \text{a.e. in } \Omega_i\,. \tag{4}$$

* This work was supported by the BMBF–Programm "Mathematik für Innovationen in Industrie und Dienstleistungen". We thank J. Schreiber for computational assistance.

This entails $k_i(p_i)\nabla p_i = \nabla u_i$ and, therefore, problem (1), (2) and (3) can be rewritten in the following form, in which the nonlinearity only appears on Γ, but now as a discontinuity condition on the primal variable:

Find a function u in Ω, $u_i := u_{|\Omega_i} \in H^1(\Omega_i)$, $i = 1, 2$, $u_{|\partial\Omega} = 0$, such that

$$-\Delta u_i = f \qquad \text{on } \Omega_i, \quad i = 1, 2 \tag{5}$$

$$\kappa_1^{-1}(u_1) = \kappa_2^{-1}(u_2) \qquad \text{on } \Gamma \tag{6}$$

$$\nabla u_1 \cdot \mathbf{n} = \nabla u_2 \cdot \mathbf{n} \qquad \text{on } \Gamma\,. \tag{7}$$

In the linear case, where k_i, $i = 1, 2$, are constant functions, Dirichlet–Neumann and Robin methods are well-understood iteration procedures for the treatment of non-overlapping elliptic domain decomposition problems, see, e.g., [7, 8, 10]. We introduce nonlinear versions of these methods applied to (5), (6) and (7) without using linearization. In one space dimension, both on the continuous and on the discrete level, we obtain convergence results by extending approaches used in the linear case, see [1]. We also obtain mesh-independent convergence rates for the damped Dirichlet–Neumann method, but not for the Robin method, just as in the linear case. However, these generalizations of the convergence proofs for the linear setting do not work in dimensions higher than one. Therefore, we investigate the qualitative and quantitative convergence properties in 2D numerically.

Concerning the nonlinear Dirichlet–Neumann method, we observe asymptotically mesh-independent optimal convergence rates for a certain mesh-independent optimal damping parameter. Moreover, if the nonlinearities k_1 and k_2 are of different orders of magnitude, the Dirichlet–Neumann method converges considerably faster than if they are of the same order of magnitude. Strangely enough, this observation can be made plausible by investigations that have been carried out on corresponding settings for the Robin method in the linear case, see [5].

As to the nonlinear Robin method, we observe degenerating optimal convergence rates and parameters if the two Robin parameters involved in the method coincide. What is more, we can even establish formulas, which quantitatively describe the asymptotic behaviour of this degeneracy, and which are very similar to the ones, that have been discovered for the Robin method applied to the linear case, cf. [9]. Results from the theory of optimized Schwarz methods in linear cases (see, e.g., [7]) show, that the convergence speed can be further increased by allowing the two Robin parameters to be different. Indeed, we obtain a better asymptotic behaviour for our test cases if we choose the parameters independently from each other. Finally, if the nonlinearities k_1 and k_2 are of different orders of magnitude, the optimized Robin method with different parameters converges quite fast with mesh-independent convergence rates, which, again, reproduces the linear situation as considered in [5].

Altogether, the observations we make in our nonlinear numerical examples, resemble strikingly well the proved results for linear cases.

2 Transmission Problem with Jumping Nonlinearities

In this section, we introduce some further notation (cf. [10]) and give a weak formulation of problem (5), (6) and (7). Then, we point out the equivalence of it with Steklov–Poincaré interface equations (cf. [3]).

In addition to the notation and definitions above, we introduce the spaces

$$V_i := \{v_i \in H^1(\Omega_i) \,|\, v_{i|\partial\Omega\cap\partial\Omega_i} = 0\}, \quad V_i^0 := H_0^1(\Omega_i), \quad \Lambda := H_{00}^{1/2}(\Gamma)$$

and for $w_i, v_i \in V_i$ the form $a_i(w_i, v_i) := (\nabla w_i, \nabla v_i)_{\Omega_i}$, where $(\cdot,\cdot)_{\Omega_i}$ stands for the L^2 inner product on Ω_i. The norm in Λ will be denoted by $\|\cdot\|_\Lambda$.

Let R_i, $i = 1, 2$, be any continuous extension operator from Λ to V_i. Then the variational formulation of problem (5), (6) and (7) reads as follows:

Find $u_i \in V_i$, $i = 1, 2$, such that

$$a_i(u_i, v_i) = (f, v_i)_{\Omega_i} \qquad \forall v_i \in V_i^0, \;\; i = 1, 2 \tag{8}$$

$$\kappa_1^{-1}(u_{1|\Gamma}) = \kappa_2^{-1}(u_{2|\Gamma}) \qquad \text{in } \Lambda \tag{9}$$

$$a_1(u_1, R_1\mu) - (f, R_1\mu)_{\Omega_1} = -a_2(u_2, R_2\mu) + (f, R_2\mu)_{\Omega_2} \quad \forall \mu \in \Lambda\,. \tag{10}$$

For details concerning the Kirchhoff transformations in the weak sense in (9), i.e., in the sense of superposition operators on $H^1(\Omega_i)$, see [2], where one can also find a proof of

Proposition 1. *The weak form of problem (1), (2) and (3) is equivalent to (8), (9) and (10).*

Now, for a given $\lambda \in \Lambda$ (and omitting brackets for operators applied to λ from now on), we consider the harmonic extensions $H_i(\kappa_i\lambda) \in V_i$ of the Dirichlet boundary value $\kappa_i\lambda$ on Γ for $i = 1, 2$. With these operators and denoting by $\langle\cdot,\cdot\rangle$ the duality pairing between Λ' and Λ, we recall that the Steklov–Poincaré operators $S_i : \Lambda \to \Lambda'$ are defined by

$$\langle S_i\eta, \mu\rangle = a_i(H_i\eta, H_i\mu) \quad \forall \eta, \mu \in \Lambda\,, \quad i = 1, 2\,.$$

Furthermore, let $\mathcal{G}_i f$ be the solutions of the subproblems (8) with homogeneous Dirichlet data $(\mathcal{G}_i f)_{|\partial\Omega_i} = 0$. We define the functional $\chi = \chi_1 + \chi_2 \in \Lambda'$ by

$$\langle \chi_i, \mu\rangle = (f, H_i\mu)_{\Omega_i} - a_i(\mathcal{G}_i f, H_i\mu) \quad \forall \mu \in \Lambda\,, \quad i = 1, 2\,.$$

Proposition 2. *By (4) and the relation*

$$u_i = H_i\kappa_i\lambda + \mathcal{G}_i f\,, \quad i = 1, 2\,,$$

between λ and u_i as well as with $\lambda_2 = \kappa_2\lambda$, problem (8), (9) and (10) is equivalent to each of the two Steklov–Poincaré interface equations

$$\textit{find } \lambda \in \Lambda: \qquad (S_1\kappa_1 + S_2\kappa_2)\lambda = \chi\,, \tag{11}$$

$$\textit{find } \lambda_2 \in \Lambda: \qquad (S_1\kappa_1\kappa_2^{-1} + S_2)\lambda_2 = \chi\,. \tag{12}$$

3 Nonlinear Dirichlet–Neumann and Robin Methods

In this section, we note the nonlinear Dirichlet–Neumann and Robin methods that we apply to (8), (9) and (10) in weak forms. We give Steklov–Poincaré formulations of the methods and convergence results in 1D generalizing linear theory.

3.1 The Methods and Their Steklov–Poincaré Formulations

The nonlinear Dirichlet–Neumann method applied to problem (8), (9) and (10) reads:
Given $\lambda_2^0 \in \Lambda$, find $u_1^{k+1} \in V_1$ and $u_2^{k+1} \in V_2$ for each $k \geq 0$ such that

$$a_1(u_1^{k+1}, v_1) = (f, v_1)_{\Omega_1} \qquad \forall v_1 \in V_1^0 \tag{13}$$

$$u_{1|\Gamma}^{k+1} = \kappa_1 \kappa_2^{-1}(\lambda_2^k) \qquad \text{in } \Lambda \tag{14}$$

and then

$$a_2(u_2^{k+1}, v_2) = (f, v_2)_{\Omega_2} \qquad \forall v_2 \in V_2^0 \tag{15}$$

$$a_2(u_2^{k+1}, H_2\mu) - (f, H_2\mu)_{\Omega_2} = -a_1(u_1^{k+1}, H_1\mu) + (f, H_1\mu)_{\Omega_1} \quad \forall \mu \in \Lambda\,. \tag{16}$$

Then, with some damping parameter $\theta \in (0, 1)$, the new iterate is defined by

$$\lambda_2^{k+1} := \theta\, u_{2|\Gamma}^{k+1} + (1-\theta)\lambda_2^k\,. \tag{17}$$

For the analysis (cf. [1, Sect. 3.3.2/3]), it is necessary to carry out the damping in the transformed space and to have a linear preconditioner in

Proposition 3. *The Dirichlet–Neumann method (13), (14), (15), (16) and (17) applied to problem (8), (9) and (10) is a* preconditioned Richardson procedure *for Eq. (12) with S_2 as a preconditioner. The iteration is given by $T_\theta : \Lambda \to \Lambda$ defined as*

$$T_\theta : \lambda_2^k \mapsto \lambda_2^{k+1} = \lambda_2^k + \theta S_2^{-1}(\chi - (S_1\kappa_1\kappa_2^{-1} + S_2)\lambda_2^k)\,. \tag{18}$$

In contrast to the Dirichlet–Neumann method, the Robin iteration is related to the symmetric equation (11), and it comes with two acceleration parameters $\gamma_1,\ \gamma_2 > 0$ rather than one. For problem (8), (9) and (10) it reads:
Given a $u_2^0 \in V_2$ find $u_1^{k+1} \in V_1$ and $u_2^{k+2} \in V_2$ for $k \geq 0$ such that

$$a_1(u_1^{k+1}, v_1) = (f, v_1)_{\Omega_1} \quad \forall v_1 \in V_1^0 \tag{19}$$

$$\begin{aligned} a_1(u_1^{k+1}, R_1\mu) - (f, R_1\mu)_{\Omega_1} + \gamma_1(\kappa_1^{-1}u_1^{k+1}, \mu)_\Gamma = \\ - a_2(u_2^k, R_2\mu) + (f, R_2\mu)_{\Omega_2} + \gamma_1(\kappa_2^{-1}u_2^k, \mu)_\Gamma \quad \forall \mu \in \Lambda \end{aligned} \tag{20}$$

and then

$$a_2(u_2^{k+1}, v_2) = (f, v_2)_{\Omega_2} \quad \forall v_2 \in V_2^0 \tag{21}$$

$$\begin{aligned} a_2(u_2^{k+1}, R_2\mu) - (f, R_2\mu)_{\Omega_1} + \gamma_2(\kappa_2^{-1}u_2^{k+1}, \mu)_\Gamma = \\ - a_1(u_1^{k+1}, R_1\mu) + (f, R_1\mu)_{\Omega_1} + \gamma_2(\kappa_1^{-1}u_1^{k+1}, \mu)_\Gamma \quad \forall \mu \in \Lambda\,. \end{aligned} \tag{22}$$

With the notation

$$\langle I\eta, \mu\rangle = (\eta, \mu)_\Gamma \quad \forall \eta, \mu \in \Lambda\,.$$

we obtain the following formulation of the Robin method in terms of Steklov–Poincaré operators (cf. [1, Sect. 3.4.2]), generalizing linear theory in [4, Sect. 5.4].

Proposition 4. *The Robin iteration (19)–(22) applied to (8)–(10) is equivalent to the* Alternating Direction Iterative (ADI) method *applied to (11). With a given $\lambda_2^0 \in \Lambda$ the operator $T_{\gamma_1,\gamma_2} : \Lambda \to \Lambda$, $T_{\gamma_1,\gamma_2} : \lambda_2^k \mapsto \lambda_2^{k+1}$ provided the ADI method is given by*

$$\lambda_2^{k+1} = (\gamma_2 I + S_2\kappa_2)^{-1}(\chi + (\gamma_2 I - S_1\kappa_1)(\gamma_1 I + S_1\kappa_1)^{-1}(\chi + (\gamma_1 I - S_2\kappa_2)\lambda_2^k))\,.$$

3.2 Convergence Results

The approach for proving convergence is as follows, cf. [1]. First, note that a fixed point λ of the iterative scheme in Proposition 3 or 4 is a solution of (12) or (11), respectively. Secondly, convergence proofs for linear cases can be extended so that Banach's fixed point theorem can be applied to T_θ and T_{γ_1,γ_2}.

We give sufficient conditions for convergence which are almost the same for both methods. In case of the Dirichlet–Neumann method they entail that T_θ is a contraction if θ is small enough, so that we obtain mesh-independent convergence rates. This is not provided by the convergence proof for the Robin method, and, even in linear cases, it is not true for the Robin iteration.

Generalizing [10, pp. 118/9] for the Dirichlet–Neumann method, we obtain

Theorem 1. *Let β_2 be the Lipschitz and α_2 be the coercivity constant of S_2. Let $S_1\kappa_1\kappa_2^{-1}$ be Lipschitz continuous with Lipschitz constant β_1 and strongly monotone with monotonicity constant α_1. Then (12) has a unique solution $\lambda_2 \in \Lambda$. Furthermore, for any given $\lambda_2^0 \in \Lambda$ and any $\theta \in (0, \theta_{\max})$ with $\theta_{\max}$ as in (23) the sequence given by (18) converges in Λ to λ_2. Theoretically optimal (i.e., minimal) convergence rates ρ_{opt} for corresponding optimal damping parameters θ_{opt} are given by*

$$\theta_{opt} = \frac{\theta_{\max}}{2} = \frac{\alpha_1+\alpha_2}{(\beta_1+\beta_2)^2}\cdot\frac{\alpha_2^2}{\beta_2} \quad \text{and} \quad \rho_{opt}^2 = 1 - \left(\frac{\alpha_1+\alpha_2}{\beta_1+\beta_2}\right)^2 \cdot \left(\frac{\alpha_2}{\beta_2}\right)^2 . \tag{23}$$

Theorem 2. *The assumptions in Theorem 1 are satisfied in 1D.*

We do not know whether the assertion of Theorem 1 is true for higher dimensions. We remark, however, that there are operators $S_1\kappa_1\kappa_2^{-1} : \Lambda \to \Lambda'$ in 2D, that are not monotone, see [1, Sect. 3.3.4].

Theorem 3. *We assume that the problems in (8) and (10) are discretized by piecewise linear finite elements and that in (9) piecewise linear interpolation is applied to the function after having been Kirchhoff–transformed at the nodes of the interface. Then Theorem 1 can also be applied to this discretization with the same constants and, thus, leads to mesh-independent optimal convergence rates and optimal damping parameters.*

For proving convergence of the Robin method (generalizing the linear result in [4, pp. 99/100]) we need $S_1\kappa_1, S_2\kappa_2 : \Lambda \to \Lambda'$ to be Lipschitz continuous and strongly monotone, which, by Theorem 2, is satisfied in 1D.

Theorem 4. *Let $\gamma_1 = \gamma_2 = \gamma > 0$ and $\Omega \subset \mathbb{R}$. Then for any initial iterate $\lambda_2^0 \in \Lambda$ the operator $\mathcal{T}_\gamma = \mathcal{T}_{\gamma_1,\gamma_2}$ in Proposition 4 provides a sequence $(\lambda_2^k)_{k\geq 0}$ which converges in Λ to the unique fixed point of $\mathcal{T}_\gamma$. Moreover, the sequence $(u_i^k)_{k\geq 1}$, $i = 1, 2$, of Robin iterates converges to the solution of (8), (9) and (10).*

For the discretization of problem (8), (9) and (10) in Theorem 3 the corresponding discrete version of the Robin method converges to the discrete solution.

4 Parameter Studies for the Dirichlet–Neumann Method

The purpose of this section is to apply our nonlinear Dirichlet–Neumann method (13), (14), (15), (16) and (17) to two concretely specified cases of the transmission problem in two space dimensions, discretized as in Theorem 3. After a detailed description of these two examples we present the numerical results which we discuss and compare to the linear case.

We consider problem (1), (2) and (3) on the unit Yin Yang domain Ω within a circle of radius 1 as shown in Fig. 1, with the coarse grid. We denote the white subdomain together with the grey circle B_1 by Ω_1 and the grey subdomain with the white circle B_2 by Ω_2. Furthermore, we select data f on Ω with $f_{|B_i} = f_i$ vanishing outside $B_1 \cup B_2$ and nonlinearities

$$k_i(p_i) = \begin{cases} K_i\, p_{b,i} \max\{(-p_i)^{-3\lambda_i - 2},\, c\} & \text{for } p_i \leq -1 \\ 1 & \text{for } p_i \geq -1 \end{cases}$$

with parameters K_i, $p_{b,i}$, λ_i specified in Tables 1 and 2. The ellipticity constant $c > 0$ is supposed to enforce convergence.

Our choice represents a nondegenerate stationary Richards equation without gravity on Ω_1 and Ω_2 containing two different soil types. f_1 and f_2 can be regarded as a source and a sink. In Case I, which we call mildly heterogeneous, we only alter one soil parameter $\lambda_1 \neq \lambda_2$ and choose $p_{b,i} = -1.0$ and $K_i = 2.0 \cdot 10^{-3}$ in both subdomains Ω_i as well as $c = 0.1$. In Case II, which we refer to as strongly heterogeneous, we change all parameters and use $c = 0.01$.

Starting with the coarse grid (level 1), we apply uniform refinement in order to obtain finer meshes, i.e., higher (refinement) levels. We discretize (8), (9) and (10) as described in Theorem 3. Figures 2 and 3 show the solutions p on Ω for the mildly

	f_i	λ_i
$i = 1$	1.0	0.1
$i = 2$	-1.0	1.0

Table 1. Case I.

	f_i	λ_i	$p_{b,i}$	K_i
$i = 1$	$5.0 \cdot 10^{-5}$	0.165	-0.373	$1.67 \cdot 10^{-7}$
$i = 2$	$-2.5 \cdot 10^{-3}$	0.694	-0.0726	$6.54 \cdot 10^{-5}$

Table 2. Case II.

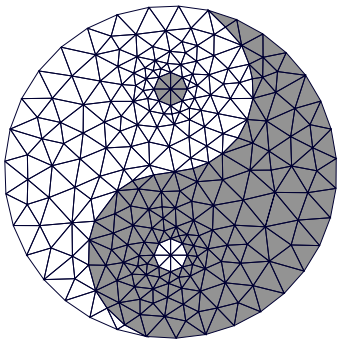


Fig. 1. Yin Yang domain Ω.

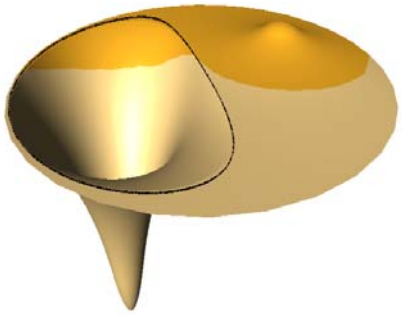


Fig. 2. Solution p on Ω in Case I (mildly heterogeneous).

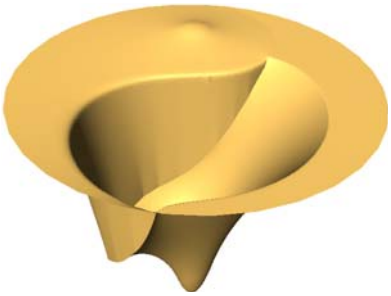


Fig. 3. Solution p on Ω in Case II (strongly heterogeneous).

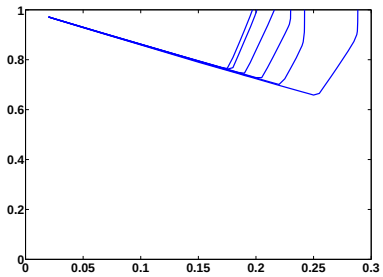


Fig. 4. ρ vs. θ on levels 1 (rightmost curve) to 6 (leftmost curve) in Case I.

and the strongly heterogeneous case, respectively. The crater-like parts of the graphs (indicated by a black line in Fig. 2) correspond to the nonlinear (hydrologically, the unsaturated) regime of the equation.

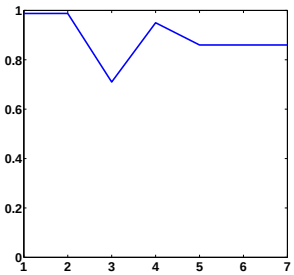


Fig. 5. θ_{opt} vs. level in Case II.

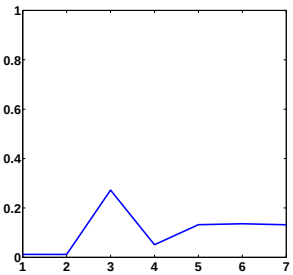


Fig. 6. ρ_{opt} vs. level in Case II.

For Case I, Fig. 4 shows average convergence rates ρ of the Dirichlet–Neumann method with respect to the damping parameter θ on the first six levels, from the rightmost curve representing the first level to the leftmost curve corresponding to the 6th level. The convergence rates are measured in the energy norm for the transformed variables. Starting with the initial iterates $u_i^0 = 0$, $i = 1, 2$, the Dirichlet–Neumann iteration is stopped when the relative error is below 10^{-12}. Each of the local problems on the subdomains is solved by 50 iterations of a linear multigrid which leads to numerically exact solutions. For the implementation we used the numerics environment DUNE [6].

Figure 4 shows that, as on the continuous level in Theorem 1, one obtains convergence if the damping parameter $\theta \in (0, 1)$ is below a threshold $\theta_{\max}$, and one observes optimal convergence rates ρ_{opt} for a certain θ_{opt}. Both the threshold and the optimal parameter as well as the corresponding optimal rates are level-dependent – however, these values seem to stabilize for higher levels. Concretely, the damping parameter $\theta_{opt} \approx 0.17$ leads to the optimal convergence rates $\rho_{opt} \approx 0.77$ on levels 5, 6 and 7. This indicates that mesh-independence is obtained in this 2D-case as was proved for 1D-cases (Theorem 3) and is known in linear settings (see [10, pp. 122–128]). Finally, we have the relationship $\rho_{opt} \approx 1 - \frac{7}{5}\,\theta_{opt}$ on all levels 1 to 7, which reflects (23).

In principle, the situation for Case II is the same as for Case I, see Figs. 5 and 6. Again, optimal convergence rates corresponding to optimal damping parameters seem to stabilize asymptotically for high levels, but now we need considerably less damping $\theta_{opt} \approx 0.85$ for much better optimal rates $\rho_{opt} \approx 0.15$ (on levels 5, 6 and 7) than in Case I. In addition, even for overrelaxation, i.e. for parameters $\theta > 1$, convergence can be observed (concretely, we obtain $\theta_{opt} = \theta_{\max}/2$ as in (23)). In contrast to Case I, the convergence rates remain stable even if we choose a much smaller $c > 0$, e.g., $c = 10^{-100}$.

A possible reason for this considerably improved convergence behaviour of the Dirichlet–Neumann method might be the big jumps of the diffusion coefficients K_1 and K_2 in Case II. Surprisingly, the numerical results in the next section, where we present the convergence behaviour of the nonlinear Robin method for the two test cases, will shed some light on this phenomenon, again supported by linear theory. Here, we want to discuss this issue heuristically, regardless of the linear or nonlinear nature of the problem, by considering the corresponding constants in Theorem 1. Motivated by $K_1 \ll K_2$ in Table 2, we assume that $\alpha_2 \simeq \beta_2$ have the same order of magnitude which is "big" compared to $\alpha_1 \simeq \beta_1$. Then, considering (23), we estimate roughly

$$\rho_{opt} = \sqrt{1 - \frac{\alpha_1 + \alpha_2}{\beta_2}\,\theta_{opt}} \simeq 1 - \frac{1}{2}\,\theta_{opt}\,.$$

(Compare this to the striking relationship $\rho_{opt} = 1 - \theta_{opt}$ obtained for levels 1 to 7 in Figs. 5 and 6.) With the same arguments, we find that θ_{opt} has the order of magnitude of 1 in this case, whereas it has the order of magnitude of α_1/β_2 if we exchange the Dirichlet-subdomain Ω_1 and the Neumann-subdomain Ω_2. Indeed, here, we only observe convergence for very small damping parameters in Case II, whereas we do

hardly see any change in Case I. Also, the convergence rates are very bad for Case II after exchanging domains. This, however, cannot be inferred from the formula in (23), but by numerical stability: One can argue that the smaller K_1 is, the better the Dirichlet problem is conditioned on Ω_1 (with respect to the Dirichlet value), and the bigger K_2 is, the better the Neumann problem is conditioned on Ω_2 (with respect to the Neumann value). For more illuminating theory on linear cases with discontinuous coefficients, which confirms some of our findings in Case II, consult [5, p. 97]. Altogether, in such asymmetric cases, the asymmetry of the Dirichlet–Neumann method reveals itself dramatically.

5 Parameter Studies for the Robin Method

In this last section, we present numerical results obtained by applying the nonlinear Robin method (19)–(22) to the test cases introduced in Sect. 4. For both cases, we first consider the Robin method with one Robin parameter $\gamma = \gamma_1 = \gamma_2$, for which our convergence result (Theorem 4) in 1D is valid, and secondly, we investigate the situation with different γ_1 and γ_2. In contrast to the Dirichlet–Neumann method, each subproblem (19)–(20) and (21)–(22) in the Robin iteration is nonlinear. We solve these local problems by a monotone multigrid method, see [1, Sect. 3.4.5]. The latter is stopped if the relative error of succeeding iterates in the energy norm drops below 10^{-12}. Otherwise, we use the same stopping criterion and average convergence rates as for the Dirichlet–Neumann method above.

Using the Robin iteration with $\gamma = \gamma_1 = \gamma_2$, we find that the numerical results of the two cases are virtually the same. Therefore, we only present Case II here. As one can see in Fig. 7, there are certain ranges for the Robin parameter γ on each level 1–6, where convergence rates are bounded away from 1. This is remarkable since Theorem 4 guarantees convergence for all $\gamma > 0$ in 1D. Furthermore – as for the Dirichlet–Neumann method – there is an optimal convergence rate ρ_{opt} obtained for an optimal γ_{opt} on each level. However – in contrast to the Dirichlet–Neumann method – these optimal rates and the corresponding parameters seem to degenerate rather than become asymptotically mesh-independent. The situation in Case I is almost the same as in Case II. However, the range of Robin parameters, for which an acceptable convergence speed is observed in the numerics, is about 10^4 times bigger than in Case II. Thus, a good choice of γ seems to be correlated to the factor in front of the Laplacian (compare (20)), which is by some orders of magnitude bigger in Case I than in Case II.

In convergence proofs for the Robin method on the continuous level, as in the original [8], one usually does not derive convergence rates (compare Sect. 3.2). This is because, usually, they are just not available. On the contrary, degeneracy of convergence rates is observed and proved on the discrete level for fine mesh sizes. In the world of optimized Schwarz methods, the latter can even be formulated quantitatively in form of asymptotic convergence results. For example, in linear cases the asymptotic behaviour

$$\gamma_{opt}^{lin} = \mathcal{O}(h^{-1/2}) \quad \text{and} \quad \rho_{opt}^{lin} = 1 - \mathcal{O}(h^{1/2}) \tag{24}$$

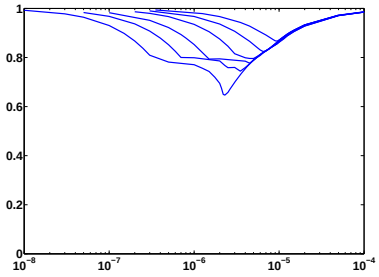


Fig. 7. ρ vs. γ on levels 1 (leftmost) to 6 (rightmost) for $\gamma_1 = \gamma_2$ in Case II

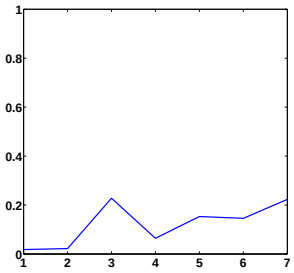


Fig. 8. ρ_{opt} vs. level for $\gamma_1 \neq \gamma_2$ in Case II

of the optimal parameters and convergence rates with respect to the mesh size h is known for quite general domains, see [9]. Now, if we investigate the asymptotics of the optimal parameters and rates in the nonlinear case II, displayed in Fig. 7, with respect to h, we find

$$\gamma_{opt} = \mathcal{O}(h^{-0.45}) \quad \text{and} \quad \rho_{opt} = 1 - \mathcal{O}(h^{0.44})\,. \tag{25}$$

Thus, we do not only observe an asymptotic behaviour of a similar kind as in the linear case, but even with similar exponents. The situation for Case I is virtually the same.

The convergence speed of the Robin method can be further increased by allowing the Robin parameters γ_1 and γ_2 to be different. We have carried out extensive numerical parameter studies for the performance of the nonlinear Robin method in both our cases on levels 1–8. Figures 9 and 10 shall serve as examples of the results we obtained on the 4th level in Case I (with 34,000 parameter pairs) and in Case II (with 77,000 parameter pairs), respectively. First of all, in both graphics, which contain the case $\gamma = \gamma_1 = \gamma_2$ on the diagonal, one can clearly see that the convergence speed can be increased by an appropriate choice of different Robin parameters.

Now, however, the situations in Case I and in Case II are completely different. We start by considering Case I, where the slopes of the nonlinearities in the subdomains are different but not their order of magnitude. Here, we observe that the convergence rates are nearly symmetric with respect to the diagonal $\gamma_1 = \gamma_2$ and that two local minima occur off the diagonal – a left (asymptotically global) one and a right one in Fig. 9. Although the convergence speed can be increased by choosing different instead of equal Robin parameters, asymptotically we still obtain degenerating optimal parameters and rates. However, we observe a weaker mesh-dependence of the convergence rates than for $\gamma_1 = \gamma_2$ in (25). Concretely, we find the asymptotic behaviour

$$\gamma_{1,opt} = \mathcal{O}(h^{-0.37})\,, \quad \gamma_{2,opt} = \mathcal{O}(h^{-0.55}) \quad \text{and} \quad \rho_{opt} = 1 - \mathcal{O}(h^{0.34}) \tag{26}$$

for the left minima and a similar one for the right minima.

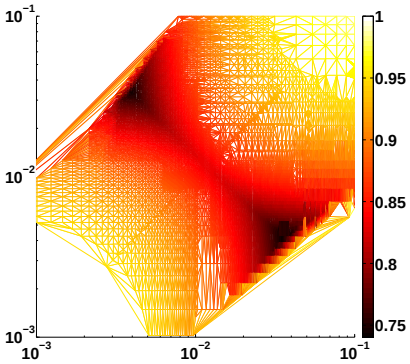


Fig. 9. ρ vs. γ_1 (x-axis) and γ_2 (y-axis) on level 4 for Case I.

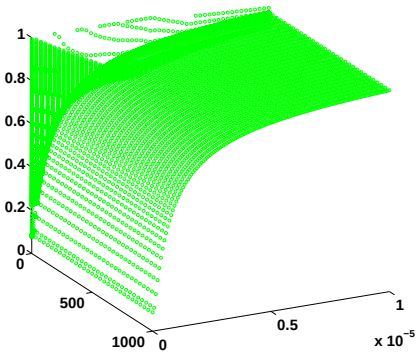


Fig. 10. ρ vs. γ_1 (x-axis) and γ_2 (y-axis) on level 4 for Case II.

As before in (24), our observations (26) in the nonlinear case I can be compared to known results from the linear theory of optimized Schwarz methods. In [7, p. 17] the asymptotic behaviour of different optimized Robin parameters and corresponding convergence rates has been derived for a linear equation on $\mathbb{R}^2$ decomposed into two half planes. The asymptotics is given by the formulas

$$\gamma_{1,opt}^{lin} = \mathcal{O}(h^{-1/4}), \quad \gamma_{2,opt}^{lin} = \mathcal{O}(h^{-1/4}) \quad \text{and} \quad \rho_{opt}^{lin} = 1 - \mathcal{O}(h^{1/4}). \tag{27}$$

A comparison with (27) shows that, quantitatively, the asymptotic behaviour of the different optimal Robin parameters in (26) does not seem to follow the linear results. Also, we do not obtain the same degree of acceleration of the convergence speed in (26) as suggested by the linear case. However, we observe a similar kind of asymptotic behaviour for ρ_{opt} and, at least, the asymptotics lies between the situations (24) and (27).

In contrast to Case I, the situation in Case II is very unsymmetric with respect to the diagonal $\gamma_1 = \gamma_2$, and we do no longer observe two distinct local minima of convergence rates. We rather have a whole strip of parameter pairs, where one parameter γ_2 is more or less fixed while the other γ_1 is free (as long as it is big enough), in which nearly constant globally minimal rates occur. Even for the global minimum, which is not distinct, one observes a difference in order of magnitude of at least $\gamma_{1,opt} \approx 10^4\,\gamma_{2,opt}$ on levels 1–8. Most importantly, however, the globally minimal rates in the strip are asymptotically stable, i.e., mesh-independent. This can be seen in Fig. 8, where the value for the 7th level is the same as for the 8th level. Note that with extreme values $\gamma_{1,opt} \gg \gamma_{2,opt}$ subproblems (19)–(20) and (21)–(22) resemble Dirichlet and Neumann problems, respectively, i.e. the Robin method becomes an undamped Dirichlet–Neumann method. This observation is quite striking if we compare Fig. 8 for the optimized Robin method with two different parameters with Fig. 6, which shows the optimal convergence rates for the damped Dirichlet–Neumann method.

We close this section by mentioning a known result on the Robin method applied to a linear equation with discontinuous coefficients $K_1/K_2 < 1$ in $\mathbb{R}^2$, decomposed into two half planes, see [5, p. 84]. The asymptotic behaviour in this case is given by

$$\gamma_{1,opt}^{lin} = \mathcal{O}(1)\,, \quad \gamma_{2,opt}^{lin} = \mathcal{O}(h^{-1}) \quad \text{and} \quad \rho_{opt}^{lin} = \frac{K_1}{K_2} - \mathcal{O}(h^{1/2})\,.$$

Although, again, we cannot confirm the asymptotic behaviour for the optimized Robin parameters in our Case II, this rare result of a mesh-independent convergence rate for the Robin method makes our findings in this and in the previous section on the good convergence of our optimized methods in Case II a bit more understandable.

References

1. H. Berninger. *Domain Decomposition Methods for Elliptic Problems with Jumping Nonlinearities and Application to the Richards Equation*. PhD thesis, Freie Universität, Berlin, 2007.
2. H. Berninger. Non-overlapping domain decomposition for the Richards equation via superposition operators. In *Domain Decomposition Methods in Science and Engineering XVIII*, volume 70 of *Lecture Notes in Computational Science and Engineering*, pp. 169–176. Springer Berlin, Heidelberg, New York, 2009.
3. H. Berninger, R. Kornhuber, and O. Sander. On nonlinear Dirichlet–Neumann algorithms for jumping nonlinearities. In *Domain Decomposition Methods in Science and Engineering XVI*, volume 55 of *Lecture Notes in Computational Science and Engineering*, pp. 483–490. Springer Berlin, Heidelberg, New York, 2007.
4. M. Discacciati. *Domain Decomposition Methods for the Coupling of Surface and Groundwater Flows*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne 2004.
5. O. Dubois. *Optimized Schwarz Methods for the Advection-Diffusion Equation and for Problems with Discontinuous Coefficients*. PhD thesis, McGill University, 2007.
6. P. Bastian et al. A generic grid interface for parallel and adaptive scientific computing. Part II: Implementation and tests in DUNE. *Computing*, 82(2–3):121–138, 2008.
7. M.J. Gander. Optimized Schwarz methods. *SIAM J. Numer. Anal.*, 44(2):699–731, 2006.
8. P.L. Lions. On the Schwarz alternating method. III: A variant for nonoverlapping subdomains. In *Domain Decomposition Methods for Partial Differential Equations, Proceedings of the 3rd International Symposium*, pp. 202–223. SIAM, Philadelphia, PA 1990.
9. S.H. Lui. A Lions non-overlapping domain decomposition method for domains with an arbitrary interface. *IMA J. Numer. Anal.*, 29(2):332–349, 2009.
10. A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford Science Publications, Oxford, 1999.

Part II

Minisymposia

Optimal Interface Conditions for an Arbitrary Decomposition into Subdomains

Martin J. Gander and Felix Kwok

Section de mathématiques, Université de Genève, Geneva CH-1211, Switzerland,
Martin.Gander@unige.ch; Felix.Kwok@unige.ch

Summary. The use of Dirichlet-to-Neumann operators as transmission conditions is known to yield optimal Schwarz methods that converge in a finite number of iterations when the subdomain decomposition has tree-like connectivity. However, it remains an open problem whether it is possible to construct a finitely terminating algorithm for arbitrary decompositions. In this article, we construct a Schwarz method that converges in exactly two steps for any decomposition into subdomains with minimal overlap. In this method, every subdomain must communicate with all other subdomains, but only data along subdomain boundaries need to be exchanged.

1 Optimal Interface Conditions

The convergence rate of Schwarz-type domain decomposition methods is very sensitive to the transmission condition used. Thus, it is natural to ask, for a given PDE and a given decomposition into subdomains, whether there exists a set of optimal interface conditions that leads to convergence in a finite number of steps. For a decomposition into vertical strips, we know that the Dirichlet-to-Neumann (DtN) operators yield such an optimal algorithm, see [4, 5, 6]. A similar result for decompositions whose connectivity graph contains no cycles is shown in [7]. It remains an open question to show whether similar operators exist for arbitrary decompositions.

The goal of this paper is to show that such an operator exists, at least in the discrete case, if we allow global communication between the subdomains, i.e., if each subdomain has access to the interface values of all the other subdomains. More precisely, we construct a subdomain iteration that converges to the exact solution in two steps by exchanging only data along subdomain boundaries.

We note that in general, these optimal interface conditions are nonlocal pseudo-differential operators, which are difficult to use in practice. Thus, the algorithm presented here is not meant to be implemented in a practical solver. However, practical algorithms can by derived by approximating the optimal operators by differential operators, see [3, 5] as well as [2] and references therein. Thus, our results serve as a starting point for this approximation process.

2 Notation and Assumptions

Let $\Omega \subset\subset \mathbb{R}^n$ be an open set. Suppose we want to solve the elliptic PDE

$$\mathcal{L}u = f \quad \text{on } \Omega, \qquad u = g \quad \text{on } \partial\Omega \tag{1}$$

by discretizing it to obtain the non-singular system $A\mathbf{u} = \mathbf{f}$ and using a domain decomposition method. Let Σ be the degrees of freedom therein. Suppose Ω is subdivided into nonoverlapping subdomains $\tilde{\Omega}_j$, $j = 1, \ldots, N$, and let $\tilde{\Sigma}_j$ be the discrete degrees of freedom contained within $\tilde{\Omega}_j$. Let $\{\Omega_j\}_{j=1}^N$ be an *overlapping* decomposition with degrees of freedom Σ_j, such that $\tilde{\Omega}_j \subset \Omega_j$ (and correspondingly $\tilde{\Sigma}_j \subset \Sigma_j$), and let R_j and $R_{\neg j}$ be operators that restrict Σ onto Σ_j and $\Sigma \setminus \Sigma_j$ respectively. We then define, for each $l = 1, \ldots, N$, the operator $\tilde{R}_l$, which has the same size as R_l, such that

$$[\tilde{R}_l]_{ij} = \begin{cases} 1 & \text{if } [R_l]_{ij} = 1 \text{ and } j \in \tilde{\Sigma}_l, \\ 0 & \text{otherwise.} \end{cases}$$

For each $j = 1, \ldots, N$, we define the matrices

$$A_j = R_j A R_j^T, \quad B_j = R_j A R_{\neg j}^T, \quad C_j = R_{\neg j} A R_j^T, \quad D_j = R_{\neg j} A R_{\neg j}^T.$$

We assume that D_j is nonsingular for all j, so that the Schur complement $A_j - B_j D_j^{-1} C_j$ is well-defined and non-singular. We now state the main assumption that will be used throughout the paper.

Assumption 1 (Sufficient Overlap) For all $j = 1, \ldots, N$, we have

$$\tilde{R}_j^T (R_j A - A_j R_j) = 0. \tag{2}$$

Assumption 1 states that the overlapping subdomain Σ_j needs to be sufficiently large, so that if v is a degree of freedom in $\tilde{\Sigma}_j$, then its stencil does not extend beyond Σ_j. This assumption is easily satisfied if the PDE is discretized using a compact stencil, because we can always construct Σ_j (and hence Ω_j) based on $\tilde{\Sigma}_j$ by extending it to include all points touched by the stencil.

3 Construction of the Method

The first step in constructing the method is to observe that the exact subdomain solution $\mathbf{u}_j = R_j \mathbf{u}$ can be obtained by solving the *Schur complement system*

$$(A_j - B_j D_j^{-1} C_j)\mathbf{u}_j = R_j \mathbf{f} - B_j D_j^{-1} R_{\neg j} \mathbf{f}. \tag{3}$$

If each subdomain has access to the right-hand side of all the other subdomains, then in principle we would be able to obtain $\mathbf{u}_j$ in one pass by solving each Schur complement system independently. However, this would not lead to an optimal Schwarz

method, because Schwarz methods only exchange information on $\mathbf{u}_i$ along subdomain boundaries. Thus, to construct an optimal Schwarz method, we must try to recover $R_{\neg j}\mathbf{f}$ using subdomain solutions only.

To do so, let us examine more closely what happens when we solve (3). First, we rewrite (3) using the definitions of A_j, B_j:

$$R_j A(R_j^T - R_{\neg j}^T D_j^{-1} C_j)\mathbf{u}_j = R_j\mathbf{f} - R_j A R_{\neg j}^T D_j^{-1} R_{\neg j}\mathbf{f}. \tag{4}$$

If we multiply (4) from the left by $\tilde{R}_j^T$, then the sufficient overlap assumption (2) implies that $\tilde{R}_j^T R_j A = \tilde{R}_j^T A_j R_j$. Thus, we get

$$\tilde{R}_j^T A_j R_j (R_j^T - R_{\neg j}^T D_j^{-1} C_j)\mathbf{u}_j = \tilde{R}_j^T R_j\mathbf{f} - \tilde{R}_j^T A_j R_j R_{\neg j}^T D_j^{-1} R_{\neg j}\mathbf{f}.$$

Since $R_j R_j^T = I$ and $R_j R_{\neg j}^T = 0$ (they restrict to Σ_j and $\Sigma \setminus \Sigma_j$ respectively, which are disjoint sets), the above equation simplifies to

$$\tilde{R}_j^T A_j \mathbf{u}_j = \tilde{R}_j^T R_j\mathbf{f}. \tag{5}$$

This means if $\mathbf{u}_j$ is the solution of (3), it is always possible to reconstruct $\tilde{R}_j\mathbf{f}$, the portion of $\mathbf{f}$ located in the *nonoverlapping* part of the subdomain, using only the subdomain solution $\mathbf{u}_j$. Since $\mathbf{f} = \sum_{i=1}^N \tilde{R}_i^T R_i\mathbf{f}$ and $R_{\neg j}\tilde{R}_j^T = 0$, we can substitute these relations into (3) to obtain the following algorithm.

Algorithm 1 For $k = 1, 2, \ldots$, and for $j = 1, \ldots, N$, solve

$$(A_j - B_j D_j^{-1} C_j)\mathbf{u}_j^{k+1} = R_j\mathbf{f} - \textstyle\sum_{i\neq j} T_{ji}\mathbf{u}_i^k, \tag{6}$$

where $T_{ji} = B_j D_j^{-1} R_{\neg j}\tilde{R}_i^T A_i$ is the transmission operator from Σ_i to Σ_j.

Theorem 1. *Let* $\mathbf{u}$ *be the exact solution to the problem* $A\mathbf{u} = \mathbf{f}$. *Then for any initial guess* $\mathbf{u}_j^0$, *Algorithm 1 converges to the exact solution in at most two iterations, i.e.,* $\mathbf{u}_j^2 = R_j\mathbf{u}$.

Proof. Since $\mathbf{u}_j^1$ is the solution of (3), by (5) we have $\tilde{R}_j^T A_j \mathbf{u}_j^1 = \tilde{R}_j^T R_j\mathbf{f}$. So the second step of Algorithm 1 gives

$$\begin{aligned}
(A_j - B_j D_j^{-1} C_j)\mathbf{u}_j^2 &= R_j\mathbf{f} - \sum_{i\neq j} B_j D_j^{-1} R_{\neg j}\tilde{R}_i^T A_i \mathbf{u}_i^1 \\
&= R_j\mathbf{f} - B_j D_j^{-1} R_{\neg j} \sum_{i\neq j} \tilde{R}_i^T R_i\mathbf{f} \\
&= R_j\mathbf{f} - B_j D_j^{-1} R_{\neg j} \sum_{i=1}^{N} \tilde{R}_i^T R_i\mathbf{f} \qquad (\text{since } R_{\neg j}\tilde{R}_j^T = 0) \\
&= R_j\mathbf{f} - B_j D_j^{-1} R_{\neg j}\mathbf{f},
\end{aligned}$$

which is exactly the Schur complement formulation of the system with the correct right hand side. This implies $\mathbf{u}_j^2 = R_j\mathbf{u}$, as required.

We now compare Algorithm 1 with the well-known parallel Schwarz method with optimal transmission conditions for the *tree case*:

$$(A_j - B_j D_j^{-1} C_j)\mathbf{u}_j^{k+1} = R_j\mathbf{f} - \sum_{(i,j)\in E}(B_j R_{\neg j} R_i^T + B_j D_j^{-1} C_j R_j R_i^T)\mathbf{u}_i^k,$$

where the sum is over all Ω_i that are neighbors of Ω_j. We know that the classical optimal algorithm only converges after $D+1$ iterations, where D is the diameter of the connectivity graph, see [4, 6, 7]. In contrast, Theorem 1 shows that Algorithm 1 will converge in at most two iterations, regardless of the number of subdomains and the topology of the decomposition. This comes at a cost: Algorithm 1 requires global communication among subdomains at every iteration, unlike its classical counterpart, which only requires communication between neighbors. Finally, we will show numerically in Sect. 5 that the classical algorithm can fail to converge when the decomposition is not a tree, while Algorithm 1 converges for decompositions with arbitrary connectivity.

4 Sparsity Pattern

Formula (6) seems to suggest at every step of Algorithm 1, every subdomain must have access to the entire solution in every other subdomain. This is in fact not the case. To understand which values really need to be transmitted, we study the sparsity pattern of T_{ji}, the operator through which subdomain j obtains information from $\mathbf{u}_i$. We show that this operator contains mostly zero columns, which means the corresponding nodal values are in fact discarded (and thus not needed). We first introduce the notion of the support of a vector.

Definition 1 (Support of a vector) Let $\mathbf{v}$ be a vector with degrees of freedom in Σ. Then the *support* of $\mathbf{v}$, denoted by $\mathrm{supp}(\mathbf{v})$, is the set of all points in Σ corresponding to nonzero entries in $\mathbf{v}$.

The following equivalences are immediate based on the definitions of supp:

(i) $\mathrm{supp}(\mathbf{v}) \subset \Sigma_j \iff R_j^T R_j \mathbf{v} = \mathbf{v} \iff R_{\neg j}\mathbf{v} = 0,$
(ii) $\mathrm{supp}(\mathbf{v}) \subset \tilde{\Sigma}_j \iff \tilde{R}_j^T R_j \mathbf{v} = \mathbf{v},$
(iii) $\mathrm{supp}(\mathbf{v}) \cap \Sigma_j = \emptyset \iff R_{\neg j}^T R_{\neg j}\mathbf{v} = \mathbf{v} \iff R_j\mathbf{v} = 0.$
(iv) $\mathrm{supp}(\mathbf{v}) \cap \tilde{\Sigma}_j = \emptyset \iff \tilde{R}_j\mathbf{v} = 0.$

Next, we identify the zero columns of T_{ji}. We do so by multiplying T_{ji} by a standard basis vector $\mathbf{e}_x$ and checking whether the product is zero.

Lemma 1. *Let $x \in \Sigma_i$ and $\mathbf{e}_x$ be its basis vector (1 at x and 0 everywhere else). Then we have $T_{ji}R_i\mathbf{e}_x = 0$ in each of the following cases:*

(i) $\mathrm{supp}(A\mathbf{e}_x) \cap \tilde{\Sigma}_i = \emptyset$;
(ii) $\mathrm{supp}(A\mathbf{e}_x) \subset \Sigma_j$;
(iii) $\{x\} \cup \mathrm{supp}(A\mathbf{e}_x) \subset \tilde{\Sigma}_i \setminus \Sigma_j$.

Proof. First, we rewrite $T_{ji}R_i\mathbf{e}_x$ as

$$T_{ji}R_i\mathbf{e}_x = B_jD_j^{-1}R_{\neg j}\tilde{R}_i^TA_iR_i\mathbf{e}_x = B_jD_j^{-1}R_{\neg j}\tilde{R}_i^TR_iA\mathbf{e}_x$$

by the sufficient overlap condition. We then consider the three cases:

(i) $\mathrm{supp}(A\mathbf{e}_x)\cap\tilde{\Sigma}_i=\emptyset$: we have

$$T_{ji}R_i\mathbf{e}_x = B_jD_j^{-1}R_{\neg j}(\tilde{R}_i^TR_iA\mathbf{e}_x) = B_jD_j^{-1}R_{\neg j}(R_i^T\underbrace{\tilde{R}_iA\mathbf{e}_x}_{=0}) = 0.$$

(ii) $\mathrm{supp}(A\mathbf{e}_x)\subset\Sigma_j$:

We have $\mathrm{supp}(\tilde{R}_iR_i^TA\mathbf{e}_x)\subset\mathrm{supp}(A\mathbf{e}_x)\subset\Sigma_j$, so $R_{\neg j}\tilde{R}_iR_i^TA\mathbf{e}_x=0$.

(iii) $\{x\}\cup\mathrm{supp}(A\mathbf{e}_x)\subset\tilde{\Sigma}_i\setminus\Sigma_j$:

Since $\mathrm{supp}(A\mathbf{e}_x)\subset\tilde{\Sigma}_i$, we have $\tilde{R}_iR_i^TA\mathbf{e}_x=A\mathbf{e}_x$. Now consider the linear system $D_j\mathbf{y}=R_{\neg j}A\mathbf{e}_x$, which can be rewritten as $R_{\neg j}AR_{\neg j}^T\mathbf{y}=R_{\neg j}A\mathbf{e}_x$. Since $x\notin\Sigma_j$, we see that $\mathbf{y}=R_{\neg j}\mathbf{e}_x$ satisfies the equation (because $R_{\neg j}^TR_{\neg j}\mathbf{e}_x=\mathbf{e}_x$). This is also the unique solution because D_j is nonsingular. Thus, we have $D_j^{-1}R_{\neg j}\tilde{R}_i^TR_iA\mathbf{e}_x=D_j^{-1}R_{\neg j}A\mathbf{e}_x=R_{\neg j}\mathbf{e}_x$. Now we multiply from the left by B_j to obtain

$$T_{ji}R_i\mathbf{e}_x = \underbrace{R_jAR_{\neg j}^T}_{B_j}R_{\neg j}\mathbf{e}_x = R_jA\mathbf{e}_x = 0,$$

since $\mathrm{supp}(A\mathbf{e}_x)$ lies completely outside Σ_j.

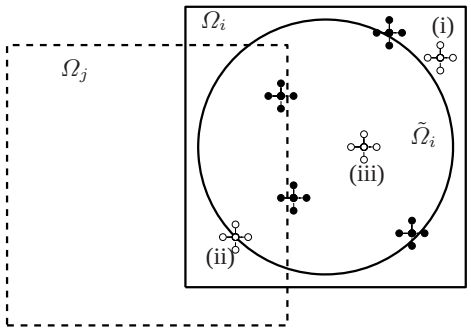


Fig. 1. A sketch showing stencils associated with different points in Ω_i. Stencils with solid nodes indicate points x at which $T_{ji}R_i\mathbf{e}_x\neq 0$; those with hollow nodes indicate points at which $T_{ji}R_i\mathbf{e}_x=0$ for the three cases indicated in Lemma 1.

Each of the three cases in Lemma 1 is illustrated in Fig. 1, where the hollow stencils indicate points that get mapped to zero by T_{ji}. Case (i) (top right-hand corner) happens when the stencil falls completely outside $\tilde{\Omega}_i$; case (ii) (bottom-left of

Ω_i) occurs when the stencil falls entirely within Ω_j. Finally, case (iii) occurs when the stencil is completely inside $\tilde{\Omega}_i \setminus \Omega_j$, just like the stencil near the center of the graphic. Thus, we see from Fig. 1 that the only points with $T_{ji}R_i\mathbf{e}_x \neq 0$ are those indicated by solid nodes, i.e. those that are so close to the boundary of $\tilde{\Omega}_i \setminus \Omega_j$ that their stencils straddle the boundary. These, in fact, are the only nodal values that must be transmitted. For a five-point stencil, this corresponds to a layer with a thickness of two nodes (one on each side of the boundary, see Fig. 2 in the next section); for wider stencils, e.g., for higher-order equations, this layer becomes thicker, but the number of values transmitted is still proportional to the length of $\partial(\tilde{\Omega}_i \setminus \Omega_j)$, which is one dimension lower than the set of all nodes in $\tilde{\Omega}_i$. If we define P_{ji} to be the restriction operator from Σ_i to the set of boundary nodes along $\partial(\tilde{\Omega}_i \setminus \Omega_j)$, then we can rewrite Algorithm 1 as follows:

Algorithm 2 For $k = 1, 2, \ldots$, and for $j = 1, \ldots, N$, solve

$$(A_j - B_j D_j^{-1} C_j)\mathbf{u}_j^{k+1} = R_j\mathbf{f} - \textstyle\sum_{i\neq j} \tilde{T}_{ji}\tilde{\mathbf{u}}_i^k,$$

where $\tilde{T}_{ji} = B_j D_j^{-1} R_{\neg j} \tilde{R}_i^T A_i P_{ji}^T$ and $\tilde{\mathbf{u}}_i^k = P_{ji}\mathbf{u}_i^k$.

Remark Algorithms 1 and 2 have identical iterates if the same initial guesses are used. Their only difference is that the latter does not transmit data corresponding to zero columns in T_{ji}, i.e., data that would be discarded anyway. This reduces the communication costs by a factor of H/h, where h is the fine mesh parameter and H is the size of the subdomain.

5 Numerical Examples

In this section, we present two examples in which we compare the convergence behavior of Algorithm 2 with that of the classical parallel Schwarz method with optimal transmission conditions, which is known to converge in a finite number of iterations in the tree case. For simplicity, in both cases we solve the 2D Poisson equation with Dirichlet boundary conditions, using the standard 5-point discretization. However, since the methods are derived purely algebraically, they are in principle applicable to any discretized PDE, provided we can define the subdomains so that they satisfy the sufficient overlap assumption, and that the subdomain problems are well posed.

Example 1 Here we decompose a rectangular domain into 6 vertical strips, as shown in Fig. 2a. Since the diameter of the connectivity graph is $D = 5$, we know that the parallel Schwarz method with optimal transmission conditions will converge in at most 6 steps; this is verified by the numerical results shown in Table 1. In contrast, Algorithm 2 converges in exactly two steps; this is in agreement with Theorem 1. Finally, Fig. 2a shows the communication pattern for both algorithms. As predicted by Lemma 1, the only nodal values that need to be transmitted are located on either side of the subdomain boundaries. Also, whereas the classical algorithm only takes

information from its neighbors, Algorithm 2 communicates with every subdomain, which makes it possible to converge in two iterations.

Example 2 In this example, we use the 4×4 decomposition shown in Fig. 2b. Since the connectivity graph is no longer a tree, we can no longer expect optimal parallel Schwarz to converge after a finite number of steps. Indeed, we see from Table 1 that the iteration diverges. This happens because of two reasons. First, since there are points belonging to more than two subdomains (i.e., cross points), optimal parallel Schwarz actually applies redundant updates at these points, leading to divergence, see [1]. In addition, unlike the tree case, $\partial\Omega_j$ is divided among several subdomains, so the boundary values obtained by Ω_j are no longer the trace of a harmonic function; instead, they are the trace of a function that fails to be harmonic at the partition points. Despite these difficulties, Algorithm 2 still converges in two iterations; the operators $\tilde{T}_{ji}$ are able to extract the right interface information and combine them the right way for the method to converge.

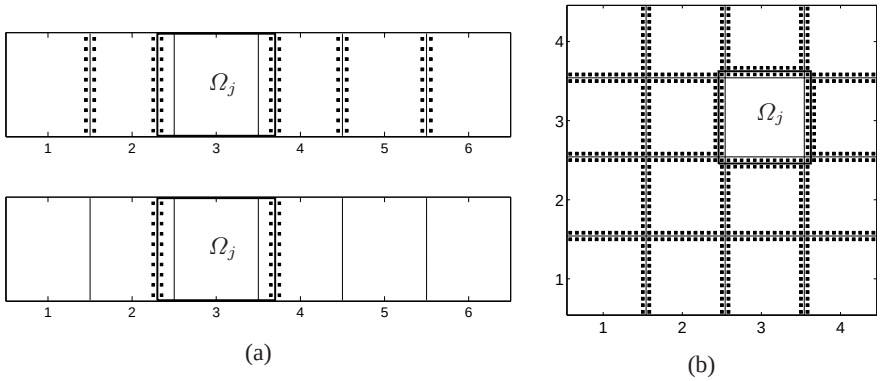


Fig. 2. Communication pattern for two decompositions into subdomains. *Black squares* indicate nodal values required by Ω_j, which is enclosed by *thick solid lines*. (**a**) decomposition into *vertical strips*. The *top figure* shows the values required by Algorithm 2, and the *bottom* those required by classical Parallel Schwarz with optimal transmission conditions. (**b**) a 4×4 decomposition, shown with the communication pattern for Algorithm 2.

6 Conclusion

We presented a new Schwarz method that converges in exactly two iterations when the domain decomposition satisfies the sufficient overlap assumption. Unlike the classical algorithm, the optimal transmission conditions we derived can handle arbitrary subdomain topologies. In our algorithm, each subdomain must communicate with all the other subdomains at each step; however, one only needs to exchange

Table 1. Parallel Schwarz with optimal transmission conditions versus Algorithm 2. In each case, we report the maximum L^∞ errors over all subdomains.

	Example 1 (6×1)		Example 2 (4×4)	
Its.	Parallel Schwarz	Algorithm 2	Parallel Schwarz	Algorithm 2
1	3.605×10^{0}	3.681×10^{0}	6.987×10^{1}	6.965×10^{1}
2	2.176×10^{-1}	1.066×10^{-14}	1.191×10^{2}	8.527×10^{-13}
3	1.252×10^{-2}		5.438×10^{1}	
4	7.328×10^{-4}		4.652×10^{2}	
5	3.278×10^{-5}		1.118×10^{3}	
6	1.066×10^{-14}		3.894×10^{3}	

data along a coarse grid structure containing the subdomain boundaries. Since its derivation is based only on sparse matrices, the method is in principle applicable to any PDE, or even systems of PDEs, as long as the subdomain problems remain solvable. Ongoing work focuses on deriving approximate local operators to obtain more efficient implementations.

References

1. E. Efstathiou and M.J. Gander. Why restricted additive Schwarz converges faster than additive Schwarz. *BIT*, 43(suppl.):945–959, 2003.
2. M.J. Gander. Optimized Schwarz methods. *SIAM J. Numer. Anal.*, 44(2):699–731 (electronic), 2006.
3. M.J. Gander, F. Magoules, and F. Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24:38–60, 2002.
4. F. Magoulès, F. Roux, and S. Salmon. Optimal discrete transmission conditions for a nonoverlapping domain decomposition method for the Helmholtz equation. *SIAM J. Sci. Comput.*, 25(5):1497–1515 (electronic), 2004.
5. F. Nataf and F. Rogier. Factorization of the convection-diffusion operator and the Schwarz algorithm. *Math. Models Methods Appl. Sci.*, 5:67–93, 1995.
6. F. Nataf, F. Rogier, and E. De Sturler. Optimal interface conditions for domain decomposition methods. Technical Report, École Polytech., Paris, 1994.
7. F. Nier. Remarques sur les algorithmes de décomposition de domaines. In *Seminaire: Équations aux Dérivées Partielles, 1998–1999*, Exp. No. IX, 26pp., Sémin. Équ. Dériv. Partielles. École Polytech., Palaiseau, 1999.

Optimized Schwarz Methods for Domains with an Arbitrary Interface

Shiu Hong Lui

Department of Mathematics, University of Manitoba, Winnipeg, Manitoba, Canada R3T 2N2, luish@cc.umanitoba.ca

1 Introduction

Optimized Schwarz methods form a class of domain decomposition methods for the solution of partial differential equations. Optimized Schwarz methods employ a first or higher order boundary condition along the artificial interface to accelerate its convergence. In the literature, analysis of optimized Schwarz methods rely on Fourier analysis and so the domains are restricted to be regular (rectangle or disk). By expressing the interface operator in terms of Poincare–Steklov operators, we are able to derive upper bounds of the spectral radius of the operator for Poisson-like problems for two essentially arbitrary subdomains. For a first order (Robin) boundary operator, an optimal choice of the parameter in the boundary operator leads to an upper bound of $1 - O(h^{1/2})$ of the spectral radius, where h is the discretization parameter. For a certain higher order boundary operator, a clever choice of the two parameters in the boundary operator leads to an upper bound of $1 - O(h^{1/4})$ of the spectral radius. These agree with the predicted rates for rectangular subdomains available in the literature and are also the observed rates in numerical simulations. This contribution summarizes the author's work in [11, 12].

Let Ω be a bounded domain in $\mathbf{R}^N$ with a smooth boundary. Suppose Ω is composed of two nonoverlapping open subdomains, that is, $\overline{\Omega} = \overline{\Omega}_1 \cup \overline{\Omega}_2$ with $\Omega_1 \cap \Omega_2 = \emptyset$. Assume that the artificial boundary $\Gamma = \overline{\Omega}_1 \cap \overline{\Omega}_2$ is non-trivial (non-zero measure in R^{N-1}) and is a smooth curve. We shall always assume that $\partial\Omega_i \setminus \Gamma$ is non-trivial for both $i = 1, 2$.

Recall the trace space

$$H_{00}^{1/2}(\Gamma) = \{v|_\Gamma,\ v \in H_0^1(\Omega)\}$$

with dual $H^{-1/2}(\Gamma)$. For $i = 1, 2$, let

$$V_i = \{v_i \in H^1(\Omega_i),\ v_i = 0 \text{ on } \partial\Omega_i \cap \partial\Omega\}.$$

Define the trace operators $T_i : V_i \to H_{00}^{1/2}(\Gamma)$ by

$$T_i v_i = v_i|_\Gamma, \qquad v_i \in V_i.$$

For simplicity, consider the model problem

$$-\triangle u = f \text{ on } \Omega, \qquad u = 0 \text{ on } \partial\Omega.$$

One candidate for the subdomain problem is

$$\begin{aligned} -\triangle u_i &= f \text{ on } \Omega_i, \\ u_i &= p \text{ on } \Gamma \end{aligned}$$

with $u_i \in V_i$ for some function $p \in H_{00}^{1/2}(\Gamma)$. Note that p is the correct function ($p = T_i u$) if

$$\frac{\partial u_1}{\partial \nu_1} + \frac{\partial u_2}{\partial \nu_2} = 0 \text{ on } \Gamma.$$

This is known as the transmission condition. Define $u_i = u_i^e + z_i$ where $u_i^e = \mathcal{H}_i p \in V_i$ is the harmonic extension of p:

$$\begin{aligned} -\triangle u_i^e &= 0 \text{ on } \Omega_i, \\ u_i^e &= p \text{ on } \Gamma \end{aligned}$$

and $z_i = \triangle_i^{-1} f$ where $\triangle_i$ is the Laplacian operator with domain $H_0^1(\Omega_i)$. Define the Poincare–Steklov operators $S_i : H_{00}^{1/2}(\Gamma) \to H^{-1/2}(\Gamma)$ by

$$S_i p = \frac{\partial \mathcal{H}_i p}{\partial \nu_i}$$

or by

$$\langle S_i p, q\rangle = \int_{\Omega_i} \nabla p^e \cdot \nabla q^e, \qquad \forall p, q \in H_{00}^{1/2}(\Gamma)$$

with $p^e = \mathcal{H}_i p,\ q^e = \mathcal{H}_i q$. In the above inner product, S_i is self-adjoint and positive definite. Hence the transmission condition can also be expressed as

$$(S_1 + S_2)u|_\Gamma = w \tag{1}$$

for some w.

2 First-Order Boundary Condition

In [10], the author defined the Schwarz sequence $\{u_i^{(n)} \in V_i,\ n \geq 0\}$ by

$$\begin{aligned} -\triangle u_i^{(n)} &= f \text{ on } \Omega_i, \\ \frac{\partial u_i^{(n)}}{\partial \nu_i} + \lambda u_i^{(n)} &= g_i^{(n)} \text{ on } \Gamma. \end{aligned} \tag{2}$$

Here λ is a positive constant. Noting that $\nu_1 = -\nu_2$ on Γ, the Robin data can be updated as

$$g_{3-i}^{(n+1)} = -\frac{\partial u_i^{(n)}}{\partial \nu_i} + \lambda u_i^{(n)} \text{ on } \Gamma, \qquad i = 1, 2.$$

The iteration can be started for any initial $g_i^{(0)} \in L^2(\Gamma)$. In practice, the choice $g_i^{(0)} = 0$ is convenient.

The following is an equivalent update ([2]):

$$g_{3-i}^{(n+1)} = 2\lambda u_i^{(n)} - g_i^{(n)} \text{ on } \Gamma, \qquad i = 1, 2. \tag{3}$$

Note that the subdomain computations can be carried out concurrently. Many authors have studied the convergence of this method and the choice of the optimal parameter. See [1, 12, 15] which are most pertinent to this paper.

The function g_2 can be eliminated in (3) to obtain the following equation for g_1:

$$\begin{aligned} &\Big[I - (I - 2\lambda(S_2 + \lambda)^{-1})(I - 2\lambda(S_1 + \lambda)^{-1})\Big] g_1 \\ &= 2\lambda b \equiv 2\lambda(T_2 z_2 - (I - 2\lambda(S_2 + \lambda)^{-1}) T_1 z_1). \end{aligned}$$

The operator for g_1 has alternative representations

$$\begin{aligned} &I - (S_2 + \lambda)^{-1}(S_2 - \lambda)(S_1 - \lambda)(S_1 + \lambda)^{-1} \\ &= (S_2 + \lambda)^{-1}\big((S_2 + \lambda)(S_1 + \lambda) - (S_2 - \lambda)(S_1 - \lambda)\big)(S_1 + \lambda)^{-1} \\ &= 2\lambda(S_2 + \lambda)^{-1}(S_1 + S_2)(S_1 + \lambda)^{-1}. \end{aligned}$$

Thus the above equation for g_1 is equivalent to

$$(S_2 + \lambda)^{-1}(S_1 + S_2)(S_1 + \lambda)^{-1} g_1 = b.$$

Recognizing that $(S_1 + \lambda)^{-1} g_1 = T_1(u - z_1)$ where u is the exact solution of the global Poisson equation and (g_1, g_2) is the solution of (3), we see that Lions' method is an iterative method which solves (1) using the left preconditioner $(S_2 + \lambda)^{-1}$.

Lions' method is equivalent to the following iterative method

$$g_1^{(n+1)} = \mathcal{G}_h g_1^{(n-1)} + b \tag{4}$$

to solve for the discrete counterpart of the boundary function g_1 where

$$\begin{aligned} \mathcal{G}_h &\equiv (I - 2\lambda(S_{2,h} + \lambda)^{-1})(I - 2\lambda(S_{1,h} + \lambda)^{-1}) \\ &= (S_{2,h} + \lambda)^{-1}(S_{2,h} - \lambda)(S_{1,h} + \lambda)^{-1}(S_{1,h} - \lambda). \end{aligned}$$

Here $S_{i,h}$ is a finite element discretization of S_i.

For a square matrix A, let the spectral radius of A be denoted by $\rho(A)$. The convergence of the iteration (4) depends on $\rho(\mathcal{G}_h)$ which will be analyzed below. In the following, $|\cdot|$ denotes the two-norm. We shall use c, c_1, c_2 to denote positive constants whose values may differ in different occurrences.

The analysis for the upper bound of $\rho(\mathcal{G}_h)$ is identical to that for the ADI method to solve PDEs. This is because $\mathcal{G}_h$ has the same form as the operator in the ADI method. Note

$$\rho(\mathcal{G}_h) \leq |\mathcal{G}_h| \leq |(S_{1,h}+\lambda)^{-1}(S_{1,h}-\lambda)|\,|(S_{2,h}+\lambda)^{-1}(S_{2,h}-\lambda)|,$$

Since $S_{1,h}$ and $S_{2,h}$ are symmetric and their eigenvalues have the same asymptotic behaviour, it is not difficult to show

Theorem 1.

$$\rho(\mathcal{G}_h) \leq \begin{cases} 1-c_1\lambda h, & \lambda \leq h^{-1/2}; \\ 1-c_2\lambda^{-1}, & \lambda \geq h^{-1/2}. \end{cases} \tag{5}$$

In case $\lambda = O(h^{-1/2})$, *then* $\rho(\mathcal{G}_h) \leq 1 - ch^{1/2}$.

A lower bound for $\rho(\mathcal{G}_h)$ is considerably more difficult to establish than an upper bound. In fact, we have only been able to obtain a lower bound for λ in special intervals. For $\lambda = h^s$ with $s \in (-\infty,-1)\cup(0,\infty)$, the upper bound established in the theorem is actually sharp. In the more interesting range $s \in [-1,0]$, the analysis is more complicated because $\mathcal{G}_h$ is a product of two symmetric indefinite matrices. We conjecture that the bounds in (5) are sharp for $s \in [-1,0]$ as well.

We conclude this section by mentioning that the analysis has been extended to the case of PDEs with discontinuous coefficients. See [3].

3 Higher-Order Boundary Condition

One popular optimized Schwarz method using a second order boundary condition along the artificial interface is

$$-\frac{d^2u_i}{d\tau^2} + \eta\frac{du_i}{d\nu_i} + \lambda u_i = g_i \text{ on } \Gamma$$

where η and λ are positive parameters and τ is a unit tangent vector along Γ. In the literature, see [4, 5, 6, 7, 8, 9, 13, 14], for instance, Fourier analysis is used to analyze the convergence of the schemes, which means that the theory is applicable only to regular (rectangular) subdomains.

For $i = 1, 2$, the subdomain problems are

$$\begin{aligned} -\triangle u_i^{(n)} &= f \text{ on } \Omega_i, \\ -\frac{\partial^2 u_i^{(n)}}{\partial\tau^2} + \eta\frac{\partial u_i^{(n)}}{\partial\nu_i} + \lambda u_i^{(n)} &= g_i^{(n)} \text{ on } \Gamma \end{aligned} \tag{6}$$

where $g_i^{(n)}$ is some given function. Henceforth, we shall assume $f \equiv 0$. Unfortunately, we are also unable to prove a rate of convergence of $g_i^{(n)}$ to zero in non-rectangular geometry. Instead, we propose a different boundary condition for which a spectral radius estimate $1 - O(h^{1/4})$ can be proven for a general class of domains.

This is the same estimate as that for (6) for rectangular domains which is available in the literature.

We now give a heuristic derivation of our new boundary condition. Along Γ,

$$0 = f = \triangle u = \frac{\partial^2 u}{\partial \nu^2} + \frac{\partial^2 u}{\partial \tau^2} + Lu$$

where $L = \nabla \cdot \tau \, \partial_\tau + \nabla \cdot \nu \, \partial_\nu$ is a linear first order differential operator. We shall be taking $\eta = O(h^{-3/4})$ and $\lambda = O(h^{-1})$ where h is the discretization parameter and thus the term containing L will be insignificant. Ignoring it, (6) can be approximated as

$$\frac{\partial^2 u_i^{(n)}}{\partial \nu_i^2} + \eta \frac{\partial u_i^{(n)}}{\partial \nu_i} + \lambda u_i^{(n)} = g_i^{(n)} \text{ on } \Gamma. \tag{7}$$

A natural update for the boundary function $g_i^{(n)}$ is

$$g_{3-i}^{(n+1)} = g_i^{(n)} - 2\eta \frac{\partial u_i^{(n)}}{\partial \nu_i}. \tag{8}$$

To see this, note that $\nu_1 = -\nu_2$ and

$$\begin{aligned} g_{3-i}^{(n+1)} &= \frac{\partial^2 u_{3-i}^{(n+1)}}{\partial \nu_{3-i}^2} + \eta \frac{\partial u_{3-i}^{(n+1)}}{\partial \nu_{3-i}} + \lambda u_{3-i}^{(n+1)} \\ &\equiv \frac{\partial^2 u_i^{(n)}}{\partial \nu_i^2} - \eta \frac{\partial u_i^{(n)}}{\partial \nu_i} + \lambda u_i^{(n)} \\ &= g_i^{(n)} - 2\eta \frac{\partial u_i^{(n)}}{\partial \nu_i}. \end{aligned}$$

We next approximate the second normal derivative in (7) by S_i^2, leading to the new boundary condition

$$(S_i^2 + \eta S_i + \lambda) T_i u_i^{(n)} = g_i^{(n)} \text{ on } \Gamma.$$

We assume that

$$S_i T_i u_i^{(n)} \in H_{00}^{1/2}(\Gamma) \tag{9}$$

so that $S_i^2 T_i u_i^{(n)} \in H^{-1/2}(\Gamma)$. Two examples of Γ where the assumption (9) holds are one side of a rectangle and an arc of a circle, provided that $u_i^{(n)}$ is sufficiently smooth. For these two cases, S_i can be worked out analytically and it can be seen that S_i^2 and $\partial^2/\partial \nu_i^2$ differ when acting upon low order modes. Their difference goes to zero as the order of modes goes to infinity. It is in this sense that S_i^2 approximates the second normal derivative and is the reason why the algorithms employing the two boundary conditions have similar convergence rates. By writing an equivalent form

$$\frac{\partial^2}{\partial \nu_i^2} + \eta S_i + \lambda + L$$

of the boundary operator (6), we clearly see the two approximations involved in the proposed boundary operator $S_i^2 + \eta S_i + \lambda$: replacement of the second normal derivative by S_i^2 and the removal of L.

Define, for $i = 1, 2$,

$$\begin{aligned} -\triangle u_i^{(n)} &= 0 \text{ on } \Omega_i \\ (S_i^2 + \eta S_i + \lambda) T_i u_i^{(n)} &= g_i^{(n)}. \end{aligned} \tag{10}$$

The parameters η and λ are positive. The update (8) is still applicable here and can be written as

$$g_{3-i}^{(n+1)} = g_i^{(n)} - 2\eta S_i T_i u_i^{(n)}.$$

Since $T_i u_i^{(n)} = (S_i^2 + \eta S_i + \lambda)^{-1} g_i^{(n)}$, the update for the boundary function becomes

$$g_{3-i}^{(n+1)} = g_i^{(n)} - 2\eta S_i (S_i^2 + \eta S_i + \lambda)^{-1} g_i^{(n)}.$$

Eliminate $g_2^{(n)}$ from the above to obtain $g_1^{(n+1)} = \mathcal{K} g_1^{(n-1)}$ where

$$\mathcal{K} = \Big(I - 2\eta S_2 (S_2^2 + \eta S_2 + \lambda)^{-1}\Big)\Big(I - 2\eta S_1 (S_1^2 + \eta S_1 + \lambda)^{-1}\Big). \tag{11}$$

The discrete iteration is

$$g_1^{(n+1)} = \mathcal{K}_h g_1^{(n)} \tag{12}$$

where $\mathcal{K}_h$ denotes a finite element discretization of $\mathcal{K}$. Convergence of this iteration depends on $\rho(\mathcal{K}_h)$. If $\rho(\mathcal{K}_h) < 1$, then $g_1^{(n)} \to 0$, the exact solution. Since $\rho(\mathcal{K}_h) \leq |\mathcal{K}_h|$,

$$\rho(\mathcal{K}_h) \leq |I - 2\eta S_{2,h} (S_{2,h}^2 + \eta S_{2,h} + \lambda)^{-1}| \, |I - 2\eta S_{1,h} (S_{1,h}^2 + \eta S_{1,h} + \lambda)^{-1}|$$

with the matrices on the right-hand side symmetric. The proof of the following theorem appears in [11].

Theorem 2. *Let $\lambda = O(h^{-1})$ and $\eta = O(h^{-3/4})$. Then $\rho(\mathcal{K}_h) \leq 1 - O(h^{1/4})$.*

The above theorem gives an upper bound of the spectral radius. As before, a lower bound is much more difficult to establish. The following are some partial results. Suppose $\eta < O(1)$. Then

$$\rho(\mathcal{K}_h) = \begin{cases} 1 - O(\eta h), & \lambda \leq O(h^{-1}); \\ 1 - O(\eta \lambda^{-1}), & \lambda \geq O(h^{-1}). \end{cases}$$

Suppose $\eta > O(h^{-1})$ and $\lambda < O(\eta)$. Then

$$\rho(\mathcal{K}_h) = \begin{cases} 1 - O(\eta^{-1}), & \lambda < O(1); \\ 1 - O(\eta^{-1} \lambda h), & \lambda > O(h^{-2}). \end{cases}$$

We give one MATLAB numerical experiment. Let the domain be the rectangle $[0, 1.6] \times [0, 1]$ and the artificial interface be the line $y = x - 0.2$. Hence the two subdomains are trapezoids. Using a simple finite difference scheme, the result is shown

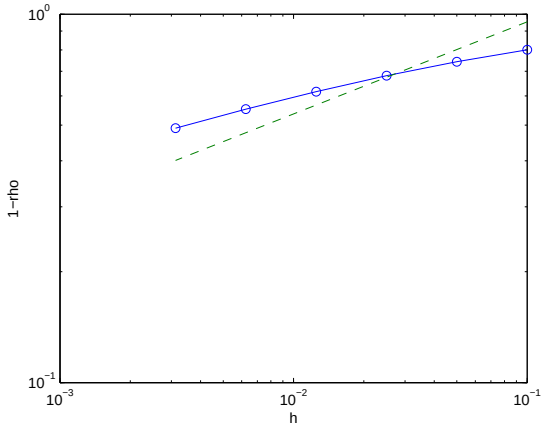


Fig. 1. *Solid line* is a plot of $1 - \rho(\mathcal{K}_h)$ versus h for two trapezoidal subdomains while the *dashed line* is a plot of $1 - O(h^{1/4})$.

in Fig. 1. Observe that for larger values of h, the spectral radius is actually better than the prediction $1 - O(h^{1/4})$. However, the spectral radius seems to approach the predicted rate for smaller values of h. For other numerical results, see [11].

There are a number of mathematical questions about the new boundary condition which have not been answered. Although the discrete iteration (12) is well defined and convergent, it remains to show well-posedness at the continuous level for the boundary condition (10). Also, the geometric meaning of the assumption (9) requires investigation. While we have not been able to establish a convergence rate for (6) on arbitrary domains, it is hoped that the present analysis gives some new insight to the convergence of (6).

References

1. V.I. Agoshkov and V.I. Lebedev. Generalized Schwarz algorithm with variable parameters. *Soviet J. Numer. Anal. Math. Modelling*, 5:1–26, 1990.
2. Q. Deng. An analysis for a nonoverlapping domain decomposition iterative procedure. *SIAM J. Sci. Comput.*, 18:1517–1525, 1997.
3. O. Dubois and S.H. Lui. Convergence estimates for an optimized Schwarz method for pdes with discontinuous coefficients. *Numer. Algorithm*, 51:115–131, 2009.
4. B. Engquist and H.K. Zhao. Absorbing boundary conditions for domain decomposition. *Appl. Numer. Math*, 27:341–365, 1998.
5. E. Flauraud, F. Nataf, and F. Willien. Optimized interface conditions in domain decomposition methods for problems with extreme contrasts in the coefficients. *J. Comput. Appl. Math.*, 189:539–554, 2006.
6. M.J. Gander. Optimized Schwarz methods. *SIAM J. Numer. Anal.*, 44:699–731, 2006.
7. M.J. Gander, F. Magoules, and F. Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24:38–60, 2002.

8. M.J. Gander and G.H. Golub. A nonoverlapping optimized Schwarz method which converges with arbitrary weak dependence on h. In I. Herrera, D.E. Keyes, O.B. Widlund, and R. Yates, editors, *Fourteen International Conference on Domain Decomposition Methods in Science and in Engineering*, pp. 281–287, DDM.org, Mexico, 2003.
9. M.J. Gander, L. Halpern, and F. Nataf. Optimized Schwarz methods. In T. Chan, T. Kako, H. Kawarada, and O. Pironneau, editors, *Twelfth International Conference on Domain Decomposition Methods in Science and in Engineering*, pp. 15–28, DDM.org, Japan, 2001.
10. P. L. Lions. On the Schwarz alternating method III. In T.F. Chan, R. Glowinski, J. Periaux, and O. Widlund, editors, *Third International Symposium on Domain Decomposition Methods*, pp. 202–223, SIAM, Philadelphia, 1990.
11. S.H. Lui. Convergence estimates for an higher-order optimized Schwarz method for domains with an arbitrary interface. *Preprint*, 2009.
12. S.H. Lui. A Lions nonoverlapping domain decomposition method for domains with an arbitray interface. *IMA J. Numer. Anal.*, 29(2) 332–3249, 2009.
13. F. Nataf. Convergence rate of some domain decomposition methods for overlapping and nonoverlapping subdomains. *Numer. Math.*, 75:357–377, 1997.
14. F. Nataf. Recent developments in optimized Schwarz methods. In O.B. Widlund and D.E. Keyes, editors, *Sixteen International Conference on Domain Decomposition Methods in Science and in Engineering*, pp. 115–125, Springer, New York, NY, 2007.
15. L. Qin and X. Xu. On a parallel Robin-type nonoverlapping domain decomposition method. *SIAM J. Numer. Anal.*, 44:2539–2558, 2006.

Can the Discretization Modify the Performance of Schwarz Methods?

Victorita Dolean[1] and Martin J. Gander[2]

[1] Laboratoire J.-A. Dieudonné, University de Nice Sophia-Antipolis, UMR CNRS 6621, 06108 Nice Cedex 02, France, dolean@unice.fr
[2] Section de Mathématiques, Université de Genève, 1211 Genève 4, Switzerland, martin.gander@unige.ch

Summary. Schwarz domain decomposition methods can be analyzed both at the continuous and discrete level. For consistent discretizations, one would naturally expect that the discretized method performs as predicted by the continuous analysis. We show in this short note for two model problems that this is not always the case, and that the discretization can both increase and decrease the convergence speed predicted by the continuous analysis.

1 Introduction

Classical Schwarz methods have been analyzed historically both at the continuous and the discrete level, see for example [7, 8, 9, 10, 11] and references therein for continuous analysis, [5, 12] and references therein for analysis at the discrete level. Over the last decade, optimized Schwarz methods have been extensively developed at the continuous level. These methods converge significantly faster than the classical Schwarz methods, see for example [6], and references therein. More recently, Schwarz methods have also been developed for systems of partial differential equations, see for example [4] for Euler equations, [2] for the Cauchy–Riemann equations, or [1, 3] for Maxwell's equations, and it was observed in two particular cases that a discretized Schwarz method converged faster than predicted by the continuous analysis. The purpose of this note is to explain this observation for the case of the Cauchy–Riemann equations, and also to reveal a previously not observed discrepancy for the case of the positive definite Helmholtz operator, $\eta - \Delta$, $\eta > 0$ (note that we do not treat the indefinite Helmholtz operator, where $\eta < 0$).

2 The Cauchy–Riemann Equations

Classical and optimized Schwarz methods have been analyzed in [2] at the continuous level for the Cauchy Riemann equations,

$$\mathcal{L}\mathbf{u} := \sqrt{\eta}\mathbf{u} + A\partial_x\mathbf{u} + B\partial_y\mathbf{u} = \mathbf{f}, \qquad A = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \tag{1}$$

and it was observed in the classical Schwarz case that the discretized algorithms converged faster than predicted by the continuous analysis. The finite volume discretization used in these experiments was on a Cartesian mesh with mesh points $x_{lm} = (l\Delta x, m\Delta y)$, $l, m \in \mathbb{Z}$, namely

$$\begin{aligned} L\mathbf{u}_{l,m} &:= \begin{pmatrix} L_1\mathbf{u}_{l,m} \\ L_2\mathbf{u}_{l,m} \end{pmatrix} = \begin{pmatrix} f_{l,m} \\ g_{l,m} \end{pmatrix} =: \mathbf{f}_{l,m}, \\ L_1\mathbf{u}_{l,m} &:= \sqrt{\eta}u_{l,m} + (-D_x^+ - \tfrac{D_y^+ - D_y^-}{2})u_{l,m} + \tfrac{D_y^+ + D_y^-}{2}v_{l,m}, \\ L_2\mathbf{u}_{l,m} &:= \sqrt{\eta}v_{l,m} + (D_x^- - \tfrac{D_y^+ - D_y^-}{2})v_{l,m} + \tfrac{D_y^+ + D_y^-}{2}u_{l,m}, \end{aligned} \tag{2}$$

where $D_x^\pm$ and $D_y^\pm$ are the usual finite difference operators in x and y directions. We consider now a decomposition of $\Omega = \mathbb{R}^2$ into two subdomains $\Omega_1 = (-\infty, a) \times \mathbb{R}$ and $\Omega_2 = (b, \infty) \times \mathbb{R}$. In the interior of Ω_1 the Eq. (2) is verified for all $l < l_1$ and for Ω_2, it is verified for all $l > l_2$. A discrete Schwarz algorithm with general transmission conditions is

$$\begin{aligned} L\mathbf{u}_{l,m}^{1,n} &= \mathbf{f}_{l,m}, \quad l < l_1, & L\mathbf{u}_{l,m}^{2,n} &= \mathbf{f}_{l,m}, \quad l > l_2, \\ L_2\mathbf{u}_{l_1,m} &= g_{l_1,m}, & L_1\mathbf{u}_{l_2,m} &= f_{l_2,m}, \\ u_{l_1,m}^{1,n} + S^1 v_{l_1,m}^{1,n} &= u_{l_1,m}^{2,n-1} + S^1 v_{l_1,m}^{2,n-1}, & v_{l_2,m}^{2,n} + S^2 u_{l_2,m}^{2,n} &= v_{l_2,m}^{1,n-1} + S^2 u_{l_2,m}^{1,n}. \end{aligned} \tag{3}$$

where l_1, l_2 are the indices of the interface points, and $S^{1,2}$ are finite difference operators that may contain parameters chosen in order to obtain better convergence than with the classical algorithm. If only information following the characteristics are exchanged, $S^{1,2} \equiv 0$, we obtain the classical Schwarz algorithm, see [2].

To simplify the analysis, we use the same discretization step in the x and y direction, $h := \Delta x = \Delta y$. We denote the overlap parameter by $\delta := l_1 - l_2$, and use a discrete Fourier transform to study convergence properties of algorithm (3). Since we study the evolution of the error, it is sufficient to study the homogeneous counterpart of (3), and we look for the solutions of the form

$$\mathbf{u}_{l,m}^{j,n} = \sum_k \alpha^{j,n}(k) e^{lh\lambda(k)} e^{ikmh} \begin{pmatrix} \hat{u}_{k,m}^{j,n} \\ \hat{v}_{k,m}^{j,n} \end{pmatrix}, \tag{4}$$

where $j = 1, 2$ denotes the subdomain index and n the iteration number of the Schwarz algorithm. At each iteration and in each subdomain, the iterates satisfy for each discrete frequency k the system of equations

$$\begin{aligned} \sqrt{\eta}\hat{u}_{k,m} - \tfrac{e^{\lambda(k)h}-1}{h}\hat{u}_{k,m} + \tfrac{2-e^{ikh}-e^{-ikh}}{2h}\hat{u}_{k,m} + \tfrac{e^{ikh}-e^{-ikh}}{2h}\hat{v}_{k,m} &= 0, \\ \sqrt{\eta}\hat{v}_{k,m} + \tfrac{1-e^{-\lambda(k)h}}{h}\hat{v}_{k,m} + \tfrac{2-e^{ikh}-e^{-ikh}}{2h}\hat{v}_{k,m} + \tfrac{e^{ikh}-e^{-ikh}}{2h}\hat{u}_{k,m} &= 0. \end{aligned} \tag{5}$$

If we denote by $\phi := \frac{e^{\lambda(k)h}-1}{h}$ and by $\hat{w} := \frac{\hat{v}_{k,m}}{\hat{u}_{k,m}}$, we obtain from the first equation of (5) that

$$\phi = \sqrt{\eta} + \frac{a_k h}{2} + b_k \hat{w}, \tag{6}$$

where $b_k = \frac{i\sin(kh)}{h} = ik + \mathcal{O}(h)$ and $a_k = \frac{2(1-\cos(kh))}{h^2} = k^2 + \mathcal{O}(h)$ are the symbols of the discrete first and second order derivative with respect to y. Replacing this result into the second equation of (5) we obtain an equation for $\hat{w}$,

$$\begin{aligned} b_k\left(\left(\sqrt{\eta}+\tfrac{a_k h}{2}\right)h+1\right)\hat{w}^2 + \left(\left(\sqrt{\eta}+\tfrac{a_k h}{2}\right)\left(\left(\sqrt{\eta}+\tfrac{a_k h}{2}\right)h+2\right)+b_k^2 h\right)\hat{w} \\ + b_k\left(\left(\sqrt{\eta}+\tfrac{a_k h}{2}\right)h+1\right) = 0. \end{aligned}$$

This equation has solutions $\hat{w}_{1,2}$, which give two corresponding values of $\phi_{1,2}$ with opposite signs, whose asymptotic behavior for h small is

$$\phi_{1,2}(k,h) = \pm\sqrt{\eta + k^2} + \mathcal{O}(h).$$

Since subdomain solutions need to remain bounded, they must be of the form

$$\mathbf{u}_{l,m}^{j,n} = \sum_k \alpha^{j,n}(k)(\phi_j h + 1)^l e^{ikmh} \begin{pmatrix} \hat{u}_{k,m}^{j,n} \\ \hat{v}_{k,m}^{j,n} \end{pmatrix}. \tag{7}$$

If we denote by $\sigma_{1,2}$ the Fourier symbols of the operators $S^{1,2}$, and insert (7) into the interface conditions of algorithm (3), we obtain for each frequency k

$$\begin{aligned} \alpha^{1,n}(k)(\hat{u}_{k,m}^1 + \sigma_1 \hat{v}_{k,m}^1)(\phi_1 h+1)^{l_1} &= \alpha^{2,n-1}(k)(\hat{u}_{k,m}^2 + \sigma_1 \hat{v}_{k,m}^2)(\phi_2 h+1)^{l_1}, \\ \alpha^{2,n}(k)(\hat{v}_{k,m}^2 + \sigma_2 \hat{u}_{k,m}^2)(\phi_2 h+1)^{l_2} &= \alpha^{1,n-1}(k)(\hat{v}_{k,m}^1 + \sigma_2 \hat{u}_{k,m}^1)(\phi_1 h+1)^{l_2}. \end{aligned}$$

Taking into account that $\hat{w} = \frac{\hat{v}_{k,m}}{\hat{u}_{k,m}}$ and using (6), the convergence factor of algorithm (3) is

$$\begin{aligned} \rho(k,\eta,\delta,h) &= \left(\frac{\alpha^{2,n}}{\alpha^{2,n-2}}\right)^{\frac{1}{2}} = \left(\frac{1+\sigma_1\hat{v}_2}{1+\sigma_1\hat{v}_1}\cdot\frac{\sigma_2+\hat{v}_1}{\sigma_2+\hat{v}_2}\right)^{\frac{1}{2}}\cdot\left(\frac{\phi_2 h+1}{\phi_1 h+1}\right)^{\frac{\delta}{2}} \\ &= \left(\frac{b_k+\sigma_1\left(\phi_2-\sqrt{\eta}-\frac{a_k h}{2}\right)}{b_k+\sigma_1\left(\phi_1-\sqrt{\eta}-\frac{a_k h}{2}\right)}\cdot\frac{b_k\sigma_2+\left(\phi_1-\sqrt{\eta}-\frac{a_k h}{2}\right)}{b_k\sigma_2+\left(\phi_2-\sqrt{\eta}-\frac{a_k h}{2}\right)}\right)^{\frac{1}{2}}\cdot\left(\frac{\phi_2 h+1}{\phi_1 h+1}\right)^{\frac{\delta}{2}}. \end{aligned} \tag{8}$$

The maximum $\rho_{\max}$ of this convergence factor over all relevant frequencies $k \in [0, k_{\max}]$, with the estimate $k_{\max} = \frac{\pi}{h}$, determines the overall contraction factor of the method, and hence the rate of convergence of the associated algorithm. Different classes of interface conditions were studied at the continuous level in [2]:

Case 1: $\sigma_1 = \sigma_2 = 0$. This case corresponds to the classical Schwarz algorithm which exchanges characteristic information at the interfaces.

Proposition 1. *Let $\sigma_1 = \sigma_2 = 0$. In the non-overlapping case of algorithm (3) , $\delta = 0$, the convergence factor attains its maximum for h small at $k_b = 2^{\frac{1}{2}}\cdot 3^{-\frac{1}{4}}\eta^{\frac{1}{8}}\cdot h^{-\frac{3}{4}}$, which leads to the overall contraction factor*

$$\rho_{max} := \rho(k_b,\eta,0,h) = 1 - 2^{\frac{3}{2}}\cdot 3^{-\frac{3}{4}}\eta^{\frac{3}{8}}h^{\frac{3}{4}} + \mathcal{O}(h).$$

In the overlapping case of algorithm (3) , we have for

$$\begin{aligned} \delta = 1 &: k_b = \eta^{\frac{1}{4}}\cdot h^{-\frac{1}{2}}, \qquad \rho_{max} = 1 - 2\eta^{\frac{1}{4}}\cdot h^{\frac{1}{2}} + \mathcal{O}(h), \\ \delta = 2 &: k_b = \eta^{\frac{1}{4}}\cdot 2^{-\frac{1}{2}}\cdot h^{-\frac{1}{2}}, \ \rho_{max} = 1 - 2^{\frac{3}{2}}\eta^{\frac{1}{4}}\cdot h^{\frac{1}{2}} + \mathcal{O}(h). \end{aligned} \tag{9}$$

Remark 1. In the non-overlapping case, the convergence factor predicted by the continuous analysis in [2] was $1-\mathcal{O}(h)$, but faster convergence was observed numerically, a gap closed by the present analysis. In the overlapping case however, for $\delta = 1, 2$ and probably also bigger δ, the convergence factors from the discrete and continuous analysis have the same asymptotic behavior, see [2].

Case 2: $\sigma_1 = \frac{b_k}{\sqrt{\eta}+p}$, $\sigma_2 = \frac{\sqrt{\eta}-p}{b_k}$, a case with one parameter $p > 0$ to choose for best performance. Since b_k is the discrete symbol of the tangential derivative, this case corresponds to the optimized algorithm where local operators are used in the transmission conditions expressed with first order derivatives. Note that even if we have b_k in the denominator, it suffices to multiply both sides of the transmission conditions with b_k in order to obtain local operators.

Proposition 2. *Let $\sigma_1 = \frac{b_k}{\sqrt{\eta}+p}$ and $\sigma_2 = \frac{\sqrt{\eta}-p}{b_k}$. In the non-overlapping case of algorithm (3), $\delta = 0$, the optimized parameter p^* is for h small solution of*

$$\rho(k_1(p), \eta, 0, h, p) = \rho(k_{max}, \eta, 0, h, p), \tag{10}$$

where $k_1(p)$ is a maximum of ρ, and we have the asymptotic result

$$k_1 = \frac{C_{k_1}}{h}, \quad p^* = \frac{Cp}{\sqrt{h}}, \quad \rho_{max} = 1 - \frac{1}{4C_p} \cdot (3C_p^2 + 8\sqrt{\eta})\sqrt{h} + \mathcal{O}(h).$$

The constants C_{k_1} and C_p can be explicitly computed: if θ denotes the real root of $6x^3 - 20x^2 + 19x - 3 = 0$, then we get $C_{k_1} = \arccos(\theta) = 1.373593$, and

$$\begin{aligned} Cp = {} & 2/(-3 \cdot \cos(C_{k_1})^3 + 6 \cdot \cos(C_{k_1})^2 + 3 \cdot \cos(C_{k_1}) - 6 \\ & +16 \cdot ((-1 + \cos(C_{k_1})) \cdot (3 \cdot \cos(C_{k_1}) - 5))^{1/2}) \cdot (-(6 \cdot \cos(C_{k_1})^3 \\ & -12 \cdot \cos(C_{k_1})^2 - 6 \cdot \cos(C_{k_1}) + 12 \\ & -32 \cdot ((-1 + \cos(C_{k_1})) \cdot (3 \cdot \cos(C_{k_1}) - 5))^{1/2}) \cdot \eta^{1/2} \cdot (\cos(C_{k_1})^3 \\ & -2 \cdot \cos(C_{k_1})^2 - \cos(C_{k_1}) + 2))^{1/2} \\ = {} & 0.7460898 \cdot \eta^{\frac{1}{4}}. \end{aligned}$$

Proposition 3. *Let $\sigma_1 = \frac{b_k}{\sqrt{\eta}+p}$ and $\sigma_2 = \frac{\sqrt{\eta}-p}{b_k}$. In the overlapping case, $\delta = 1$, the optimized parameter p^* is for h small solution of the equation*

$$\rho(k_1(p), \eta, \delta, h, p) = \rho(k_{min}, \eta, \delta, h, p), \tag{11}$$

where again $k_1(p)$ is a maximum of ρ, and $k_{min} \geq 0$ is the minimum frequency on the interface, and we have asymptotically

$$p^* = 2^{-\frac{1}{3}} \cdot (k_{min}^2 + \eta)^{\frac{1}{3}} \cdot h^{-\frac{1}{3}}, \quad k_1 = 2^{\frac{1}{3}} \cdot (k_{min}^2 + \eta)^{\frac{1}{6}} \cdot h^{-\frac{2}{3}},$$

$$\rho_{max} = 1 - 4 \cdot (\eta + k_{min}^2)^{\frac{1}{6}} \cdot 2^{\frac{1}{3}} \cdot h^{\frac{1}{3}} + \mathcal{O}(h).$$

The same asymptotic behavior is also obtained for bigger overlap, $\delta > 1$.

Remark 2. In both Propositions 2 and 3, the asymptotic analysis of the discretized algorithm presented here and the continuous algorithm from [2] predict the same asymptotic performance.

Case 3: $\sigma_1 = \sigma_2 = \sigma = \frac{b_k}{\sqrt{\eta}+p}$, where we can again choose $p > 0$ for best performance.

Proposition 4. *The optimized parameter p^* is for h small solution of the equation*

$$\rho(k_1(p), \eta, \delta, h, p) = \rho(k_2(p), \eta, \delta, h, p), \tag{12}$$

where $k_1(p)$ and $k_2(p)$ are maxima of ρ. In the case $\delta \leq 3$, which means no or small overlap (at most 3 mesh cells), we have the asymptotic result

$$p^* = \frac{Cp}{h}, \quad k_1 = \frac{C_{k_1}}{\sqrt{h}}, \quad k_2 = \frac{C_{k_2}}{h},$$

$$\rho_{\max} = 1 - 2\frac{2C_{k_1}^2 + C_p\sqrt{\eta} + \delta C_{k_1}^2 C_p}{C_p C_{k_1}}\sqrt{h} + \mathcal{O}(h).$$

In the case with more overlap, $\delta \geq 4$, we obtain for h small

$$p^* = \frac{Cp}{\sqrt{h}}, \quad k_1 = \frac{C_{k_1}}{h^{\frac{1}{4}}}, \quad k_2 = \frac{C_{k_2}}{h^{\frac{3}{4}}},$$

$$\rho_{\max} = 1 - 2\frac{2C_{k_1}^2 + C_p\sqrt{\eta}}{C_p C_{k_1}}h^{\frac{1}{4}} + \mathcal{O}(h^{\frac{1}{2}}).$$

The constants can again be computed: for example for the zero or small overlapping case, we obtain

$$\begin{aligned} \delta = 0 &: C_p = 0.383205,\ C_{k_1} = 0.437724\eta^{\frac{1}{4}},\ C_{k_2} = 2.29295, \\ \delta = 1 &: C_p = 0.068781,\ C_{k_1} = 0.182338\eta^{\frac{1}{4}},\ C_{k_2} = 2.71717, \end{aligned} \tag{13}$$

and for a case with bigger overlap, $\delta = 4$, we get

$$C_p = \frac{1}{2}\eta^{\frac{1}{4}},\ C_{k_1} = \frac{1}{2}\eta^{\frac{3}{8}},\ C_{k_2} = \frac{1}{2}\eta^{\frac{1}{8}}. \tag{14}$$

We observed that for $\delta > 4$ the factor one half in the constants (14) is replaced by a factor that becomes smaller and smaller, as δ becomes larger.

Remark 3. Again there is a substantial difference between the continuous analysis from [2] and the discrete analysis presented here: the continuous analysis predicted the convergence factor $1 - \mathcal{O}(h^{\frac{1}{3}})$ without overlap, and $1 - \mathcal{O}(h^{\frac{1}{4}})$ with overlap.

Such differences are not only restricted to the somewhat exotic example of the Cauchy–Riemann equations, they were also observed when the classical Schwarz method is applied to Maxwell's equations, see [3], and we will show in the next section that even in the case of simple positive definite scalar partial differential equations such differences can occur.

3 The Positive Definite Helmholtz Equation

Optimized Schwarz methods have been analyzed thoroughly for the positive definite Helmholtz equation at the continuous level in [6], and extensive numerical tests have been presented which illustrate the performance predicted by the continuous analysis. We show in this section that there are certain, quite natural discretizations which can lead again to differences in the performance.

We use the same Cartesian mesh on $\Omega = \mathbb{R}^2$ with mesh points $x_{l,m} = (l\Delta x, m\Delta y)$, $l, m \in \mathbb{Z}$, and we consider the five point finite difference discretization of the positive definite Helmholtz equation $(\eta - \Delta)u = f$,

$$Lu_{l,m} := \left(\eta - D_x^+ D_x^- - D_y^+ D_y^-\right) u_{l,m} = f_{l,m}. \tag{15}$$

With the same decomposition as in Sect. 2, a general discrete Schwarz algorithm applied to (15) is

$$\begin{array}{llll} Lu_{l,m}^{1,n} = f_{l,m}, & l < l_1, & Lu_{l,m}^{2,n} = f_{l,m}, & l > l_2, \\ B_1 u_{l_1,m}^{1,n} = B_1 u_{l_1,m}^{2,n-1}, & j \in \mathbb{Z}, & B_2 u_{l_2,m}^{2,n} = B_2 u_{l_2,m}^{1,n-1}, & j \in \mathbb{Z}, \end{array} \tag{16}$$

where $B_{1,2}$ denote the discrete transmission conditions (Dirichlet or Robin). We set again $h := \Delta x = \Delta y$ and $\delta := l_1 - l_2$ for the overlap. Using a discrete Fourier analysis in the y direction, one can show the following results:

Proposition 5. *For Dirichlet transmission conditions, $B_{1,2} = Id$, and one mesh size overlap, $\delta = 1$, the asymptotic convergence factor of algorithm (16) for h small is given by*

$$\rho_{\max} = 1 - \sqrt{\eta}h + \mathcal{O}(h^2),$$

which is identical to the result obtained from a continuous analysis.

Proposition 6. *For Robin transmission conditions discretized by one-sided finite differences, $B_1 := D_x^- + p$ and $B_2 := D_x^+ - p$, the optimized Robin parameter and asymptotic convergence factor of algorithm (16) for h small are given by*

$$\begin{array}{l} \delta = 1\text{: } p^* = 2^{\frac{1}{4}}\eta^{\frac{1}{4}}h^{-\frac{1}{2}}, \quad \rho_{\max} = 1 - \eta^{\frac{1}{4}}2^{\frac{3}{4}}\sqrt{h} + \mathcal{O}(h), \\ \delta = 2\text{: } p^* = 2^{-\frac{1}{3}}\eta^{\frac{1}{3}}h^{-\frac{1}{3}}, \rho_{\max} = 1 - 2\eta^{\frac{1}{6}}2^{\frac{1}{3}}h^{\frac{1}{3}} + \mathcal{O}(h^{\frac{2}{3}}). \end{array} \tag{17}$$

Remark 4. In the case $\delta = 2$ with overlap $2h$, and one-sided finite difference discretization of the normal derivative, the asymptotic performance of the discretized algorithm is as predicted by the continuous analysis, see [6]. However with $\delta = 1$, which means minimal overlap, the asymptotic performance of the discretized overlapping algorithm is worse, like predicted for the non-overlapping algorithm by the continuous analysis in [6]. The benefit of the overlap is thus lost with this discretization!

For Robin transmission conditions obtained by centered finite differences, the algorithm (16) is given by

$$\begin{aligned} Lu^{1,n}_{l,m} &= f_{l,m}, \qquad l < l_1,\ j \in \mathbb{Z}, \\ (D_x^- + (\eta - D_y^+ D_y^-)\tfrac{h}{2} + p)u^{1,n}_{l_1,m} &= (D_x^+ - (\eta - D_y^+ D_y^-)\tfrac{h}{2} + p)u^{2,n-1}_{l_1,m} + hf_{l_1,m}, \\ Lu^{2,n}_{l,m} &= f_{l,m}, \qquad l > l_2,\ j \in \mathbb{Z}, \\ (D_x^+ - (\eta - D_y^+ D_y^-)\tfrac{h}{2} - p)u^{2,n}_{l_2,m} &= (D_x^- + (\eta - D_y^+ D_y^-)\tfrac{h}{2} - p)u^{1,n-1}_{l_2,m} - hf_{l_2,m}. \end{aligned} \tag{18}$$

Proposition 7. *For the discrete optimized Schwarz algorithm (18), the optimized Robin parameter and asymptotic convergence factor are for h small given by*

$$\begin{aligned} &\delta = 0 : p^* = 2^{\frac{1}{4}} \sqrt{\tfrac{2^{\frac{1}{2}}+1}{4+3\cdot 2^{\frac{1}{2}}}} \eta^{\frac{1}{4}} h^{-\frac{1}{2}}, \ \rho_{\max} = 1 - 2\eta^{\frac{1}{4}}(2 + 3\cdot 2^{-\frac{1}{2}})^{\frac{1}{2}}\sqrt{h} + \mathcal{O}(h), \\ &\delta = 1 : p^* = 2^{-\frac{1}{3}} \eta^{\frac{1}{3}} h^{-\frac{1}{3}}, \qquad\qquad \rho_{\max} = 1 - 4\eta^{\frac{1}{6}} 2^{\frac{1}{3}} h^{\frac{1}{3}} + \mathcal{O}(h^{\frac{2}{3}}). \end{aligned} \tag{19}$$

Remark 5. With the centered finite difference approximation of the normal derivative, the discretized optimized Schwarz algorithm for the positive definite Helmholtz equation has the same asymptotic convergence behavior as predicted by the continuous analysis in [6].

4 Conclusions

As we have seen, the discretization can modify the convergence behavior of Schwarz algorithms, compared to the predicted behavior by a continuous analysis. We note however that in all cases we have analyzed, different behavior is only observed when the overlap is sufficiently small. In the case of enough overlap, the results of the discrete and continuous analysis are consistent. This observation suggests that the reason for possibly different behavior of the discrete algorithm could lie in the fact that the physical properties are in those cases not well enough resolved in the overlapping region of very few grid points.

References

1. A.A. Rodríguez and L. Gerardo-Giorda. New nonoverlapping domain decomposition methods for the harmonic Maxwell system. *SIAM J. Sci. Comput.*, 28(1):102–122 (electronic), 2006. ISSN 1064-8275.
2. V. Dolean and M.J. Gander. Why classical Schwarz methods applied to certain hyperbolic systems converge even without overlap. In *Domain Decomposition Methods in Science and Engineering XVII*, volume 60 of *Lecture Notes in Computational Science and Engineering*, pp. 467–475. Springer, Berlin, 2008.
3. V. Dolean, M.J. Gander, and L. Gerardo-Giorda. Optimized Schwarz methods for Maxwell's equations. *SIAM J. Sci. Comput.*, 31(3):2193–2213, 2009.
4. V. Dolean, S. Lanteri, and F. Nataf. Convergence analysis of a Schwarz type domain decomposition method for the solution of the Euler equations. *Appl. Numer. Math.*, 49: 153–186, 2004.

5. M. Dryja and O.B. Widlund. An additive variant of the Schwarz alternating method for the case of many subregions. Technical Report 339, also Ultracomputer Note 131, Department of Computer Science, Courant Institute, 1987.
6. M.J. Gander. Optimized Schwarz methods. *SIAM J. Numer. Anal.*, 44(2):699–731 (electronic), 2006. ISSN 0036-1429.
7. P.-L. Lions. On the Schwarz alternating method. I. In R. Glowinski, G.H. Golub, G.A. Meurant, and J. Périaux, editors, *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pp. 1–42. SIAM, Philadelphia, PA, 1988.
8. P.-L. Lions. On the Schwarz alternating method. II. In T. Chan, R. Glowinski, J. Périaux, and O. Widlund, editors, *Domain Decomposition Methods*, pp. 47–70. SIAM, Philadelphia, PA, 1989.
9. P.-L. Lions. On the Schwarz alternating method. III: a variant for nonoverlapping subdomains. In T.F. Chan, R. Glowinski, J. Périaux, and O. Widlund, editors, *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations, held in Houston, Texas, March 20–22, 1989*, SIAM, Philadelphia, PA, 1990.
10. A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford Science Publications, Oxford 1999.
11. H.A. Schwarz. Über einen Grenzübergang durch alternierendes Verfahren. *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich*, 15:272–286, May 1870.
12. A. Toselli and O. Widlund. *Domain Decomposition Methods – Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2005.

The Pole Condition: A Padé Approximation of the Dirichlet to Neumann Operator

Martin J. Gander[1] and Achim Schädle[2]

[1] Mathematics Section, University of Geneva, CH-1211, Geneva, Switzerland, Martin.gander@unige.ch
[2] Mathematisches Institut, Heinrich-Heine-Universität, D-40225 Düsseldorf, Germany

1 Introduction

When a problem is posed on an unbounded domain, the domain needs to be truncated in order to perform computations, and the pole condition is a new technique developed over the last few years for this purpose. The subject of domain truncation is already an established research field. It was started in 1977 in a seminal paper by Enquist and Majda [6], where a systematic method to obtain absorbing boundary conditions (ABCs) is introduced for wave propagation phenomena. Absorbing boundary conditions are approximations of transparent boundary conditions (TBCs), which, when used to truncate the unbounded domain, lead by definition precisely to the restriction of the original solution on the unbounded domain. Unfortunately transparent boundary conditions often involve expensive non-local operators and are thus inconvenient. Absorbing boundary conditions became immediately a field of interest of mathematicians in approximation theory, see for example [3, 9]. Recent reviews on non-reflecting or absorbing boundary conditions concerning the wave equation are [8] by Hagstrom and more recently Givoli [7]. Non-reflecting boundary conditions for the transient Schrödinger equation are reviewed by Antoine et al. [1].

For the description of resonances for Schrödinger operators, the exterior complex scaling (ECS) method was introduced by Simon [14] in 1979. In the early nineties, a technique called perfectly matched layers (PMLs) was developed by Bérenger [4]. Here the idea is to add a layer just outside where the domain is truncated. In this layer, a modified equation is solved, which can be interpreted as an area with different artificial material, which absorbs outgoing waves, without creating reflections. The PML can be interpreted as a complex coordinate stretching in the layer, by which the original equation is transformed into a new one with appropriate properties, see [5, 15]. Hence it is equivalent to ECS.

Absorbing boundary conditions and perfectly matched layers are two competing techniques with the same purpose, namely to truncate an unbounded domain for computational purposes. In 2003, a new technique for the derivation and approximation of transparent boundary conditions was proposed by Schmidt, Hohage and Zschiedrich [11], based on the so called pole condition:

> "The pole condition is a general concept for the theoretical analysis and the numerical solution of a variety of wave propagation problems. It says that the Laplace transform of the physical solution in the radial direction has no poles in the lower complex half-plane."

The pole condition leads to a numerical method for domain truncation which is easy to implement and has shown great promise in numerical experiments for a variety of problems, see [10, 12, 13]. We show in this paper for a model problem of diffusive nature an error estimate for the numerical method based on the pole condition: the domain truncation achieved is a Padé approximation of the transparent boundary condition.

2 Model Problem

We consider on the domain $\Omega_g := \mathbb{R} \times (0, \pi)$ the elliptic model problem

$$\begin{aligned} (\eta - \Delta)u &= f \ \text{ in } \Omega_g, \\ u(x,0) = u(x,\pi) &= 0, \end{aligned} \tag{1}$$

where $\eta > 0$, and we seek bounded solutions. For an illustration, see Fig. 1. In order

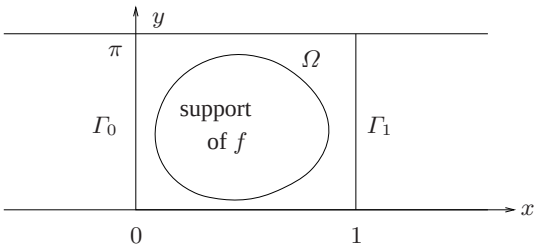


Fig. 1. Domain and support of f.

to perform computations on this problem, we truncate the domain in the unbounded x-direction. We assume that f is compactly supported in $\Omega := (0,1) \times (0,\pi)$, which suggests to truncate the domain along $\Gamma_j = j \times (0,\pi)$, $j = 0, 1$, see Fig. 1, using an artificial boundary conditions of the form

$$\mathcal{B}_0(u)(0) = 0, \quad \mathcal{B}_1(u)(1) = 0. \tag{2}$$

Expanding the solution u in eigenmodes in the y direction, which in our case is a sine-expansion for constant η and the homogeneous Dirichlet conditions at the top and bottom, yields

$$\begin{aligned} (\eta - \partial_{xx} + k^2)\hat{u} &= \hat{f}, \\ \beta_0 \hat{u}(0) = 0, \quad \beta_1 \hat{u}(1) &= 0, \end{aligned} \tag{3}$$

where β_j, $j = 0, 1$ are the symbols of the artificial boundary conditions, and $\hat{f} = \mathcal{F}(f)$ denotes the sine transform of f. A direct calculation shows that if $\beta_j = \partial_n + \sqrt{\eta + k^2}$, the truncated solution and the global solution restricted to Ω coincide, and therefore the exact or transparent boundary conditions (TBCs) are

$$\partial_n \hat{u}(0,k) + \sqrt{\eta + k^2}\hat{u}(0,k) = 0, \quad \partial_n \hat{u}(1,k) + \sqrt{\eta + k^2}\hat{u}(1,k) = 0, \quad (4)$$

and we see the well known Dirichlet to Neumann operator $\mathcal{F}^{-1}(\sqrt{\eta + k^2})$ appear. In order to obtain an absorbing boundary condition, one could therefore approximate the square root either by a polynomial or a rational function.

3 The Pole Condition

In order to explain the pole condition, we follow the quote above and perform now a Laplace transform in the radial direction, which in our case is the x direction, with dual variable $\tilde{s}$, and obtain on the right boundary

$$(\eta + k^2 - \tilde{s}^2)U(\tilde{s},k) + \partial_n \hat{u}(1,k) + \tilde{s}\hat{u}(1,k) = 0, \quad (5)$$

and a similar result on the left of the interface Γ_0. Solving for U, we obtain

$$U(\tilde{s},k) = -\frac{\partial_n \hat{u} + \tilde{s}\hat{u}}{\eta + k^2 - \tilde{s}^2}, \quad (6)$$

and thus $U(\tilde{s},k)$ has two singularities (poles), at $\tilde{s} = \pm\sqrt{\eta + k^2}$. When looking outward from the computational domain, we are interested in bounded solutions, and hence the singularities in the right half plane $\mathbb{R}(\tilde{s}) > 0$ are undesirable, as they correspond to exponentially increasing solutions. Using a partial fraction decomposition, we find

$$U(\tilde{s},k) = \frac{\hat{u}(1,k) - \frac{\partial_n \hat{u}(1,k)}{\sqrt{\eta+k^2}}}{2(\tilde{s} + \sqrt{\eta + k^2})} + \frac{\hat{u}(1,k) + \frac{\partial_n \hat{u}(1,k)}{\sqrt{\eta+k^2}}}{2(\tilde{s} - \sqrt{\eta + k^2})}, \quad (7)$$

and we see again that if $\hat{u}$ satisfies the TBC (4), the undesirable pole represented by the second term of (7) is not present, since the numerator vanishes identically. The key idea of the pole condition is to enforce that the second term can not be present, by imposing analyticity of $U(\tilde{s},k)$ in the right half of the complex plane $\mathbb{R}(\tilde{s}) > 0$. In order to do so, it is convenient to first use the Möbius transform M_{s_0} for $s_0 \in \mathbb{C}$ with positive real part, see Fig. 2, and map the right half plane into the unit circle,

$$M_{s_0} : \tilde{s} \mapsto s = \frac{\tilde{s} - s_0}{\tilde{s} + s_0}, \qquad M_{s_0}^{-1} : s \mapsto \tilde{s} = -s_0\frac{s+1}{s-1}.$$

We now exclude singularities of the solution $U(\tilde{s},k)$ in the right half of the complex plane by enforcing the representation of U in the new variable s by the power-series

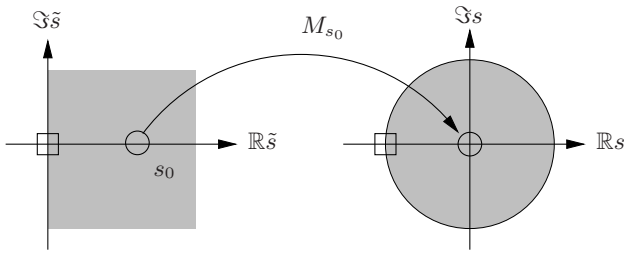


Fig. 2. Möbius transform.

$$U(s,k) = \frac{s-1}{2s_0}\Big((s-1)\sum_{n=0}^{\infty} a_n s^n - \hat{u}\Big). \tag{8}$$

We chose this particular ansatz, because it satisfies automatically the condition from Laplace transform theory that if $\hat{u}$ exists, we must have

$$\lim_{\tilde{s}\to\infty} \tilde{s}U(\tilde{s},k) = \lim_{s\to 1} -s_0\frac{s+1}{s-1}U(s,k) = \hat{u}(1,k).$$

To simplify the notation, we set $\tilde{\eta} := \eta + k^2$ in what follows. Inserting the power-series expansion (8) into Eq. (5), and collecting terms, we obtain

$$\left(\frac{\tilde{\eta}(s-1)^2}{2s_0} - \frac{s_0(s+1)^2}{2}\right)\sum_{n=0}^{\infty} a_n s^n = \left(\frac{\tilde{\eta}(s-1)}{2s_0} - s_0\frac{s+1}{2} - \partial_\nu\right)\hat{u}(1,k). \tag{9}$$

Matching powers of s, we obtain the equations for the power series coefficients a_n,

$$\left(\tilde{\eta} - s_0^2\right)a_0 + \left(s_0^2 + \tilde{\eta}\right)\hat{u}(1,k) = -2s_0\partial_\nu\hat{u}(1,k), \tag{10}$$

$$\left(\tilde{\eta} - s_0^2\right)a_1 - 2\left(\tilde{\eta} + s_0^2\right)a_0 - \left(\tilde{\eta} - s_0^2\right)\hat{u}(1,k) = 0, \tag{11}$$

$$\left(\tilde{\eta} - s_0^2\right)a_{n+1} - 2\left(\tilde{\eta} + s_0^2\right)a_n + \left(\tilde{\eta} - s_0^2\right)a_{n-1} = 0, \quad n = 1,\ldots L-2, \tag{12}$$

$$-2\left(\tilde{\eta} + s_0^2\right)a_{L-1} + \left(\tilde{\eta} - s_0^2\right)a_{L-2} = 0, \tag{13}$$

where we truncated the power series expansion at the L-th term. We observe that the expansion coefficients satisfy a three term recurrence relation similar to the relation satisfied by the original solution in the x direction, when a five point finite difference stencil is used for the discretization, and since the expansion coefficients depend on k, and $\tilde{\eta} = \eta + k^2$, the recurrence relation shows that the expansion coefficients also satisfy a second order differential equation in the y direction. This permits an easy implementation of the expansion coefficients on the same grid as the solution, as illustrated in Fig. 3, and is the reason why it is so easy to use the pole condition truncation. Note that this is the same system of equations for the a_n as obtained using a Galerkin ansatz in the Hardyspace of the unit disc by Hohage and Nannen [10].

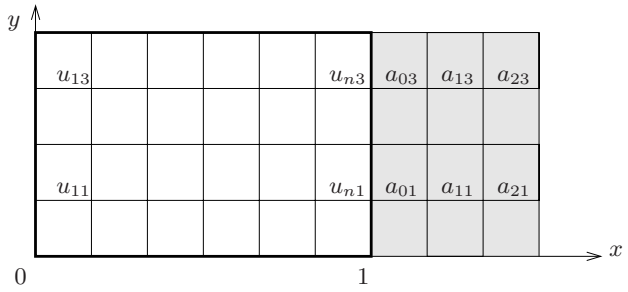


Fig. 3. Implementation of the expansion coefficients on the same grid as the interior unknowns.

4 Error Estimate

In order to gain insight into the approximation we obtain from the truncation at the L-th term, we define

$$b := \frac{\tilde{\eta} + s_0^2}{\tilde{\eta} - s_0^2} = \frac{\eta + k^2 + s_0^2}{\eta + k^2 - s_0^2}, \tag{14}$$

and we start resolving the recurrence relation from the last term (13), which implies

$$a_{L-1} = \frac{1}{2b} a_{L-2}.$$

Using this result and (12) for $n = L - 2$ then gives

$$a_{L-2} = \frac{1}{2b - \frac{1}{2b}} a_{L-3} = \frac{1}{2b-} \frac{1}{2b} a_{L-3},$$

and continuing like this, we arrive when using (12) for $n = 1$ at

$$a_1 = a_0 \, \frac{1}{2b - \frac{1}{2b- \ldots \frac{1}{2b}}} = a_0 \, \frac{1}{2b-} \frac{1}{2b-} \frac{1}{2b-} \cdots \frac{1}{2b} = \sum_{n=1}^{L-1} \frac{1}{2b-},$$

a truncated continued fraction expansion. Using now (11) and (10), and rearranging terms, we obtain the representation of the approximate operator which is defined by the pole condition, namely

$$\partial_\nu \hat{u}(1,k) + \frac{\eta + k^2 - s_0^2}{2s_0} \left(b - \sum_{n=1}^{L} \frac{1}{2b-} \right) \hat{u}(1,k) = 0. \tag{15}$$

Comparing this relation with the TBC from (4), we see that the term containing the continued fraction expansion must represent an approximation of the DtN operator $\sqrt{\eta + k^2}$.

Theorem 1. *If the truncation level L of the continued fraction expansion* (15) *is going to infinity, it represents the exact Dirichlet to Neumann operator,*

$$\frac{\eta + k^2 - s_0^2}{2s_0}\left(b - \sum_{n=1}^{\infty}\frac{1}{2b-}\right)\hat{u}(1,k) = \sqrt{\eta + k^2}\hat{u}(1,k) = -\partial_\nu \hat{u}(1,k), \quad (16)$$

independently of the expansion point s_0, and therefore the truncation condition obtained from the pole condition converges to the TBC.

Proof. The continued fraction in (16) maybe rewritten as

$$b - \sum_{n=1}^{\infty}\frac{1}{2b-} = b - \frac{1}{2b - \frac{1}{x}} \quad \text{with} \quad x = 2b - \frac{1}{x}. \quad (17)$$

The roots of $x^2 - 2bx + 1$ are $x_{1,2} = \pm\sqrt{b^2-1} + b$. Inserting $x = \sqrt{b^2-1} + b$ into (17), and using the identity

$$b - \frac{1}{b + \sqrt{b^2-1}} = \sqrt{b^2-1},$$

we find from (15) and using the definition for b in (14) that

$$\frac{\eta + k^2 - s_0^2}{2s_0}\left(\left(\frac{\eta + k^2 + s_0^2}{\eta + k^2 - s_0^2}\right)^2 - 1\right)^{\frac{1}{2}}\hat{u} = -\partial_\nu \hat{u},$$

which can be simplified to give the result. □

We are now interested in obtaining an error estimate if the power series is truncated at the L-th term. To this end, we use the following well known result for truncated continued fraction expansions.

Theorem 2 (Sect. 4 [2]). *The L-th truncated continued fraction expansion can be represented by*

$$a_0 + \sum_{n=1}^{L}\frac{b_n}{a_n+} = \frac{A_L}{B_L},$$

where A_n and B_n are defined by the recurrence relations

$$\begin{aligned} A_{-1} = 1,\ A_0 = a_0, \quad A_{n+1} = a_{n+1}A_n + b_{n+1}A_{n-1}, \\ B_{-1} = 0,\ B_0 = 1, \quad B_{n+1} = a_{n+1}B_n + b_{n+1}B_{n-1}. \end{aligned} \quad (18)$$

In what follows we will call $(a_n)_n$ the denominator sequence and $(b_n)_n$ the numerator sequence.

Theorem 3. *The truncated recurrence relation* (10)*,* (11)*,* (12) *and* (13) *from the pole condition represents an (L+1,L)-Padé approximation of the symbol of the DtN operator $s_0\sqrt{1+z}$ about $z = 0$, where $z = \frac{\eta + k^2 - s_0^2}{s_0^2}$.*

Proof. The Padé approximation of $(1+z)^{\frac{1}{2}}$ expanded at $z=0$ is given by the continued fraction

$$(1+z)^{\frac{1}{2}} = 1 + \frac{\frac{1}{2}z}{1+}\frac{\frac{1}{2}z}{2+}\frac{\frac{1}{2}z}{1+}\frac{\frac{1}{2}z}{2+}\ldots.$$

Hence the denominator sequence is $a_0 = 1$ and $a_n = \frac{3+(-1)^n}{2}$, $n \geq 1$, whereas the numerator sequence is given by $b_n = \frac{1}{2}z$, see [2], equation (6.4) on page 139. Using Theorem 2, the L-th approximation is given by the fraction of A_L and B_L. Using the recurrence relations (18) with leading terms $2n+1$, $2n$, and $2n-1$, the even terms can be eliminated to give

$$\begin{array}{lll} A_{-1} = 1,\ A_1 = \frac{z+2}{2}, & A_{2n+1} = (2+z)A_{2n-1} - \frac{z^2}{4}A_{2n-3}, \\ B_{-1} = 0,\ B_1 = 1, & B_{2n+1} = (2+z)B_{2n-1} - \frac{z^2}{4}B_{2n-3}. \end{array} \tag{19}$$

Using the variable $z = \frac{\eta+k^2-s_0^2}{s_0^2}$ in the continued fraction representation for the square root stemming from the pole condition (15), we find that the denominator sequence is

$$c_0 = \frac{\eta+k^2+s_0^2}{2s_0} = s_0\frac{z+2}{2},\ c_n = 2\frac{\eta+k^2+s_0^2}{\eta+k^2-s_0^2} = \frac{2(2+z)}{z}, \quad \text{for } n \geq 1,$$

and the numerator sequence is

$$d_1 = -\frac{\eta+k^2-s_0^2}{2s_0} = -s_0\frac{z}{2},\ d_n = -1, \quad \text{for } n \geq 2.$$

Using again Theorem 2, the L-th approximation is given by the fraction of C_L and D_L, which are given by

$$\begin{array}{llll} C_{-1} = 1,\ C_0 = s_0\frac{z+2}{2}, & C_1 = s_0(\frac{z}{2}+4+\frac{4}{z}), & C_{n+1} = \frac{2(2+z)}{z}C_n - C_{n-1}, \\ D_{-1} = 0,\ D_0 = 1, & D_1 = \frac{2}{z}(2+z), & D_{n+1} = \frac{2(2+z)}{z}D_n - D_{n-1}. \end{array}$$

If we define $\tilde{C}_n := (z/2)^n C_n$, $\tilde{D}_n := (z/2)^n D_n$, we obtain for $n \geq 0$

$$\begin{array}{lll} \tilde{C}_0 = s_0\frac{z+2}{2}, & \tilde{C}_1 = s_0\frac{z}{2}(\frac{z}{2}+4+\frac{4}{z}), & \tilde{C}_{n+1} = (2+z)\tilde{C}_n - \frac{z^2}{4}\tilde{C}_{n-1}, \\ \tilde{D}_0 = 1, & \tilde{D}_1 = 2+z, & \tilde{D}_{n+1} = (2+z)\tilde{D}_n - \frac{z^2}{4}\tilde{D}_{n-1}, \end{array} \tag{20}$$

which is the same recurrence as (19). Since $\tilde{C}_0 = s_0 A_1$, $\tilde{C}_1 = s_0 A_3$, $\tilde{D}_0 = B_1$ and $\tilde{D}_1 = B_3$, the proof is complete. □

References

1. X. Antoine, A. Arnold, C. Besse, M. Ehrhardt, and A. Schädle. A review of transparent and artificial boundary conditions techniques for linear and nonlinear Schrödinger equations. *Commun. Comput. Phys.*, 4(4):729–796, 2008.

2. G.A. Baker. *Padé Approximants Part I: Basic Theory*. Encyclopedia of Mathematics and Its Applications. Addison-Wesley, Reading, MA, 1981.
3. A. Bayliss and E. Turkel. Radiation boundary conditions for wave-like equations. *Commun. Pure Appl. Math.*, 33(6):707–725, 1988.
4. J.P. Bérenger. A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.*, 114(2):185–200, 1994.
5. W.C. Chew and W.H. Weedon. A 3d perfectly matched medium from modified Maxwell's equations with stretched coordinates. *Microwave Opt. Technol. Lett.*, 7(13):599–604, 1994.
6. B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. *Math. Comput.*, 31(139):629–651, 1977.
7. D. Givoli. High-order local non-reflecting boundary conditions: a review. *Wave Motion*, 39(4):319–326, 2004.
8. T. Hagstrom. Radiation boundary conditions for the numerical simulation of waves. *Acta Numer.*, 8:47–106, 1999.
9. L. Halpern. Absorbing boundary conditions for the discretization schemes of the one-dimensional wave equation. *Math. Comput.*, 38(158):415–429, 1982.
10. T. Hohage and L. Nannen. Hardy space infinite elements for scattering and resonance problems. *SIAM J. Numer. Anal.*, 47:972–996, 2009.
11. T. Hohage, F. Schmidt, and L. Zschiedrich. Solving time-harmonic scattering problems based on the pole condition. I: Theory. *SIAM J. Math. Anal.*, 35:183–210, 2003.
12. D. Ruprecht, A. Schädle, F. Schmidt, and L. Zschiedrich. Transparent boundary conditions for time-dependent problems. *SIAM J. Sci. Comput.*, 30:2358–2385, 2008.
13. F. Schmidt, T. Hohage, R. Klose, A. Schädle, and L. Zschiedrich. Pole condition: A numerical method for Helmholtz-type scattering problems with inhomogeneous exterior domain. *J. Comput. Appl. Math.*, 218(1):61–69, 2008.
14. B. Simon. The definition of molecular resonance curves by the method of exterior complex scaling. *Phys. Lett.*, 71A(2,3):211–214, 1979.
15. F.L. Teixeira and W.C. Chew. General closed-form PML constitutive tensors to match arbitrary bianisotropic and dispersive linear media. *IEEE Microwave Guided Wave Lett.*, 8(6):223–225, 1998.

Discontinuous Galerkin and Nonconforming in Time Optimized Schwarz Waveform Relaxation

Laurence Halpern[1], Caroline Japhet[1], and Jérémie Szeftel[2]

[1] LAGA, Université Paris XIII, 93430 Villetaneuse, France, halpern@math.univ-paris13.fr; japhet@math.univ-paris13.fr, partially supported by french ANR (COMMA) and GdR MoMaS.

[2] Department of Mathematics, Princeton University, Fine Hall, Washington Road, Princeton NJ 08544-1000, USA; C.N.R.S., MAB, Université Bordeaux 1, 33405 Talence Cedex, France, partially supported by NSF Grant DMS-0504720, jszeftel@math.princeton.edu.

1 Introduction

In many fields of applications such as reactive transport or ocean-atmosphere coupling, models with very different spatial and time scales have to be coupled. Optimized Schwarz Waveform Relaxation methods (OSWR), applied to linear advection-reaction-diffusion problems in [1, 8], provide efficient solvers for this purpose. They have two main advantages: first, they are global in time and thus permit non conforming space-time discretization in different subdomains, and second, few iterations are needed to compute an accurate solution, due to optimized transmission conditions. It has been proposed in [4] to use a discontinuous Galerkin method in time as a subdomain solver. Rigorous analysis can be made for any degree of accuracy and local time-stepping, and finally time steps can be adaptively controlled by a posteriori error analysis, see [6, 7, 10].

We present here the 2D analysis of the method. The time interval is split into time windows, and in each time window, a few iterations of an OSWR algorithm are computed, using second order optimized transmission conditions. The subdomain solver is the discontinuous Galerkin method in time, and classical finite elements in space. Coupling between subdomains is done by a simple and optimal projection algorithm without any additional grid (see [2, 3]). The mathematical analysis is carried out on the problem semi-discrete in time. The nonconforming DG-OSWR domain decomposition method is proved to be well-posed and convergent for a decomposition into strips, and the error analysis is performed in the case of Robin transmission conditions. We present numerical results in two dimensions which extends the domain of validity of the approach to the fully discrete problem.

We consider the advection-reaction-diffusion equation in $\mathbb{R}^2$, written for variational purpose in the form

$$\partial_t u + \frac{1}{2}\nabla\cdot(bu) + \frac{1}{2}b\cdot\nabla u - \nabla\cdot(\nu\nabla u) + cu = f. \tag{1}$$

The initial condition is u_0. The advection and diffusion coefficients $\boldsymbol{b} = (b^1, b^2)$ and ν, as well as the reaction coefficient c, are piecewise constant, *i.e.* constant in the subdomains $\Omega_i,\ i \in \{1, ..., I\}$. The subdomains are strips $\Omega_i = (\alpha_i, \alpha_{i+1}) \times \mathbb{R}$, with $\alpha_1 = -\infty$ and $\alpha_{I+1} = +\infty$. More general geometries as well as piecewise smooth coefficients will be studied in [5]. We suppose that $\nu > 0$ and $c > 0$.

2 Local Problem and Time Discontinuous Galerkin

The optimized Schwarz waveform relaxation algorithm, as described in [1], introduces a sequence of initial boundary value problems in $\Omega = (\alpha, \beta) \times \mathbb{R}$ of the following type:

$$\begin{aligned} &\partial_t u + \tfrac{1}{2}\nabla\cdot(bu) + \tfrac{1}{2}b\cdot\nabla u - \nabla\cdot(\nu\nabla u) + cu = f \text{ in } \Omega\times(0,T),\\ &(\nu\,\partial_n - \frac{\boldsymbol{b}\cdot n}{2})u + \mathcal{S}u = g \text{ on } \Gamma\times(0,T), \end{aligned} \tag{2}$$

where n is the unit outward normal to Γ, and $\mathcal{S}$ is the boundary operator defined on $\Gamma = \{\alpha, \beta\} \times \mathbb{R}$ by $\mathcal{S}u = p\,u + q\,(\partial_t u + r\partial_y u - s\partial_{yy} u)$. Here $p,\ q,\ r$ and s are real parameters, constrained to $p > 0, q \geq 0,\ s > 0$. If $q = 0$, the boundary condition reduces to a Robin boundary condition. We define the bilinear forms m and a by $m(u, v) = (u, v)_{L^2(\Omega)} + q(u, v)_{L^2(\Gamma)}$, and

$$\begin{aligned} a(u,v) := \int_\Omega \big(\frac{1}{2}((\boldsymbol{b}\cdot\nabla u)v - (\boldsymbol{b}\cdot\nabla v)u) + \nu\nabla u\cdot\nabla v + cuv\big)dx \\ + \int_\Gamma \big(qs\partial_y u\partial_y v + q\,r\partial_y uv + puv\big)\,dy. \end{aligned} \tag{3}$$

By the Green's formula, we obtain a variational formulation of (2):

$$\frac{d}{dt}\,m(u,v) + a(u,v) = (f,v)_{L^2(\Omega)} + (g,v)_{L^2(\Gamma)}, \quad \forall v \in V, \tag{4}$$

with $V = H^1(\Omega)$ if $q = 0$ and $V = H_1^1(\Omega)$ defined below, if $q > 0$. The problem is well-posed: if $q = 0$, if f is in $L^2(0, T, L^2(\Omega))$, u_0 is in $H^1(\Omega)$, and g is in $L^2(0, T, H^{1/2}(\Gamma))$, then the subdomain problem (2) has a unique solution u in $L^2(0, T, H^2(\Omega)) \cap H^1(0, T; L^2(\Omega))$. If $q > 0$, we introduce the spaces $H_s^s(\Omega) = \{v \in H^s(\Omega), v_{|\Gamma} \in H^s(\Gamma)\}$ which are defined for $s > 1/2$. If f is in $L^2(0, T, L^2(\Omega))$, u_0 is in $H_1^1(\Omega)$, and g is in $L^2((0, T) \times \Gamma)$, then the subdomain problem (2) has a unique solution u in $L^2(0, T, H_2^2(\Omega))$ with $\partial_t u \in L^2(0, T; L^2(\Omega) \cap L^2(\Gamma))$, see [1, 9].

We now introduce the time-discontinuous Galerkin method, as described and analysed in [6]. We are given a decomposition $\mathcal{T}$ of the time interval $(0, T)$, $I_n =$

$(t_n, t_{n+1}]$, for $0 \le n \le N$, the mesh size is $k_n = t_{n+1} - t_n$. For $\mathcal{B}$ a Banach space and I an interval of $\mathbb{R}$, define for any integer $d \ge 0$

$$\mathbf{P}_d(\mathcal{B}, \mathcal{T}) = \{\varphi : (0,T) \to \mathcal{B},\ \varphi_{|I_n} = \sum_{i=0}^{d} \varphi_i t^i,\ \varphi_i \in \mathcal{B},\ 0 \le n \le N\}.$$

Let $\mathcal{B} = H_1^1(\Omega)$ if $q > 0$, $\mathcal{B} = H^1(\Omega)$ if $q = 0$. We approximate u by a function $U \in \mathbf{P}_d(\mathcal{B}, \mathcal{T})$ such that $U(0,\cdot) = u_0$ and for all V in $\mathbf{P}_d(\mathcal{B}, \mathcal{T})$,

$$\int_{I_n} (\, m(\dot{U}, V) + a(U, V))\, dt + m(U(t_n^+) - U(t_n^-), V(t_n^+)) = \int_{I_n} L(V)\, dt, \quad (5)$$

with $L(V) = (f, V)_{L^2(\Omega)} + (g, V)_{L^2(\Gamma)}$. Due to the discontinuous nature of the test and trial spaces, the method is an implicit time stepping scheme, and $U \in \mathbf{P}_d(\mathcal{B}, \mathcal{T})$ is obtained recursively on each subinterval, which makes the method very flexible.

Theorem 1. *If $p > 0$, $q \ge 0$, $s > 0$, Eq. (5) defines a unique solution.*

The result relies on the fact that the bilinear form a is positive definite. This is most easily seen by using a basis of Legendre polynomials.

We will make use of the following remark due to [7]. Introduce the Gauss-Radau points, $(0 < \tau_1 < \ldots < \tau_{d+1} = 1)$, defined such that the quadrature formula $\int_0^1 f(t)dt \approx \sum_{q=1}^{d+1} w_q f(\tau_q)$ is exact in $\mathbf{P}_{2d}$, and the interpolation operator $\mathcal{I}_n$ on $[t_n, t_{n+1}]$ at points $(t_n, t_n + \tau_1 k_n, \ldots, t_n + \tau_{d+1} k_n)$. For any $\chi \in \mathbf{P}_d$, $\mathcal{I}_n \chi \in \mathbf{P}_{d+1}$, is such that $\mathcal{I}_n\chi(t_n) = \chi(t_n^-)$, $\mathcal{I}_n\chi(t_{n+1}) = \chi(t_{n+1}^-)$, and therefore for any ψ in $\mathbf{P}_d$, we have

$$\int_{I_n} \frac{d}{dt}(\mathcal{I}_n\chi)\psi\, dt - \int_{I_n} \frac{d\chi}{dt}\, \psi\, dt = (\chi(t_n^+) - \chi(t_n^-))\psi(t_n^+). \quad (6)$$

As a consequence, we have a very useful inequality:

$$\int_{I_n} \frac{d}{dt}(\mathcal{I}_n\psi)\psi dt \ge \frac{1}{2}[\psi(t_{n+1}^-)^2 - \psi(t_n^-)^2]. \quad (7)$$

Equation (5) can be written in a strong form as

$$\begin{aligned} &\partial_t(\mathcal{I}\, U) + \tfrac{1}{2}\nabla\cdot(\boldsymbol{b}U) + \tfrac{1}{2}\boldsymbol{b}\cdot\nabla U - \nabla\cdot(\nu\nabla U) + c\, U = \mathcal{P}f \text{ in } \Omega\times(0,T), \\ &\big(\nu\,\partial_{\boldsymbol{n}} - \tfrac{\boldsymbol{b}\cdot\boldsymbol{n}}{2}\big)U + p\, U + q(\partial_t(\mathcal{I}U) + r\partial_y U - s\partial_{yy}U) = \mathcal{P}g \text{ on } \Gamma\times(0,T), \end{aligned} \quad (8)$$

where $\mathcal{P}$ is the L^2 projection in time on $\mathbf{P}_d(\mathcal{B}, \mathcal{T})$ ($\mathcal{B}$ is defined by the underlying space), and $\mathcal{I}$ is the operator whose restriction to each subinterval is $\mathcal{I}_n$. We discuss now the iterative algorithm.

3 The Optimized Schwarz Waveform Relaxation Algorithm Discretized in Time with Different Subdomain Grids

For each subdomain Ω_i, the indices of the neighbouring subdomains are $j \in \mathcal{N}_i$. Since b is constant in Ω_i, equal to b_i, $\frac{1}{2}\nabla\cdot(b_i u_i^k) + \frac{1}{2}b_i\cdot\nabla u_i^k = \nabla\cdot(b_i u_i^k)$. At the continuous level, the algorithm is

$$\partial_t u_i^k + \nabla\cdot(\boldsymbol{b}_i u_i^k - \nu_i \nabla u_i^k) + c_i u_i^k = f \text{ in } \Omega_i \times (0,T), \tag{9}$$
$$(\nu_i \partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}) u_i^k + \mathcal{S}_{ij} u_i^k = (\nu_j \partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_j\cdot\boldsymbol{n}_i}{2})\, u_j^{k-1} + \mathcal{S}_{ij}\, u_j^{k-1} \text{ on } \Gamma_{ij}, j\in\mathcal{N}_i,$$

with $\nu = \nu_i$ in Ω_i, $\mathcal{S}_{ij}u = p_{ij}\,u + q_{ij}\,(\partial_t u + r_{ij}\partial_y u - s_{ij}\partial_{yy}u)$.

Theorem 2. *For any value of $p_{ij} > 0$, $q_{ij} = q \geq 0$, $r_{ij} = r$ and $s_{ij} = s > 0$, the algorithm* (9) *converges in each subdomain to the solution u of problem* (1)*.*

The proof of this theorem will be given in [5], for general geometries and variable coefficients. It relies on elaborate energy estimates, the use of Trace Theorems and the Gronwall Lemma.

Our purpose here is to describe the discrete formulation in detail. The time partition in subdomain Ω_i is $\mathcal{T}_i$, with $N_i + 1$ intervals I_n^i, and mesh size k_n^i. In view of formulation (8), we define interpolation operators $\mathcal{I}^i$ and projection operators $\mathcal{P}^i$ in each subdomain, and we solve

$$\partial_t(\mathcal{I}^i U_i^k) + \nabla\cdot(\boldsymbol{b}_i U_i^k - \nu_i\nabla U_i^k) + c_i\, U_i^k = \mathcal{P}^i f \text{ in } \Omega_i\times(0,T), \tag{10}$$
$$\big(\nu_i\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\big)\, U_i^k + S_{ij}U_i^k = \mathcal{P}^i\big((\nu_j\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_j\cdot\boldsymbol{n}_i}{2})\, U_j^{k-1} + \widetilde{S}_{ij}U_j^{k-1}\big) \text{ on } \Gamma_{ij},$$

with $S_{ij}U = p_{ij}\,U + q_{ij}\,(\partial_t(\mathcal{I}^i U) + r_{ij}\partial_y U - s_{ij}\partial_{yy}U)$ and $\widetilde{S}_{ij}U = p_{ij}\,U + q_{ij}\,(\partial_t(\mathcal{I}^j U) + r_{ij}\partial_y U - s_{ij}\partial_{yy}U)$. If the algorithm converges, it converges to the solution of

$$\partial_t(\mathcal{I}^i U_i) + \nabla\cdot(\boldsymbol{b}_i U_i - \nu_i\nabla U_i) + c_i\, U_i = \mathcal{P}^i f \text{ in } \Omega_i\times(0,T),$$
$$\big(\nu_i\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\big)\, U_i + S_{ij}\, U_i = \mathcal{P}^i\big((\nu_j\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_j\cdot\boldsymbol{n}_i}{2})\, U_j + \widetilde{S}_{ij}\, U_j\big) \text{ on } \Gamma_{ij}. \tag{11}$$

Theorem 3. *Assume $p_{ij} = p > 0$. If $q_{ij} = 0$, or if $q_{ij} = q > 0$ with $r_{ij} = 0$, $s_{ij} = s > 0$ and $\boldsymbol{b}_i = 0$, Problem* (11) *has a unique solution $(U_i)_{i\in J}$, and U_i is the limit of the iterates of algorithm* (10)*.*

The proof is based on energy estimates (see [5]).

We now state the error estimate.

Theorem 4. *Let $k = \sup_n k_n$. If $p_{ij} = p > 0$ and $q_{ij} = 0$, the error between u and the solution U_i of* (11) *is estimated by:*

$$\sum_{i=1}^{I} \|u - U_i\|^2_{L^\infty(0,T,L^2(\Omega_i))} \leq C k^{2(d+1)} \|\partial_t^{d+1} u\|^2_{L^2(0,T;H^2(\Omega))}. \tag{12}$$

Proof. We introduce the projection operator P_i^- as

$$\forall n\in\{1,\ldots,N_i\},\ P_i^-\varphi \in \mathbf{P}_d(I_n^i),$$
$$P_i^-\varphi(t_{n+1}^i) = \varphi(t_{n+1}^i), \quad \forall\psi\in\mathbf{P}_{d-1}(I_n^i), \int_{I_n^i}(P_i^-\varphi - \varphi)(t)\psi(t)\,dt = 0.$$

We define $W_i = P_i^-(u|_{\Omega_i})$, $\Theta_i = U_i - W_i$ and $\rho_i = W_i - u|_{\Omega_i}$. Classical projection estimates in [10] yield the estimate on ρ_i:

$$\sum_{i=1}^{I} \|\rho_i\|^2_{L^\infty(0,T,L^2(\Omega_i))} \leq Ck^{2(d+1)} \|\partial_t^{d+1} u\|^2_{L^2(0,T;L^2(\Omega))}.$$

Since $U_i - u|_{\Omega_i} = \Theta_i + \rho_i$, it suffices to prove estimate (12) for Θ_i. Now, using the equations of u and U_i, and the identity $\frac{d}{dt}\mathcal{I}^i P_i^- = \mathcal{P}^i \frac{d}{dt}$, Θ_i satisfies:

$$\begin{aligned}
&\partial_t(\mathcal{I}^i\,\Theta_i) + \nabla\cdot(\boldsymbol{b}_i\Theta_i) - \nu\Delta\Theta_i + c_i\Theta_i = \mathcal{P}^i(-\nabla\cdot(\boldsymbol{b}_i\rho_i) + \nu\Delta\rho_i - c_i\rho_i) \\
&\qquad\qquad + (1-\mathcal{P}^i)\partial_t u \text{ in } \Omega_i\times(0,T), \\
&\big(\nu_i\,\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\big)\Theta_i + p\,\Theta_i = \mathcal{P}^i\big((\nu_j\,\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_j\cdot\boldsymbol{n}_i}{2})\Theta_j + p\,\Theta_j\big) \\
&\qquad\qquad - (1-\mathcal{P}^i)\big((\nu_j\,\partial_{\boldsymbol{n}_i} - \frac{\boldsymbol{b}_j\cdot\boldsymbol{n}_i}{2})W_j + p\,W_j\big) \text{ on } \Gamma_{ij}\times(0,T).
\end{aligned} \tag{13}$$

We set $\|\varphi\|_i = \|\varphi\|_{L^2(\Omega_i)}$ and $|||\varphi|||_i^2 = \nu_i\|\nabla\varphi\|^2_{L^2(\Omega_i)} + c\|\varphi\|^2_{L^2(\Omega_i)}$. Multiply the first equation of (13) by Θ_i, integrate over $(t_n^i, t_{n+1}^i)\times\Omega_i$, using (7) and integrate by parts in space. Complete the argument by using Cauchy Schwarz inequality:

$$\begin{aligned}
\frac{1}{2}\|\Theta_i((t_{n+1}^i)^-)\|_i^2 + \int_{I_n^i} |||\Theta_i(t,\cdot)|||_i^2\,dt - \int_{I_n^i}\int_{\Gamma_i} (\nu_i\partial_{\boldsymbol{n}_i}\Theta_i - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\Theta_i)\,\Theta_i dy\,dt \\
\leq \frac{1}{2}\|\Theta_i((t_n^i)^-)\|_i^2 + C\int_{I_n^i}\|\rho_i(t,\cdot)\|^2_{H^2(\Omega_i)}\,dt.
\end{aligned}$$

Rewriting the boundary integral, we obtain:

$$\begin{aligned}
&\frac{1}{2}\|\Theta_i((t_{n+1}^i)^-)\|_i^2 + \int_{I_n^i} |||\Theta_i(t,\cdot)|||_i^2\,dt \\
&\qquad + \frac{1}{4p}\sum_{j\in\mathcal{N}_i}\int_{I_n^i}\int_{\Gamma_{ij}} (\nu_i\partial_{\boldsymbol{n}_i}\Theta_i - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\Theta_i - p\Theta_i)^2\,dy\,dt \\
&\leq \frac{1}{4p}\sum_{j\in\mathcal{N}_i}\int_{I_n^i}\int_{\Gamma_{ij}} (\nu_i\partial_{\boldsymbol{n}_i}\Theta_i - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\Theta_i + p\Theta_i)^2\,dy \\
&\qquad + \frac{1}{2}\|\Theta_i((t_n^i)^-)\|_i^2 + C\int_{I_n^i}\|\rho_i(t,\cdot)\|^2_{H^2(\Omega_i)}\,dt.
\end{aligned}$$

Using the transmission condition in (13) and the fact that $\mathcal{P}^i$ and $1-\mathcal{P}^i$ are orthogonal to each other and have norm 1, we get by a trace theorem:

$$\frac{1}{2}\|\Theta_i((t_{n+1}^i)^-)\|_i^2 + \int_{I_n^i} |||\Theta_i(t,\cdot)|||_i^2 \, dt$$
$$+ \frac{1}{4p}\sum_{j\in\mathcal{N}_i}\int_{I_n^i}\int_{\Gamma_{ij}} (\nu_i\partial_{\boldsymbol{n}_i}\Theta_i - \frac{\boldsymbol{b}_i\cdot\boldsymbol{n}_i}{2}\Theta_i - p\Theta_i)^2\, dy\, dt$$
$$\leq \frac{1}{4p}\sum_{j\in\mathcal{N}_i}\int_{I_n^i}\int_{\Gamma_{ij}} (\nu_j\partial_{\boldsymbol{n}_j}\Theta_j - \frac{\boldsymbol{b}_j\cdot\boldsymbol{n}_j}{2}\Theta_j - p\Theta_j)^2\, dy + \frac{1}{2}\|\Theta_i((t_n^i)^-)\|_i^2$$
$$+ C\int_{I_n^i}\|\rho_i(t,\cdot)\|^2_{H^2(\Omega_i)}\, dt + C\int_{I_n^i}\|(1-\mathcal{P}^i)(u|_{\Omega_i})(t,\cdot)\|^2_{H^2(\Omega_i)}\, dt. \quad (14)$$

Classical error estimates in [10] imply:

$$\int_0^T \|\rho_i(t,\cdot)\|^2_{H^2(\Omega_i)}\, dt + \int_0^T \|(1-\mathcal{P}^i)(u|_{\Omega_i})(t,\cdot)\|^2_{H^2(\Omega_i)}$$
$$\leq Ck^{2(d+1)}\|\partial_t^{d+1}u\|^2_{L^2(0,T;H^2(\Omega_i))}. \quad (15)$$

Summing (14) in j and n, and using the previous equation yields (12).

4 Numerical Results

The above analysis deals with continuous problems and problems semi-discretized in time. We have implemented the algorithm with $d = 1$ and $\mathbf{P}_1$ finite elements in space in each subdomain using mortar methods like in [3], in order to permit non-matching grids in time and space on the boundary. Time windows are used in order to reduce the number of iterations of the algorithm. In the first example, the coefficients are optimized numerically using the convergence factor. In the second one, formulas from [1] are used.

We first give an example of a multidomain solution with time windows. The physical domain is $\Omega = (0,1)\times(0,2)$, the final time is $T = 4$. The initial value and the right hand side are $u_0 = f = e^{-100((x-0.55)^2+(y-1.7)^2)}$. The domain Ω is split into two subdomains $\Omega_1 = (0,0.5)\times(0,2)$ and $\Omega_2 = (0.5,1)\times(0,2)$. The reaction c is zero, the advection and diffusion coefficients are $\boldsymbol{b}_1 = (0,-1)$, $\nu_1 = 0.05$, and $\boldsymbol{b}_2 = (-0.1,0)$, $\nu_2 = 0.1$. The mesh size and time step in Ω_1 are $h_1 = 3.93.10^{-2}$ and $k_1 = 2.5.10^{-2}$, while in Ω_2, $h_2 = 8.84.10^{-2}$ and $k_2 = 6.25.10^{-2}$. In Fig. 1, we observe, at final time $T = 4$, that the approximate solution computed using 4 uniform time windows, with 3 iterations in the first time window, and then 2 iterations in the next ones (right figure), is close to the reference solution computed in one time window on a conforming finer space-time grid (left figure).

We analyze now the precision for continuous coefficients. The advection field is $\boldsymbol{b} = (-\sin(\pi(y-\frac{1}{2}))\cos(\pi(x-\frac{1}{2})), \cos(\pi(y-\frac{1}{2}))\sin(\pi(x-\frac{1}{2})))$, and the diffusion is $\nu = 1$. The exact solution is given by $u(x,t) = cos(\pi x)sin(\pi y)cos(\pi t)$, in the unit square. The domain is decomposed into 2 subdomains with the interface at $x = 0.3$. The space grid is fixed and non conforming with mesh sizes $h_1 = 0.0074$

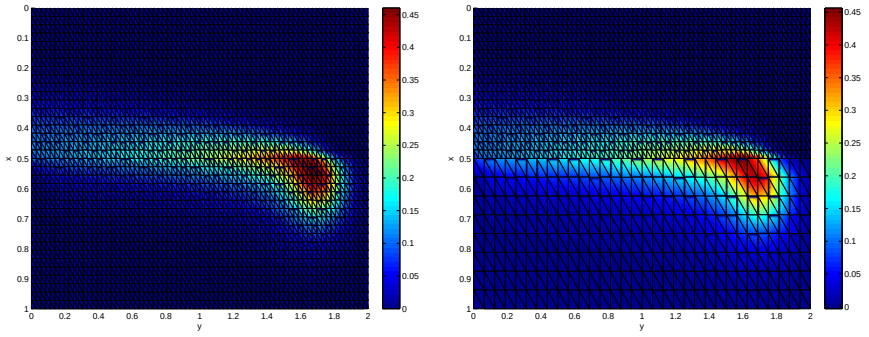


Fig. 1. Computation using discontinuous Galerkin with time windows.

and $h_2 = 0.011$. We start with four time grids : time grids 1 and 2 are the conforming finner and coarser ones with respectively 7 and 5 grid points in each domain. Time grid 3 is nonconforming with 5 grid points in Ω_1 and 7 grid points in Ω_2, and time grid 4 is nonconforming with 7 grid points in Ω_1 and 5 grid points in Ω_2. Thereafter the time steps are divided by 2 several times. Figure 2 shows the norms of the error in $L^\infty(I; L^2(\Omega_i))$ versus the number of time refinements, for subdomain 1 on the left, and subdomain 2 on the right. First we observe the order 2 in time for conforming and nonconforming cases. They fit the theoretical estimates, even though we have theoretical results only for Robin transmission conditions and the space continuous problem. Moreover, the error obtained in the nonconforming case, in the subdomain where the grid is finer, is nearly the same as the error obtained in the conforming finer case.

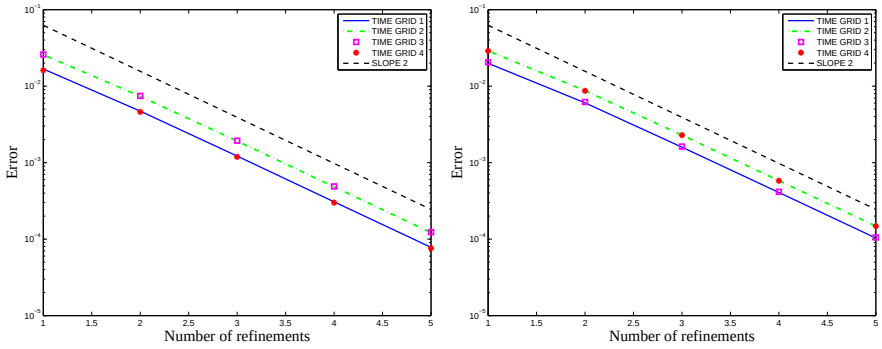


Fig. 2. Error curves versus the refinement in time, for Ω_1 (*left*) and Ω_2 (*right*).

5 Conclusions

We have extended the numerical method proposed in [4] to higher dimensions and analyzed it for heterogeneous advection-reaction-diffusion problems. It relies on the splitting of the time interval into time windows, in which a few iterations of an OSWR algorithm are performed by a discontinuous Galerkin method in time, with projection between space-time grids on the interfaces. We have shown both theoretically and numerically that the method preserves the order of the discontinuous Galerkin method.

References

1. D. Bennequin, M.J. Gander, and L. Halpern. A homographic best approximation problem with application to optimized Schwarz waveform relaxation. *Math. Comput.*, 78:185–223, 2009.
2. M.J. Gander, L. Halpern, and F. Nataf. Optimal Schwarz waveform relaxation for the one dimensional wave equation. *SIAM J. Numer. Anal.*, 41(5):1643–1681, 2003.
3. M.J. Gander, C. Japhet, Y. Maday, and F. Nataf. A new cement to glue nonconforming grids with Robin interface conditions : The finite element case. In R. Kornhuber, R.H.W. Hoppe, J. Périaux, O. Pironneau, O.B. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering*, volume 40 of *Lecture Notes in Computational Science and Engineering*, pp. 259–266. Springer Berlin, Heidelberg, New York, 2005.
4. L. Halpern and C. Japhet. Discontinuous Galerkin and nonconforming in time optimized Schwarz waveform relaxation for heterogeneous problems. In U. Langer, M. Discacciati, D.E. Keyes, O.B. Widlund, and W. Zulehner, editors, *Decomposition Methods in Science and Engineering XVII*, volume 60 of *Lecture Notes in Computational Science and Engineering*, pp. 211–219. Springer Berlin, Heidelberg, New York, 2008.
5. L. Halpern, C. Japhet, and J. Szeftel. Discontinuous Galerkin and nonconforming in time optimized Schwarz waveform relaxation for heterogeneous problems. In preparation, 2009.
6. C. Johnson, K. Eriksson, and V. Thomée. Time discretization of parabolic problems by the discontinuous Galerkin method. *RAIRO Modél. Math. Anal. Numér.*, 19, 1985.
7. C. Makridakis and R. Nochetto. A posteriori error analysis for higher order dissipative methods for evolution problems. *Numer. Math.*, 104(4):489–514, 2006.
8. V. Martin. An optimized Schwarz waveform relaxation method for the unsteady convection diffusion equation in two dimensions. *Appl. Numer. Math.*, 52:401–428, 2005.
9. J. Szeftel. *Calcul pseudo-différentiel et para-différentiel pour l'étude des conditions aux limites absorbantes et des propriétés qualitatives des EDP non linéaires*. PhD thesis, Université Paris 13, Paris, 2004.
10. V. Thomée. *Galerkin Finite Element Methods for Parabolic Problems*. Springer, Berlin, Heidelberg, New York, NY 1997.

Two-Level Methods for Blood Flow Simulation

Andrew T. Barker[1] and Xiao-Chuan Cai[2]

[1] Department of Mathematics, Center for Computation and Technology, Louisiana State University, Baton Rouge, LA 70803-4918, USA, andrewb@math.lsu.edu
[2] Department of Computer Science, University of Colorado, Boulder, CO 80309-0430, USA, cai@cs.colorado.edu

1 Introduction

We consider two-level Newton-Krylov-Schwarz algorithms for blood flow in arteries, which is a computationally difficult and practically important application area [6, 8]. In particular, the similar densities of blood and artery wall make the coupling between fluid and structure strong in both directions, so that partitioned or iterative procedures have difficulties due to the added-mass effect [4]. Instead of a partitioned procedure, we adopt a monolithic computational approach, coupling fluid to structure in one large system that is solved all at once. This tight coupling allows for robustness to parameters and makes our method immune to the added-mass effect. The resulting system is difficult to solve, but we show here that it can be solved efficiently with effective preconditioning strategies specifically designed for parallel computing.

2 Mathematical Model and Discretization

We solve the fully coupled and nonlinear equations for fluid-structure interaction with monolithic coupling of the three components, the fluid, the elastic wall structure, and the moving mesh.

Our visco-elastic model for the artery wall is

$$\rho_s \frac{\partial^2}{\partial t^2}\mathbf{x}_s = \nabla \cdot \sigma_s + \beta \frac{\partial}{\partial t}(\Delta \mathbf{x}_s) - \gamma \mathbf{x}_s \tag{1}$$

where $\mathbf{x}_s$ is the structural displacement, $\sigma_s = -p_s I + (2/3)E_s(\nabla \mathbf{x}_s + \nabla \mathbf{x}_s^T)$ is the Cauchy stress tensor that involves the unknown pressure p_s, ρ_s is the structure density, and β is a visco-elastic parameter. The γ term is included so that we can reproduce a standard fluid-structure test problem with one-dimensional structure as in [1]. To specify the grid displacements $\mathbf{x}_f$, we simply use the Laplace equation $\Delta \mathbf{x}_f = 0$ on the interior of the domain, following the practice in [10].

We model the fluid as a viscous incompressible Newtonian fluid, using the Navier–Stokes equations in the ALE frame

$$\left.\frac{\partial \mathbf{u}_f}{\partial t}\right|_Y + [(\mathbf{u}_f - \omega_g)\cdot\nabla]\mathbf{u}_f + \frac{1}{\rho_f}\nabla p_f = \nu_f \Delta \mathbf{u}_f, \tag{2}$$

$$\nabla \cdot \mathbf{u}_f = 0, \tag{3}$$

Here $\mathbf{u}_f$ is the fluid velocity vector and p_f is the pressure. The given data include the fluid density ρ_f and the kinematic viscosity $\nu_f = \mu_f/\rho_f$. The ALE mesh velocity is $\omega_g = \partial \mathbf{x}_f/\partial t$ and the Y indicates that the time derivative is to be taken in the ALE frame.

Boundary conditions for the fluid equations typically consist of a Dirichlet condition where $\mathbf{u}_f$ takes a given profile at the inlet Γ_i, and a zero traction condition $\sigma_f \cdot \mathbf{n}_f = \mu_f(\nabla \mathbf{u}_f \cdot \mathbf{n}_f) - p_f \mathbf{n}_f = 0$ at the outlet, where $\mathbf{n}_f$ is the unit outward normal. Here we have used $\sigma_f = -p_f I + \mu_f(\nabla \mathbf{u}_f)$.

The physical system, as well as our model and discretization, has strong coupling between the three fields. At the fluid-structure boundary we require that structure velocity match fluid velocity, $\mathbf{u}_f = \partial \mathbf{x}_s/\partial t$, which is a generalization of a no-slip, no penetration condition. We also enforce that the moving mesh must follow the solid displacement, so that the structure can maintain a Lagrangian description. This condition takes the form $\mathbf{x}_f = \mathbf{x}_s$. Again, this reduces to a homogeneous Dirichlet condition in the case of a rigid wall. Finally, we enforce the continuity of traction forces at the boundary. This can be written $\sigma_s \cdot \mathbf{n}_s = -\sigma_f \cdot \mathbf{n}_f$, where $\mathbf{n}_s, \mathbf{n}_f$ are the unit outward normal vectors for the solid and fluid domains, respectively, and σ_s and σ_f are the Cauchy stress tensors. The condition can be thought of as a Neumann-type condition on the structure model. It is important to emphasize that these coupling conditions are enforced implicitly as part of the monolithic system – they are never enforced as boundary conditions with given data from subproblems, as in the iterative coupling approach.

We discretize the coupled system with $Q_2 - Q_1$ finite elements for both fluid and structure. We discretize in time with the second order implicit trapezoid rule $y^{n+1} = y^n + (\Delta t/2)(F^{n+1} + F^n)$. For the sake of brevity, we skip the derivation of the weak form (which is standard) and present the fully discrete system. At each time step we solve a nonlinear system of the form

$$(\tilde{M} - (\Delta t/2)\tilde{K})y^{n+1} = (\tilde{M} + (\Delta t/2)\tilde{K})y^n \tag{4}$$

where

$$y^n = \begin{pmatrix} u_f \\ p_f \\ x_s \\ \dot{x}_s \\ p_s \\ x_f \end{pmatrix}^n, \quad \tilde{M} = \begin{pmatrix} M_f & & \\ & I & \\ & & \rho_s M_s \end{pmatrix}, \tag{5}$$

and

$$\tilde{K} = \begin{pmatrix} N(u_f) - \nu_f K_f & -Q_f^T & & & \\ Q_f & & & & \\ & & & I & \\ A_u & & A_p & K_s + \gamma M_s \;\; \beta K_s & -Q_s^T \\ & & & Q_s & \\ & & & & K_m \end{pmatrix}. \tag{6}$$

In the above, M_f, M_s are finite-element mass matrices for the fluid and structure, $N(u_f)$ represents the nonlinear term in (2), Q_f, Q_s are discrete divergence operators, and K_f, K_s, K_m are all finite-element stiffness matrices. A_u and A_p are discretizations of the traction-matching condition.

3 Two-Level Newton and Schwarz Methods

At each timestep we have to solve the nonlinear system (4). We solve this nonlinear system with a two-level Newton-Krylov-Schwarz algorithm – it is this algorithm that is the heart of this paper and the key to achieving parallel scalability and performance.

We first obtain an initial guess for the fine nonlinear problem (4) by solving an analogous nonlinear problem on the coarse grid and interpolating to the fine grid. Then we solve (4) with a standard inexact Newton method using that initial guess. At each iteration of Newton's method, we need to solve a linear Jacobian system, for which we use restarted flexible GMRES. The Newton method and the linear solver are standard, so we go on to describe the two-level hybrid preconditioner in detail.

Our hybrid preconditioner has two pieces – a one-level Schwarz domain decomposition component, and a coarse solve component. We describe the domain decomposition part first.

The domain decomposition portion, or fine-grid portion, of the two-level preconditioner is implemented as in [2]. We first partition the finite element mesh on Ω into several meshes on subdomains Ω_ℓ, each one corresponding to a processor of the parallel machine. Then we extend each subdomain Ω_ℓ to overlap its neighbors by a user-specified amount δ, and denote the overlapping domain by Ω'_ℓ. On each subdomain Ω'_ℓ we construct a subdomain operator B_ℓ, which is a restriction of the Jacobian matrix from the Newton solver. This portion of the preconditioner can be written

$$M_1^{-1} = \sum_{j=1}^{N} (R_j^0)^T B_j^{-1} R_j^\delta \tag{7}$$

where the B_j^{-1} are subdomain solves and the R_j are restriction and interpolation operators for the subdomains. Here R_ℓ^0 is a restriction operator that does not include overlap while R_ℓ^δ includes the overlap, which describes the restricted additive Schwarz variant [3]. The choice of subdomain solves and restriction and interpolation operators leads to different kinds of Schwarz preconditioners with different properties [9]. In our algorithm the $\tilde{B}_\ell^{-1}$ solves are done with LU factorization, which

is expensive, but since the subdomain solve is local to a single processor the preconditioner is scalable.

In our implementation, we combine the coarse-level and fine-level preconditioners multiplicatively, while continuing to use additive Schwarz within the fine level. You can write down the application of this hybrid preconditioner M_h^{-1} to a vector x in two steps

$$z = I_H^h B_0^{-1} I_h^H x, \tag{8}$$

$$M_h^{-1} x = z + M_1^{-1}(x - G_h' z) = z + \sum_{j=1}^{N} (R_j^0)^T B_j^{-1} R_j^\delta (x - G_h' z), \tag{9}$$

where I_h^H is a restriction from the fine grid to the coarse grid, $I_H^h = (I_h^H)^T$ is the corresponding interpolation operator from coarse grid to fine grid, and B_0^{-1} is a coarse-grid solve. In this hybrid preconditioner, the additive one-level component (9) means we can do the local subdomain solves in parallel, while we do the coarse and fine levels sequentially.

The coarse solve B_0^{-1} in (8) is normally parallel restarted GMRES, preconditioned with a one-level additive Schwarz method, using the same number of subdomains (and therefore processors) as the fine grid. The matrix that is being used in GMRES here is a Jacobian matrix, constructed independently on the coarse grid. That is, we solve (8) using the one-level algorithm described above. The only difference is that we can solve the coarse problem with a much larger error tolerance than the fine problem, saving computational cost while still being an effective preconditioner.

Using the same basic algorithm for the one-level method on the coarse as on the fine grid has two advantages. First, it is simpler to implement and allows us to reuse some data structures. And second, since we are using the full parallel collective to solve the coarse problem, it allows us to apply the preconditioner multiplicatively, since the coarse solve is done before the fine solve needs any data from it and vice versa. One potential disadvantage is the large number of subdomains of the coarse space, which could lead to the same ill-conditioning problem that drove us to use a two-level method in the first place. In practice, the coarse problem is easy enough to solve and the overlap (which is less costly to increase on the coarse grid) can be made sufficiently large to overcome this difficulty, though for very large simulations we may want to consider additional levels.

The fine and coarse grids in our implementation do not have any necessary connection to each other – they can be generated completely independently by mesh-generating software, and the interpolation and restriction between them is calculated when the program runs. In particular, the fine grid is not a refinement of the coarse grid. The fine grid is partitioned for the domain decomposition and parallel processing by Parmetis [7], and the coarse grid inherits that partition – the elements of the coarse grid are assigned to processors that contain nearby fine-grid elements.

4 Numerical Results

In this section we explore the implications of using a two-level Newton-Krylov-Schwarz method and the interplay of various parameters in that method, comparing to the one-level implementation as we go. We do simulations on a straight tube model, where we can verify results found in the literature and more carefully control the mesh size and number of unknowns, and also consider a more realistic branching artery model derived from clinical data. See [2] for a detailed verification of the same method with a less efficient preconditioner.

In the numerical results in this section, unless otherwise specified, we use an incompressible structure, the fluid density is $1{,}000\ \mathrm{kg/m^3}$, the damping parameter $\beta = 0.01$, and the kinematic viscosity of the fluid is $\nu_f = 0.0035\ \mathrm{kg/m\ s}$.

For the solver parameters, we consider the Newton solver on the fine level to have converged if the (absolute) residual is less than 10^{-6}. For fGMRES on the fine level, we have a relative tolerance that changes at each iteration, set by the Eisenstat–Walker method [5]. We restart flexible GMRES every 100 iterations.

We first test the method on a straight tube problem taken from [1]. We have a two-dimensional tube 6 cm by 1 cm, with walls at top and bottom of thickness 0.1 cm. A traction condition is applied at the left boundary to induce a pressure pulse, which then travels to the right, deforming the structure as it goes. In this example the Young's modulus $E_s = 7.5 \cdot 10^4\ \mathrm{kg/m\ s^2}$, the structure is incompressible and has a density of $1{,}100\ \mathrm{kg/m^3}$, and the inlet pressure pulse takes the form $\sigma_f \cdot \mathbf{n}_f = (-P_0/2)\,[1 - \cos((\pi t)/(.0025\mathrm{s}))]$ where $P_0 = 2.0 \cdot 10^5\ \mathrm{kg/m\ s^2}$. The timestep size is $\Delta t = 0.0001$ s.

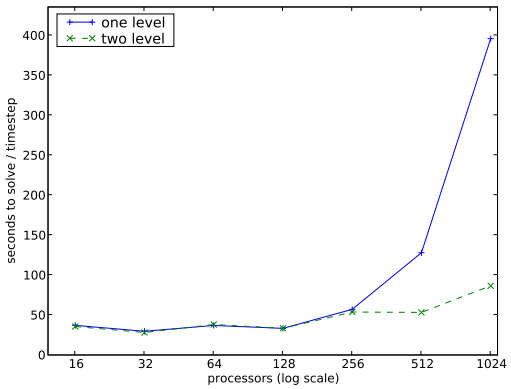


Fig. 1. Weak scaling for a straight tube test problem. The vertical axis shows average walltime in seconds per timestep of the simulation. The number of unknowns is proportional to the number of processors – 1,024 processors is $7.1 \cdot 10^6$ unknowns.

The primary motivation for the two-level preconditioner is to improve scalability for the most physically realistic cases, and we demonstrate that scalability in Fig. 1, which shows weak scaling for the straight tube example in the two-grid case, and where the scalability looks very good out to 1,024 processors. The linear iterations are also kept nearly constant for the two-level case in sharp contrast to the one-level preconditioner (results not shown).

Table 1. Effect of the coarse grid size on the solver behavior for the straight tube case. Coarse size is expressed as a fraction of the number of fine-grid unknowns, and coarse frac represents the proportion of compute time spent on the coarse grid.

Unknowns	np	Coarse size	Levels	fGMRES	Coarse frac	Walltime
$4.51 \cdot 10^5$	64	0.0	One	74.6	0.00	46.21
$4.51 \cdot 10^5$	64	0.03	Two	53.1	0.04	46.33
$4.51 \cdot 10^5$	64	0.12	Two	43.0	0.13	46.84
$7.97 \cdot 10^5$	128	0.0	One	123.2	0.00	41.08
$7.97 \cdot 10^5$	128	0.02	Two	86.9	0.06	42.87
$7.97 \cdot 10^5$	128	0.07	Two	68.7	0.11	43.49
$1.78 \cdot 10^6$	256	0.0	One	313.0	0.00	66.07
$1.78 \cdot 10^6$	256	0.01	Two	205.5	0.06	67.74
$1.78 \cdot 10^6$	256	0.03	Two	209.5	0.12	71.16
$3.16 \cdot 10^6$	512	0.0	One	882.7	0.00	78.27
$3.16 \cdot 10^6$	512	0.004	Two	$1.52 \cdot 10^3$	0.15	143.82
$3.16 \cdot 10^6$	512	0.02	Two	325.6	0.15	66.38
$3.16 \cdot 10^6$	512	0.04	Two	485.8	0.24	83.74
$7.09 \cdot 10^6$	1,024	0.0	One	$5.55 \cdot 10^3$	0.00	426.07
$7.09 \cdot 10^6$	1,024	0.02	Two	522.3	0.15	131.13
$7.09 \cdot 10^6$	1,024	0.03	Two	$4.17 \cdot 10^3$	0.38	548.94

Perhaps the most important implementation detail to consider in designing a two-level method is to choose the size of the coarse grid in order to balance the improvement in conditioning that comes from using a relatively fine coarse grid with the cost of solving the problem on the coarse grid. In Table 1, we present some comparisons of different coarse grid sizes.

In addition to the straight tube problem, we also use a pulmonary artery model taken from clinical data. Here we use a Young's modulus of $E_s = 3.0 \cdot 10^4$ kg/m s^2, and the structure is again incompressible and has a density of $1{,}000$ kg/m^3. We start the simulation from rest, with an impulsive Dirichlet inlet velocity condition of 0.05 m/s. In this more physically realistic and computationally challenging example, the difference in linear iteration counts between one- and two-level methods is even more marked. In Table 2, the two-level method results in a very sharp reduction in linear iterations and a good reduction in compute time for these problems. The two-level method can also be shown to be more robust to a variety of physical parameters.

Table 2. Solver characteristics for increasing number of subdomains, with fixed problem size (1.63 million unknowns) and fixed overlap parameter ($\delta = 0$ for two-level, $\delta = 3$ for one-level).

Subdomains	fGMRES Iterations		Walltime	
	one-level	two-level	one-level	two-level
96	442	237	270	184
112	514	245	277	182
128	487	286	216	163
160	697	282	201	105
192	899	485	168	109
224	1,040	349	152	91.1
256	1,020	382	127	79.9

Table 3. Overlap parameter comparisons for one-level and two-level methods on a branching grid.

Unknowns	np	Levels	δ	Newton	fGMRES	Walltime
$1.63 \cdot 10^6$	128	One	1	3.0	$2.35 \cdot 10^3$	406.32
$1.63 \cdot 10^6$	128	One	2	3.0	820.6	270.19
$1.63 \cdot 10^6$	128	One	3	3.0	487.4	214.43
$1.63 \cdot 10^6$	128	One	4	3.0	356.6	225.61
$1.63 \cdot 10^6$	128	Two	0	3.0	241.2	137.86
$1.63 \cdot 10^6$	128	Two	1	3.0	261.4	186.48
$1.63 \cdot 10^6$	128	Two	2	3.0	225.2	210.11
$1.63 \cdot 10^6$	128	Two	3	3.0	201.4	193.15
$1.63 \cdot 10^6$	128	Two	4	3.0	180.2	210.68
$2.40 \cdot 10^6$	256	One	2	3.0	$3.16 \cdot 10^3$	340.23
$2.40 \cdot 10^6$	256	One	3	3.0	$1.57 \cdot 10^3$	240.06
$2.40 \cdot 10^6$	256	One	4	3.0	$1.02 \cdot 10^3$	207.86
$2.40 \cdot 10^6$	256	Two	0	3.0	423.2	114.98
$2.40 \cdot 10^6$	256	Two	1	3.0	413.4	135.23
$2.40 \cdot 10^6$	256	Two	2	3.0	338.2	148.22
$2.40 \cdot 10^6$	256	Two	3	3.0	435.6	179.23
$2.40 \cdot 10^6$	256	Two	4	3.0	433.2	194.31

The overlap parameter δ in the Schwarz domain decomposition method is one way to adjust the strength of the preconditioner – a higher δ means more information transfer between subdomains, and therefore a faster convergence, but results in larger local problems. Another way to exchange information between subdomains is with a coarse grid, and in Table 3 it is clear that in the two-level method, the need to use overlap is greatly reduced.

5 Conclusion

In this paper we have developed and analyzed two-level Newton-Krylov-Schwarz methods for fluid-structure interaction in the simulation of blood flow. We have demonstrated effective, scalable parallel preconditioners for the fully coupled monolithic problem that allow complicated geometries with realistic parameter values.

References

1. S. Badia, A. Quaini, and A. Quarteroni. Splitting methods based on algebraic factorization for fluid-structure interaction. *SIAM J. Sci. Comput.*, 30(4):1778–1805, 2008.
2. A.T. Barker and X.-C. Cai. Scalable parallel methods for monolithic coupling in fluid-structure interaction with application to blood flow modeling. *J. Comput. Phys.*, 229: 642–659, 2010.
3. X.-C. Cai and M. Sarkis. A restricted additive Schwarz preconditioner for general sparse linear systems. *SIAM J. Sci. Comput.*, 21:792–797, 1999.
4. P. Causin, J.F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Comput. Methods Appl. Mech. Eng.*, 194(42–44):4506–4527, 2005.
5. S.C. Eisenstat and H.F. Walker. Choosing the forcing terms in an inexact Newton method. *SIAM J. Sci. Comput.*, 17:16–32, 1996.
6. L. Fatone, P. Gervasio, and A. Quarteroni. Multimodels for incompressible flows. *J. Math. Fluid Mech.*, 2(2):126–150, 2000.
7. G. Karypis. Metis/Parmetis web page, University of Minnesota, 2008. http://glaros.dtc.umn.edu/gkhome/views/metis.
8. C.A. Taylor and M.T. Draney. Experimental and computational methods in cardiovascular fluid mechanics. *Ann. Rev. Fluid Mech.*, 36:197–231, 2004.
9. A. Toselli and O. Widlund. *Domain Decomposition Methods—Algorithms and Theory*. Springer, Berlin, 2005.
10. A.M. Winslow. Adaptive-mesh zoning by the equipotential method. Technical Report, Argonne National Laboratory, 1981.

Newton-Krylov-Schwarz Method for a Spherical Shallow Water Model*

Chao Yang[1] and Xiao-Chuan Cai[2]

[1] Institute of Software, Chinese Academy of Sciences, Beijing 100190, P. R. China, yang@mail.rdcps.ac.cn
[2] Department of Computer Science, University of Colorado at Boulder, Boulder, CO 80309, USA, cai@cs.colorado.edu

1 Introduction

In this paper we study the application of Newton-Krylov-Schwarz method to fully implicit, fully coupled solution of a global shallow water model. In particular, we are interested in developing a scalable parallel solver when the shallow water equations (SWEs) are discretized on the cubed-sphere grid using a second-order finite volume method.

2 Governing Equations

The cubed-sphere grid of gnomonic type [7, 8] is used in this study. The grid is generated by mapping the six faces of an inscribed cube to the sphere surface using gnomonic projection. The six expanded patches are continuously attached together with proper boundary conditions. On each patch, the expressions of the SWEs in local curvilinear coordinates $(x, y) \in [-\pi/4, \pi/4]^2$ are identical. When no bottom topography is involved, the SWEs can be written in the following conservative form:

$$\frac{\partial Q}{\partial t} + \frac{1}{\Lambda}\frac{\partial(\Lambda F)}{\partial x} + \frac{1}{\Lambda}\frac{\partial(\Lambda G)}{\partial y} + S = 0, \tag{1}$$

with

$$Q = \begin{pmatrix} h \\ hu \\ hv \end{pmatrix}, \; F = \begin{pmatrix} hu \\ huu + \frac{1}{2}gg^{11}h^2 \\ huv + \frac{1}{2}gg^{12}h^2 \end{pmatrix}, \; G = \begin{pmatrix} hv \\ huv + \frac{1}{2}gg^{12}h^2 \\ hvv + \frac{1}{2}gg^{22}h^2 \end{pmatrix}, \; S = \begin{pmatrix} 0 \\ S_1 \\ S_2 \end{pmatrix},$$

* The first author was supported in part by NSFC grant 10801125, in part by 973 China grant 2005CB321702, and in part by 863 China grants 2006AA01A125. The second author was supported in part by DOE under DE-FC-02-06ER25784, and in part by NSF under grants CCF-0634894 and DMS 0913089.

and

$$S_1 = \Gamma_{11}^1(huu) + 2\Gamma_{12}^1(huv) + f\Lambda\left(g^{12}hu - g^{11}hv\right),$$
$$S_2 = 2\Gamma_{12}^2(huv) + \Gamma_{22}^2(hvv) + f\Lambda\left(g^{22}hu - g^{12}hv\right).$$

Here h is the fluid thickness, (u, v) are contravariant components of the fluid velocity, g is the gravitational constant and f is the Coriolis parameter due to the rotation of the sphere. The variable coefficients g^{mn}, Λ and Γ_{mn}^{ℓ} are only dependent on the curvilinear coordinates [12].

3 Discretizations

A uniform rectangular $N \times N$ grid is used on each patch. Grid cell $\mathcal{C}_{ij}$ is centered in (x_i, y_j), $i, j = 1, \cdots, N$, with grid size $\Delta x = \Delta y = \pi/2N$. The approximate solution in cell $\mathcal{C}_{ij}$ at time t is defined as

$$Q_{ij} \approx \frac{1}{\Lambda_{ij}\Delta x \Delta y} \int_{y_j-\Delta y/2}^{y_j+\Delta y/2} \int_{x_i-\Delta x/2}^{x_i+\Delta x/2} \Lambda(x, y)Q(x, y, t)\, d\,x\, d\,y,$$

where Λ_{ij} is evaluated at the cell center of $\mathcal{C}_{ij}$. Then we have the following semi-discrete system of the SWEs:

$$\frac{\partial Q_{ij}}{\partial t} + \frac{(\Lambda F)_{i+\frac{1}{2},j} - (\Lambda F)_{i-\frac{1}{2},j}}{\Lambda_{ij}\hbar} + \frac{(\Lambda G)_{i,j+\frac{1}{2}} - (\Lambda G)_{i,j-\frac{1}{2}}}{\Lambda_{ij}\hbar} + S_{ij} = 0. \quad (2)$$

Here the numerical fluxes are approximated using the Osher's Riemann solver [5, 6], i.e.,

$$(\Lambda F)_{i+\frac{1}{2},j} = \Lambda_{i+\frac{1}{2},j} F^{(o)}(Q^-_{i+\frac{1}{2},j}, Q^+_{i+\frac{1}{2},j}) = \Lambda_{i+\frac{1}{2},j} F(Q^*_{i+\frac{1}{2},j}),$$

with

$$h^* = \frac{1}{4gg^{11}}\left[\frac{1}{2}\left(u^- - u^+\right) + \sqrt{gg^{11}h^-} + \sqrt{gg^{11}h^+}\right]^2,$$
$$u^* = \frac{1}{2}\left(u^- + u^+\right) + \sqrt{gg^{11}h^-} - \sqrt{gg^{11}h^+},$$
$$v^* = \begin{cases} v^- + \frac{g^{12}}{g^{11}}\left(u^* - u^-\right), & \text{if } u^* \geq 0 \\ v^+ + \frac{g^{12}}{g^{11}}\left(u^* - u^+\right), & \text{otherwise,} \end{cases}$$

where we assume $|u| < \sqrt{gg^{11}h}$. The calculation of G follows an analogous way, see [12] for details. The following two reconstruction methods for constant states are considered in this study:

- Piecewise constant method (first order):

$$Q^-_{i+\frac{1}{2},j} = Q^+_{i-\frac{1}{2},j} = Q_{ij}. \quad (3)$$

- Piecewise linear method (second order):

$$Q^{-}_{i+\frac{1}{2},j} = Q_{ij} + \frac{Q_{i+1,j} - Q_{i-1,j}}{4}, \quad Q^{+}_{i-\frac{1}{2},j} = Q_{ij} - \frac{Q_{i+1,j} - Q_{i-1,j}}{4} \quad (4)$$

On each patch interface, one layer of ghost cells is needed and the numerical fluxes are calculated symmetrically across the interface to insure the numerical conservation of total mass, see [11] for details.

Given a semi-discrete system

$$\frac{\partial Q}{\partial t} + \mathcal{L}(Q) = 0,$$

we use the following second-order backward differentiation formula (BDF-2) for the temporal integration:

$$\frac{1}{2\Delta t}\left(3Q^{(m)} - 4Q^{(m-1)} + Q^{(m-2)}\right) + \mathcal{L}(Q^{(m)}) = 0. \quad (5)$$

Here $Q^{(m)}$ denotes Q evaluated at m-th time step with a fixed time step size Δt. Only at the first time step, a first-order backward Euler (BDF-1) is used.

4 Nonlinear Solver

Fully implicit method enjoys an advantage that the time-step size is no longer constrained by the CFL condition. The price to pay is that a large sparse nonlinear algebraic system has to be solved at each time step. In this study, we use Newton-Krylov-Schwarz (NKS) algorithm as the nonlinear solver.

In the NKS algorithm, to solve a nonlinear system $\mathcal{F}(X) = 0$, an inexact Newton's method is used in the outer loop. Let X_n be the approximate solution for the n-th Newton iterate, we find the next solution X_{n+1} as

$$X_{n+1} = X_n + \lambda_n s_n, \quad n = 0, 1, ... \quad (6)$$

where λ_n is the steplength decided by a linesearch procedure and s_n is the Newton correction. We then use the right-preconditioned GMRES (restarted every 30 iterations) method to solve the Jacobian system

$$J_n M^{-1}(M s_n) = -\mathcal{F}(X_n), \quad J_n = \mathcal{F}'(X_n)$$

until the linear residual $r_n = J_n s_n + \mathcal{F}(X_n)$ satisfies

$$\|r_n\| \leq \eta \|\mathcal{F}(X_n)\|.$$

We implement a hand-coded analytic method to generate the Jacobian J_n in the calculation. The accuracy (relative tolerance) of the Jacobian solver is determined uniformly by the nonlinear forcing terms $\eta = 10^{-3}$. Some more flexible methods

such as that of [2] may be used to get more efficient or more robust solutions. The Newton iteration (6) ends when the following stopping condition is satisfied

$$||\mathcal{F}(X_{n+1})|| \leq \max\{\varepsilon_r||\mathcal{F}(X_0)||, \varepsilon_a\},$$

where $\varepsilon_r, \varepsilon_a \geq 0$ are nonlinear tolerances.

To achieve uniform residual error at each time step, we use adaptive stopping conditions with both lower and upper adjustments in the NKS method. To do a lower adjustment, we do not let the iteration stop until

$$||\mathcal{F}(X_{n+1})|| \leq 1.0 \times 10^{-5}$$

even when the relative tolerance of $\varepsilon_r = 10^{-7}$ is satisfied. An upper adjustment can be done by setting the absolute tolerance to be $\varepsilon_a^{(0)} = 10^{-8}$ at the first time step and then lettting it adaptively be decided by

$$\varepsilon_a^{(m)} \leftarrow \max\{\varepsilon_a^{(m-1)}, ||\mathcal{F}(X^{(m-1)})||\},$$

where $X^{(m-1)}$ is the converged solution of previous time step.

The preconditioner M^{-1} is obtained by using the restricted additive Schwarz (RAS, [1, 9]) method based on the domain decomposition of the cubed-sphere described briefly here. The six patches of the cubed-sphere can be either simultaneously [12] or independently divided into non-overlapping subdomains. In this study, the six patches are treated in a separated way, i.e., the six patches are respectively decomposed into p non-overlapping rectangular subdomains. Each subdomain is then mapped onto one processor. Thus $6p$ is the total number of processors and subdomains as well. An overlapping decomposition can be obtained by extending each subdomain with δ layers of grid points in all directions. It should be noted that the overlapping area might lie on other patches and directions might also change.

In practice we use a point-wise ordering for both unknowns and the nonlinear equations, resulting in Jacobian matrices with 3×3-block entries. Subdomain solves are done by LU factorizations or incomplete LU (ILU) factorizations with fill-in level ℓ. Here the LU and the ILU factorizations are done in a point-wise manner, i.e., fill-ins are always 3×3 blocks rather than scalars.

5 Numerical Results

Our numerical tests are carried out on an IBM BlueGene/L supercomputer with $4{,}096$ nodes. Each node has a dual-core IBM PowerPC 440 processor running at 700 MHz and with 512 MB of memory. We use the 4-wave Rossby–Haurwitz problem in [10] as the test case in this study. The characteristic time and length scale is one day and the Earth's radius. The result on day 14 is provided in Fig. 1, consistent with the reference solutions in [3, 4].

To test the performance of the preconditioner, we use 192 processor cores to run a fixed size problem on a $512 \times 512 \times 6$ grid for 10 time steps with $\Delta t = 0.1$ days

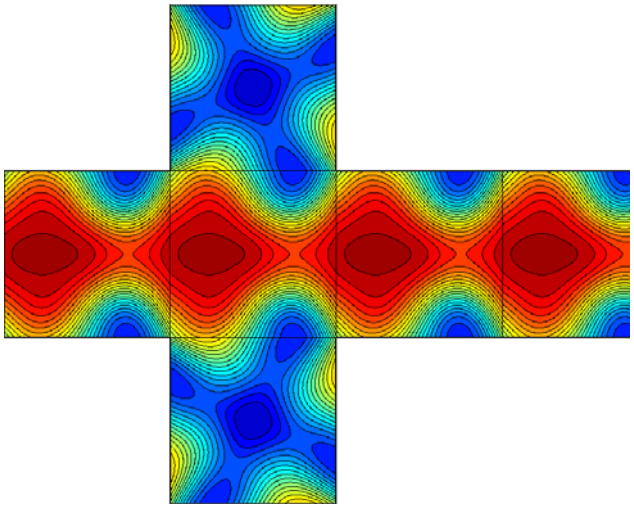


Fig. 1. Height field of Rossby–Haurwitz problem on day 14, grid size $128 \times 128 \times 6$, time step size $\Delta t = 0.1$ days. The contour levels are from $8{,}300$ to $10{,}500$ m with an interval of 100 m. The four innermost lines near to the equators are at $10{,}500$ m.

repeatedly with various levels of overlaps and fill-in ratios. First we try to use RAS preconditioner obtained directly from the Jacobian matrix J_n. In this case the ILU factorizations of the subdomain problems results in many GMRES iterations. If we use LU factorization instead, however, the factorization may fail due to insufficient memories and the performance is very poor even when the factorization succeeds.

Thus we use Jacobian matrix with first-order spatial discretization to construct the RAS preconditioner even when a high order scheme is used in the nonlinear function evaluation. This is based on the fact that Jacobian matrices are all related to the original SWEs no matter what spatial discretization is used. As it can be seen in Tables 1 and 2, the RAS precoditioner works in the NKS algorithm. Larger overlaps or subdomain ILU fill-ins help in reducing the number of GMRES iteration. However, the per-iteration work increases at the same time. The optimal choice in terms of computing time for this test is ILU(3) subdomain solvers with overlapping factor 2.

Using the optimal parameters, we run a set of large-scale tests with the same problem on a $1{,}024 \times 1{,}024 \times 6$ grid with gradually increased number of processor cores. As seen from Fig. 2, our solver scales up to 6,144 processor cores almost linearly with parallel efficiency 73.8%.

Table 1. The number of GMRES iterations per Newton iteration, averaged over the first 10 time steps.

Overlap	0	1	2	3	4
ILU(0)	309.5	299.2	294.4	292.7	291.4
ILU(1)	199.6	178.0	171.7	168.4	166.0
ILU(2)	194.1	150.2	141.0	139.3	137.8
ILU(3)	188.1	134.5	125.3	122.3	120.4
ILU(4)	184.9	127.0	116.1	111.8	110.9
LU	139.9	87.8	78.7	76.2	75.4

Table 2. The averaged compute time (in seconds) over the first 10 time steps.

Overlap	0	1	2	3	4
ILU(0)	10.13	11.13	11.38	11.65	11.90
ILU(1)	7.99	8.36	8.43	8.58	8.67
ILU(2)	8.13	7.84	7.79	7.94	8.09
ILU(3)	8.51	7.80	7.74	7.84	7.99
ILU(4)	8.90	7.96	7.83	7.87	8.04
LU	10.97	9.75	9.88	10.25	10.69

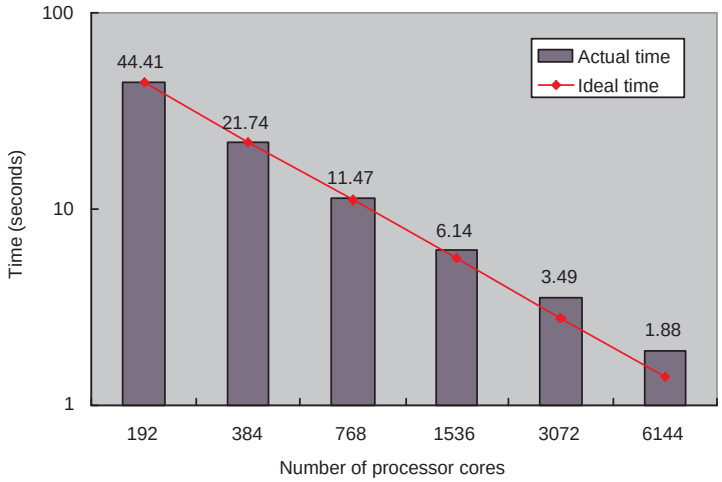


Fig. 2. Compute time curve on the Rossby–Haurwitz problem.

References

1. X.-C. Cai and M. Sarkis. A restricted additive Schwarz preconditioner for general sparse linear systems. *SIAM J. Sci. Comput.*, 21:792–797, 1999.
2. S.C. Eisenstat and H.F. Walker. Choosing the forcing terms in an inexact Newton method. *SIAM J. Sci. Comput.*, 17:1064–8275, 1996.

3. C. Jablonowski. *Adaptive Grids in Weather and Climate Modeling*. PhD thesis, University of Michigan, Ann Arbor, MI, 2004.
4. R. Jakob-Chien, J.J. Hack, and D.L. Williamson. Spectral transform solutions to the shallow water test set. *J. Comput. Phys.*, 119:164–187, 1995.
5. S. Osher and S. Chakravarthy. Upwind schemes and boundary conditions with applications to Euler equations in general geometries. *J. Comput. Phys.*, 50:447–481, 1983.
6. S. Osher and F. Solomon. Upwind difference schemes for hyperbolic systems of conservation laws. *Math. Comput.*, 38:339–374, 1982.
7. C. Ronchi, R. Iacono, and P. Paolucci. The cubed sphere: A new method for the solution of partial differential equations in spherical geometry. *J. Comput. Phys.*, 124:93–114, 1996.
8. R. Sadourny. Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids. *Mon. Wea. Rev.*, 100:211–224, 1972.
9. A. Toselli and O. Widlund. *Domain Decomposition Methods – Algorithms and Theory*. Springer, Berlin, 2005.
10. D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, and P.N. Swarztrauber. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *J. Comput. Phys.*, 102:211–224, 1992.
11. C. Yang and X.-C. Cai. A parallel well-balanced finite volume method for shallow water equations with topography on the cubed-sphere. *J. Comput. Appl. Math.*, 2010. to appear.
12. C. Yang, J. Cao, and X.-C. Cai. A fully implicit domain decomposition algorithm for shallow water equations on the cubed-sphere. *SIAM J. Sci. Comput.*, 32:418–438, 2010.

A Parallel Scalable PETSc-Based Jacobi-Davidson Polynomial Eigensolver with Application in Quantum Dot Simulation

Zih-Hao Wei[1], Feng-Nan Hwang[1], Tsung-Ming Huang[2], and Weichung Wang[3]

[1] Department of Mathematics, National Central University, Jhongli 320, Taiwan, socrates.wei@gmail.com; hwangf@math.ncu.edu.tw
[2] Department of Mathematics, National Taiwan Normal University, Taipei 116, Taiwan, min@math.ntnu.edu.tw
[3] Department of Mathematics, National Taiwan University, Taipei 106, Taiwan, wwang@math.ntu.edu.tw

Summary. The Jacobi-Davidson (JD) algorithm recently has gained popularity for finding a few selected interior eigenvalues of large sparse polynomial eigenvalue problems, which commonly appear in many computational science and engineering PDE based applications. As other inner–outer algorithms like Newton type method, the bottleneck of the JD algorithm is to solve approximately the inner correction equation. In the previous work, [Hwang, Wei, Huang, and Wang, A Parallel Additive Schwarz Preconditioned Jacobi-Davidson (ASPJD) Algorithm for Polynomial Eigenvalue Problems in Quantum Dot (QD) Simulation, Journal of Computational Physics (2010)], the authors proposed a parallel restricted additive Schwarz preconditioner in conjunction with a parallel Krylov subspace method to accelerate the convergence of the JD algorithm. Based on the previous computational experiences on the algorithmic parameter tuning for the ASPJD algorithm, we further investigate the parallel performance of a PETSc based ASPJD eigensolver on the Blue Gene/P, and a QD quintic eigenvalue problem is used as an example to demonstrate its scalability by showing the excellent strong scaling up to 2,048 cores.

1 Introduction

Many applications in computational science and engineering modeled by partial differential equations (PDEs) requires fast, accurate numerical solutions to the large-scale polynomial eigenvalue problems (EVPs), e.g., generalized EVPs in the linear stability analysis of incompressible flows and magnetohydrodynamics [4, 12, 13], quadratic EVPs in the vibration analysis of a fast train or the acoustic problem with damping [3, 5], and cubic or quintic EVPs in the estimate of discrete energy states and wave functions of the semiconductor quantum dot with non-parabolic band structure [9, 10].

The Jacobi-Davidson (JD) algorithm originally proposed by Sleijpen and Van der Vorst for linear EVPs, now has gained popularity for solving polynomial EVPs due

to several advantages. For examples, without recasting the polynomial EVPs as an enlarged linearized EVPs, one only needs to deal with the problem as the same size of the original one and the interior eigenvalues are targeted without using computational expensive shift-and-invert techniques. Moreover, the JD algorithm is parallelizable, hence it is suitable for large-scale eigenvalue computations.

The JD algorithm belongs to a class of subspace methods, which consists of two key steps: one first enlarges a subspace or a so-called search space by adding a new basis vector and then extract an approximate eigenpair from the search space through the Rayleigh-Ritz procedure. To obtain a new basis vector for the search space, at each JD iteration, one needs to solve approximately a large sparse linear system of equations, which is known as the correction equation, by an iterative method. In [8] the authors proposed a new algorithm, namely the additive Schwarz preconditioned Jacobi-Davidson algorithm (ASPJD) that imports an idea from the area of parallel Schwarz-Krylov solver to enhance the parallel scalability of the JD algorithms. The Schwarz methods [14] have been widely used and is well-understood for solving a variety of linear systems arising from the discretization of PDEs and is applied to nonlinear systems as a linear preconditioner for the Jacobian system in the Newton-Krylov-Schwarz algorithm [2] or as a nonlinear preconditioner in the additive Schwarz preconditioned inexact Newton algorithm [7]. On the other hand, however only a few studies are available in the literature for solving eigenvalue problems using Schwarz methods, e.g., Schwarz methods employed as the action of the spectral transformation in the Arnoldi methods for generalized EVPs [13] or as a preconditioner in the locally optimal block preconditioned conjugate gradient method [11].

In this paper, we continue the previous work investigating how the ASPJD algorithm performs on a parallel machine with a large number of processors, e.g., the Blue Gene/P. One of our target applications is a quintic polynomial EVPs arising from the semiconductor quantum dot simulation [7].

2 A Description of the ASPJD Algorithm

In this section, we briefly describe the ASPJD algorithm for solving polynomial eigenvalue problems of degree τ, which take the form of

$$\mathcal{A}(\lambda)x = \sum_{i=0}^{\tau} \lambda^i A_i x = 0, \tag{1}$$

where $A_i \in \mathbb{R}^{n\times n}$ are the large sparse matrices arising from some discretization of certain PDEs, $\lambda \in \mathbb{C}$ is an eigenvalue and $x \in \mathbb{C}^n$ is the corresponding eigenvector. The detailed algorithm in conjunction with other techniques, such as locking and restarting can be found in [8]. Let V be the current search space. Assume that (λ, u) is current the approximate eigenpair, which is not close enough to the exact one, (λ^*, u^*). Then the next eigenpair (λ_{new}, u_{new}) can be obtained through the following two steps:

Step 1 Update the search space $V = [V, v]$ by solving the **correction equation**.

$$\left(I - \frac{pu^*}{u^*p}\right) \mathcal{A}(\lambda)(I - uu^*)t = -r$$

approximately for $t \perp u$ by a Krylov subspace method with a preconditioner B_d^{-1} defined as

$$B_d = \left(I - \frac{pu^*}{u^*p}\right) B(I - uu^*) \approx \left(I - \frac{pu^*}{u^*p}\right) \mathcal{A}(\lambda)(I - uu^*)$$

Here $r = \mathcal{A}(\lambda)$ and $p = \mathcal{A}'(\lambda)u$, where $\mathcal{A}'(\theta) = \sum_{i=1}^{\tau} i\theta^{i-1}A_i$. Then t is orthogonalized against V, and v is defined as $v = t/\|t\|_2$.

Step 2 Perform the **Rayleigh-Ritz procedure** to extract (λ_{new}, u_{new}) from the search space V by solving the small projected PEP, $(V^T\mathcal{A}(\theta)V)s = 0$. Then set $\lambda_{new} = \theta$ and compute $u_{new} = Vs$.

In practice, one does not explicitly form B_d to perform the preconditioning operation, $z = B_d^{-1}y$ with $z \perp u$ for a given y, as it can be done equivalently by computing

$$z = B^{-1}y - \eta B^{-1}p, \text{ with } \eta = \frac{u^*B^{-1}y}{u^*B^{-1}p}$$

Note that the preconditioning operation $B^{-1}p$ and inner product $u^*B^{-1}p$ need to be computed only once for solving each correction equation and there is no need to re-compute them in the Krylov subspace iteration. Furthermore, in the ASPJD algorithm, the construction of the preconditioner B^{-1} is based on an additive Schwarz framework defined as follows.

Let $S = \{1, 2, ..., n\}$ be an index set and let each integer corresponds to one component of the eigenvector. Let $S_1, S_2, ..., S_N$ be an non-overlapping partition of S, i.e.

$$\cup_{i=1}^{N} S_i = S \quad \text{and} \quad S_i \cap S_j = \emptyset \quad i \neq j$$

To obtain an overlapping partition of S, we extend each S_i to a larger subset S_i^δ with the size of n_i, i.e. $S_i \subset S_i^\delta$. Here δ is a positive integer indicating the degree of overlap and in general $\sum_{i=1}^{N} n_i \geq n$. Using the overlapping partitions of S we define a subspace of $\mathbb{R}^n$, V_i^δ as

$$V_i^\delta = \{v | v = (v_1, ..., v_n)^T \in \mathbb{C}^n, v_k = 0 \text{ if } k \notin S_i^\delta\},$$

and the corresponding restriction operators, R_i^δ, which transfers data from $\mathbb{C}^n$ to V_i^δ. Then, the interpolation operator $(R_i^\delta)^T$ can be defined as the transpose of R_i^δ. Using the restriction operator, we define the one-level restricted additive Schwarz (RAS(δ)) preconditioner with the degree of overlapping δ as

$$B^{-1} = \sum_{i=1}^{N_s} (R_i^0)^T B_i^{-1} R_i^\delta,$$

where B_i^{-1} is the subspace inverse of B_i and $B_i = R_i^{\delta}\mathcal{A}(\lambda)(R_i^{\delta})^T$. Note that the block Jacobi preconditioner can be considered as a special case of the RAS preconditioner by setting the level of overlap equal to 0.

In the second step, we compute the eigenpair of the projected eigenvalue problem, $(V^T\mathcal{A}(\theta)V)s = 0$, by solving the corresponding linearized projected eigenvalue problem,

$$M_A z = \theta M_B z, \tag{2}$$

where

$$M_A = \begin{bmatrix} 0 & I & 0 & \dots & 0 \\ 0 & 0 & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & I \\ M_0 & M_1 & M_2 & \dots & M_{\tau-1} \end{bmatrix},$$

$$M_B = \begin{bmatrix} I & 0 & 0 & \dots & 0 \\ 0 & I & 0 & \dots & 0 \\ 0 & 0 & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & -M_\tau \end{bmatrix}, z = \begin{bmatrix} s \\ \theta s \\ \theta^2 s \\ \vdots \\ \theta^{\tau-1} s \end{bmatrix}.$$

Here $M_i = V^T A_i V$. Note that the dimension of $V^T\mathcal{A}(\theta)V$ is usually small and not larger than a user defined restarting number.

3 A PETSc-Based ASPJD Polynomial Eigensolver

The ASPJD algorithm was implemented using two powerful scientific software libraries, namely the PETSc [1] and the SLEPc [6]. As shown in Fig. 1, the design of PETSc adopts the principle of *software layering*. As an application code of PETSc, the major component in our ASPJD polynomial eigensolver, the JD object, is built on top of the KSP, a Linear Equation Solver. All PETSc libraries are based on Message Passing Interface (MPI) and two modules of linear algebra libraries: Basic Linear Algebra Subproblems (BLAS) and Linear Algebra Packages (LAPACK) library. The vector (Vec) and matrix (Mat) are two basic objects in PETSc. The eigenvector x and other working vectors are created as parallel vectors in the Vec object. The column vectors of V are stored as an array of parallel vectors. The coefficient matrices A_i and the matrix $\mathcal{A}(\theta)$ are created in a parallel sparse matrix format. We do explicitly form $\mathcal{A}(\theta)$ using parallel matrix–matrix addition and it is used in the construction of a RAS type preconditioner.

The fully parallel correction equation solve as described in Step 1 is the kernel of the JD algorithm. The ASPJD eigensolver employs a Krylov subspace method, such as GMRES or CG, which is provided by PETSc, in conjunction with the preconditioner, B_d^{-1}, where the RAS preconditioner B^{-1} is set to be a default one. For

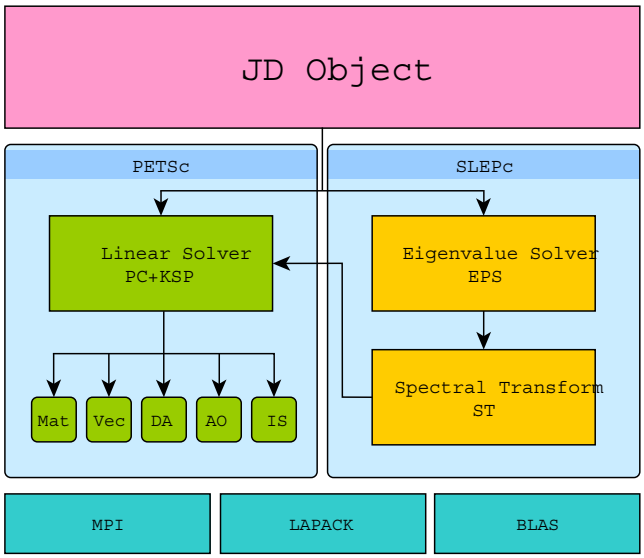


Fig. 1. The organization of PETSc, SLEPc, and the ASPJD eigensolver.

simplicity, in our current implementation both of the construction and the application of RAS are done internally by PETSc.

On each processor, the sequential QZ routine, called ZGGEVX in LAPACK, is employed to redundantly solve the same linearized projected eigenvalue problem, $M_A z = \theta M_B z$, through an interface provided by SLEPc [6]. Here, the matrices M_A and M_B, as well as M_i, are stored in the sequential dense matrix format and their sizes increase as ASPJD iterates.

4 Numerical Results

To demonstrate the scalability of our newly developed ASPJD eigensolver, we consider a quintic QD eigenvalue problem as a test case. The eigenvalue problem is derived from the second order finite volume discretization of the time-independent Schrödinger equation with non-parabolic effective mass, which is used to model a pyramidal InAs dot embedded in a cuboid GaAs matrix. The size of the resulting quintic QD eigenvalue problem is about 32 millions.

The numerical experiment was performed on the Blue Gene/P and all computation were done in double precision complex arithmetic. We claim that the JD iterations converge to an eigenpair if the absolute or the relative residuals $\|\mathcal{A}(\lambda)x\|$ is less than 10^{-10}. $V_{ini} = (1, 1, \ldots, 1)^T$ is normalized and set to be in the initial search space. We report the numerical results obtained by using the ASPJD algo-

rithm, where the correction equation is solved by right 20 (or 40) steps RAS(0) preconditioned GMRES incorporate with the ILU(0) as a subdomain solver for finding the smallest positive eigenvalue.

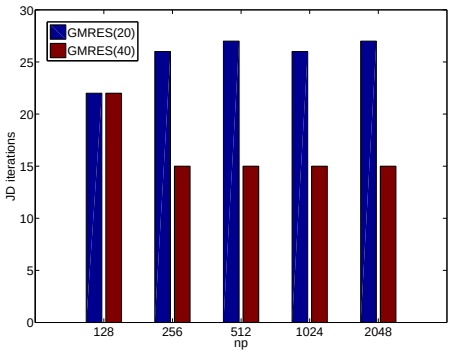


Fig. 2. The number of JD iterations with respect to np for the case of GMRES(20) and GMRES(40) as the correction equation solver.

Figure 2 shows the number of JD iterations of the ASPJD eigensolver with respect to the number of processor np, ranging from $np = 128$ to $np = 2{,}048$. We observe that except for the case of $np = 128$, the ASPJD eigensolver is quite algorithmically scalable, i.e., while the number of inner correction equation iterations is kept constant, the number of outer JD iterations remains almost the same with 26 and 15 JD iterations required to achieve convergence for the cases of GMRES(20) and GMRES(40), respectively. We may conclude that for this particular case, the number of JD iterations only depends on the number of GMRES iterations to be employed. A similar observation is made in [8] for the same test case but with a small size (about 1.5 M) and solved by the smaller number of processors (about $np = 320$).

It should be noted that the QD eigenvalue problem we consider has a special structure such that the eigenvectors corresponding to the eigenvalues of interest are localized to the dot. That is, the components of the eigenvector corresponding to the matrix (outside of the QD) are mostly zero. In our simulations, the ratio of the cuboid matrix to the pyramidal dot is about $35 : 1$ in the computational domain. Consequently, that is why we are able to decouple the original pyramidal QD eigenvalue problem problem into many subproblems using RAS(0) without a penalty in terms of an increased number of the JD iterations.

Figure 3 exhibits a very good strong scaling result for our ASPJD eigensolver for up to 2,048 processors. Note that by the definition, strong scaling means the execution time decreases in inverse proportion to the number of processors, provided that the problem size is fixed. In the ideal case, the slope of the curve is expected to be -1. The parallel efficiency for the case of GMRES(40) is about 80% based on the timing result obtained by using $np = 256$. Using a better grid partitioning

and taking the design of the network topology of the BG/P into account to reduce the communication cost might further improve the parallel scalability of the ASPJD eigensolver.

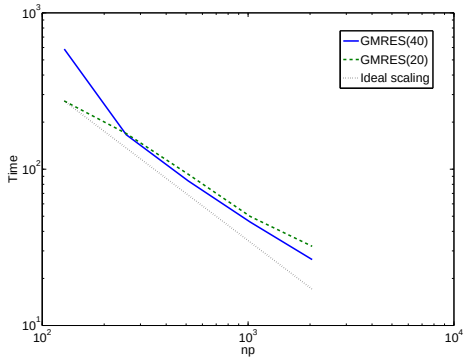


Fig. 3. Strong scalability of ASPJD on BG/P.

Acknowledgement. The authors are grateful to the BG/P computer sources provided by IBM during the workshop on computational science: IBM research and BG/P held at National Taiwan University during summer 2009. This work is partially supported by the National Science Council, the Taida Institute of Mathematical Sciences, and the National Center for Theoretical Sciences in Taiwan.

References

1. S. Balay, K. Buschelman, W.D. Gropp, D. Kaushik, M.G. Knepley, L.C. McInnes, B.F. Smith, and H. Zhang. PETSc webpage, 2010. http://www.mcs.anl.gov/petsc.
2. X.-C. Cai, W.D. Gropp, D.E. Keyes, R.G. Melvin, and D.P. Young. Parallel Newton-Krylov-Schwarz algorithms for the transonic full potential equation. *SIAM J. Sci. Comput.*, 19(1):246–265, 1998.
3. K.-W.E. Chu, T.-M. Hwang, W.-W. Lin, and C.-T. Wu. Vibration of fast trains, palindromic eigenvalue problems and structure-preserving doubling algorithms. *J. Comput. Appl. Math.*, 219:237–252, 2008.
4. K. Cliffe, H. Winters, and T. Garratt. Is the steady viscous incompressible two-dimensional flow over a backward-facing step at Re= 800 stable? *Int. J. Numer. Methods Fluids*, 17:501–541, 1993.
5. M.B. Van Gijzen. The parallel computation of the smallest eigenpair of an acoustic problem with damping. *Int. J. Numer. Methods Eng.*, 45:765–777, 1999.
6. V. Hernandez, J.E. Roman, and V. Vidal. SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems. *ACM Trans. Math. Softw.*, 31:351–362, 2005.

7. F.-N. Hwang and X.-C. Cai. A parallel nonlinear additive Schwarz preconditioned inexact Newton algorithm for incompressible Navier-Stokes equations. *J. Comput. Phys.*, 204:666–691, 2005. URL http://www.sciencedirect.com/science/article/B6WHY-4DVW0FD-3/2/17056653526b99d086bd799b21da26e4.
8. F.-N. Hwang, Z.-H. Wei, T.-M. Huang, and W. Wang. A parallel additive Schwarz preconditioned Jacobi-Davidson algorithm for polynomial eigenvalue problems in quantum dot simulation. *J. Comput. Phys.*, 229:2932–2947, 2010.
9. T.-M. Hwang, W.-W. Lin, J.-L. Liu, and W. Wang. Jacobi-Davidson methods for cubic eigenvalue problems. *Numer. Linear Algebra Appl.*, 12:605–624, 2005. URL http://dx.doi.org/10.1002/nla.423.
10. T.M. Hwang, W.C. Wang, and W. Wang. Numerical schemes for three-dimensional irregular shape quantum dots over curvilinear coordinate systems. *J. Comput. Phys.*, 226(1):754–773, 2007.
11. A.V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM J. Sci. Comput.*, 23:517–541, 2001.
12. M. Nool and A. van der Ploeg. A parallel Jacobi-Davidson-type method for solving large generalized eigenvalue problems in magnetohydrodynamics. *SIAM J. Sci. Comput.*, 22:95–112, 2000.
13. R.P. Pawlowski, A.G. Salinger, J.N. Shadid, and T.J. Mountziaris. Bifurcation and stability analysis of laminar isothermal counterflowing jets. *J. Fluid Mech.*, 551:117–139, 2006.
14. B.F. Smith, P.E. Bjørstad, and W. Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, Cambridge, 1996.

Two-Level Multiplicative Domain Decomposition Algorithm for Recovering the Lamé Coefficient in Biological Tissues

Si Liu[1] and Xiao-Chuan Cai[2]

[1] Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309, USA, sliu@colorado.edu
[2] Department of Computer Science, University of Colorado at Boulder, Boulder, CO 80309, USA, cai@cs.colorado.edu

1 Introduction

Tissue stiffness is one of the qualitative properties to distinguish abnormal tissues from normal tissues, and the stiffness changes are generally described in terms of the Lamé coefficient. In this paper, an all-at-once Lagrange-Newton-Krylov-Schwarz algorithm is developed to solve the inverse problem of recovering the Lamé coefficient in biological tissues. Specifically, we propose and study a multiplicative two-level domain decomposition preconditioner in the inexact Newton step. Numerical experiments are presented to show the efficiency and scalability of the algorithm on supercomputers.

2 Recovering the Lamé Coefficient in Biological Tissues

One of the signs in many diseases is abnormal tissue, of which shear stiffness differs greatly from that of normal tissue. Therefore, it is possible for scientists and engineers to develop new techniques for disease detection and diagnosis through reconstruction of high-resolution images of shear stiffness. In this paper, we focus on the inverse problem derived from transient elastography experiments. Previous work has shown that transient elastography experiments can determine the elastic wave displacement history through scans of the target tissue [2, 3, 5]. Our goal is to identify the Lamé coefficient that describes the shear wave speed or the mechanical stiffness changes inside the target tissue from the elastic wave time-dependent displacement.

The normalized $2D$ scalar wave equation that describes the shear wave displacement has the following form

$$\nabla \cdot (c_0^2 \, \rho \, \nabla d) - d_{tt} = 0, \tag{1}$$

with the boundary condition $\frac{\partial d}{\partial y} = g(t)$ at $y = 0$, where $d(x, y, t)$ describes the local time-dependent displacement inside the tissue, c_0 represents the speed of the background shear wave, and $g(t)$ describes the boundary source. ρ is called the Lamé coefficient, representing the stiffness profile of the tissue. Practically, ρ and d are generally twice continuously differentiable.

Without losing generality, we restrict ourselves to $2D$ domain problems:

$$\Omega = \{(x, y) \in \mathbb{R} \times \mathbb{R}, -6\,(\mathrm{cm}) \le x \le 6\,(\mathrm{cm}), 0\,(\mathrm{cm}) \le y \le 12\,(\mathrm{cm})\}$$

with piecewise smooth boundary $\Gamma = \partial\Omega$ and outer unit normal n. For convenience, the boundary Γ is separated into four pieces and is named as the North ($y = 0$), South ($y = 12$), West ($x = -6$), and East ($x = 6$) boundary, respectively.

We then take the Fourier transform of (1) and obtain the following Helmholtz equation:

$$-\nabla \cdot (\rho \nabla u) - k^2 u = 0, \text{ for } y > 0 \tag{2}$$

with the boundary conditions

$$\begin{cases} \dfrac{\partial u}{\partial r} = f(\tau), & \text{at } y = 0, \\ \lim\limits_{r\to\infty} \dfrac{1}{\sqrt{r}} \left(\dfrac{\partial u_s}{\partial r} - \tilde{i} k u_s \right) = 0, & \text{for the scattered field } u^s, \end{cases} \tag{3}$$

where u and f are the Fourier transform of d and g.

Here, the spatial variable r equals $\sqrt{x^2 + y^2}$. The wave number k equals τ/c_0. u is the total field $u = u^i + u^s$, which is the sum of the incident wave u^i and the scattered field u^s. u^i equals $\frac{1}{\tilde{i}k} f e^{\tilde{i}ky}$, where $\tilde{i}$ represents the imaginary unit $\sqrt{-1}$ throughout this paper.

Furthermore, the experiments of Catheline et al. show that there exists a dominant frequency represented as τ^*, called the central frequency [2, 3, 4]. The largest contribution of the Fourier transform is at this central frequency. Consequently, we evaluate the Eqs. (2) and (3) at the central frequency τ^* and arrive at the following equations:

$$\begin{cases} -\nabla \cdot (\rho \nabla u) - k^2 u = 0, & (x, y) \in \Omega \\ \dfrac{\partial u}{\partial n} = f, & \text{North boundary} \\ \dfrac{\partial u}{\partial n} - \tilde{i} k u = 0, & \text{South boundary} \\ \dfrac{\partial u}{\partial n} - \tilde{i} k u = -f e^{\tilde{i}ky}, & \text{East boundary and West boundary.} \end{cases} \tag{4}$$

We hereby focus on the inverse problem of recovering a high resolution image of the coefficient ρ from the observed data of u and the corresponding boundary conditions in (4).

3 Lagrange-Newton-Krylov-Schwarz Algorithm

To solve the inverse problem of recovering ρ from u, we apply the Tikhonov regularization method and solve the following minimization problem:

$$\text{minimize } J(\rho,u)=\frac{1}{2}\int_{\Omega}|u-z|^2\,d\Omega+\frac{\beta}{2}\int_{\Omega}|\nabla\rho|^2\,d\Omega, \tag{5}$$

where $z(x,y)$ denotes the measured value of $u(x,y)$. This minimization problem is subject to (4), and the H^1-semi-norm is applied as the regularization because of the continuous differentiability of ρ.

To solve this constrained optimization problem, we introduce the Lagrangian functional $\mathcal{L}$ as:

$$\begin{aligned}\mathcal{L}(\rho,u,\lambda)=&\frac{1}{2}\int_{\Omega}(u_1-z_1)^2\,d\Omega+\frac{1}{2}\int_{\Omega}(u_2-z_2)^2\,d\Omega+\frac{\beta}{2}\int_{\Omega}|\nabla\rho|^2\,d\Omega\\&+\int_{\Omega}\rho\nabla u_1\cdot\nabla\lambda_1\,d\Omega-\int_{\Omega}k^2u_1\lambda_1\,d\Omega\\&-\int_{\mathrm{N}}f\rho\lambda_1\,d\Gamma+\int_{\mathrm{S}}ku_2\rho\lambda_1\,d\Gamma+\int_{\mathrm{E\ \&\ W}}(ku_2+fcos(ky))\rho\lambda_1\,d\Gamma\\&+\int_{\Omega}\rho\nabla u_2\cdot\nabla\lambda_2\,d\Omega-\int_{\Omega}k^2u_2\lambda_2\,d\Omega\\&-\int_{\mathrm{S}}ku_1\rho\lambda_2\,d\Gamma-\int_{\mathrm{E\ \&\ W}}(ku_1-fsin(ky))\rho\lambda_2\,d\Gamma,\end{aligned} \tag{6}$$

where u_1 and u_2 represent the real and imaginary parts of u, z_1 and z_2 represent the real and imaginary parts of z, and λ_1 and λ_2 are the corresponding Lagrange multipliers.

The solution of the minimization problem is then obtained by solving the following saddle-point system:

$$\begin{cases}F^{(\rho)}\equiv\dfrac{\partial\mathcal{L}}{\partial\rho}\equiv-\beta\,\Delta\rho+\nabla Re(u)\cdot\nabla Re(\lambda)+\nabla Imag(u)\cdot\nabla Imag(\lambda)=0\\F^{(u)}\equiv\dfrac{\partial\mathcal{L}}{\partial u}\equiv(u-z)-\nabla\cdot(\rho\nabla\lambda)-k^2\lambda=0\\F^{(\lambda)}\equiv\dfrac{\partial\mathcal{L}}{\partial\lambda}\equiv-\nabla\cdot(\rho\nabla u)-k^2u=0.\end{cases} \tag{7}$$

With a finite difference discretization of the saddle-point system and a fully coupled ordering of the variables and the equations, we obtain a large system of nonlinear equations $F(X)=0$ [1]. This system is then solved by an inexact Newton method, and the Newton step is computed by:

$$\begin{aligned}&X_{k+1}=X_k+\xi_k\Delta X_k,\quad k=0,1,...\\&J(X_k)\Delta X_k=-F(X_k),\end{aligned} \tag{8}$$

where X_0 is an initial approximation, $J(X_k)=F'(X_k)$ is the Jacobian matrix at X_k, and ξ_k is the steplength determined by a linesearch procedure.

The generalized minimal residual method (GMRES) is applied to solve the Jacobian system, and the key step is to employ a good preconditioner. In our algorithm, the two-level multiplicative domain decomposition preconditioner is applied as a right preconditioner, and the preconditioning matrix is defined as:

$$M_{mult}^{-1} = A^{-1}[I - (I - AM_{AS}^{-1})(I - AM_c^{-1})(I - AM_{AS}^{-1})], \tag{9}$$

where M_{AS}^{-1} represents the one-level additive preconditioning matrix [1, 6]. M_c^{-1}, equal to $I_c^f A_c^{-1} R_f^c$, is derived from the inverse Jacobian matrix defined on a coarse mesh; I_c^f and R_f^c represent the restriction and interpolation operators.

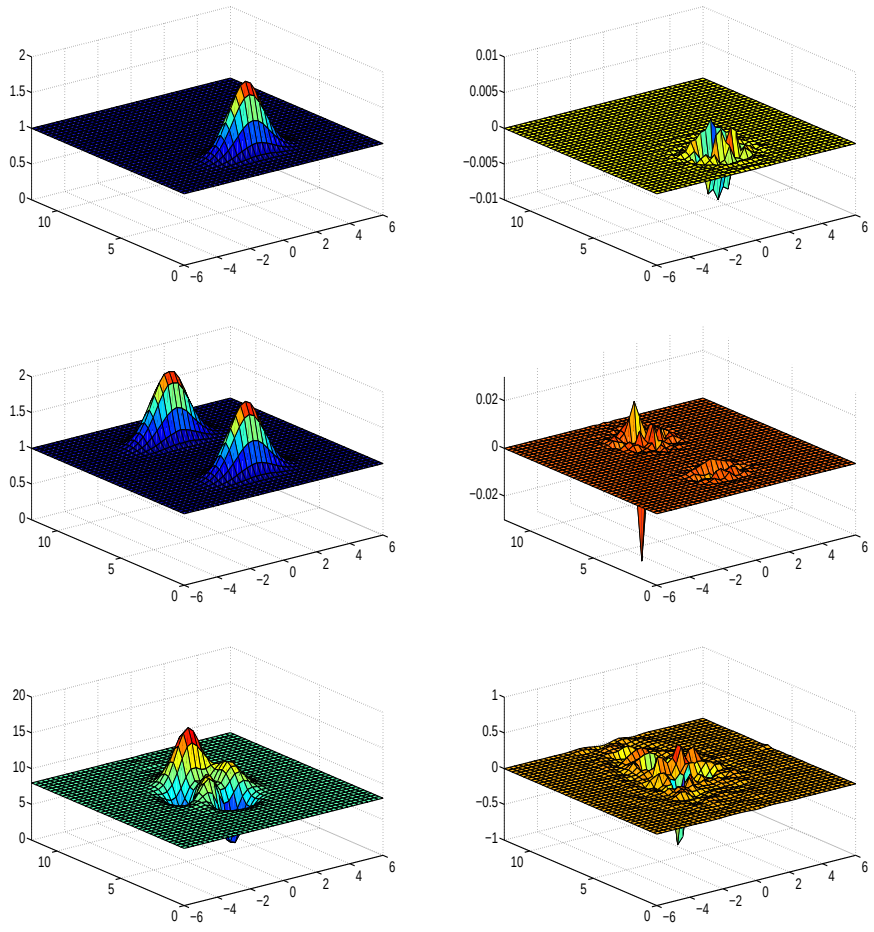


Fig. 1. This figure shows the numerical results of ρ in Test 1, 2, and 3 (from *top to bottom*) with wave number k equal to 8. The *left column* shows the numerical results of ρ, and the *right column* shows the difference between the numerical solutions and the analytic solutions.

4 Numerical Results and Discussion

Three different functions are tested in this paper. In Test 1, the Lamé coefficient to be identified is

$$\rho(x,y) = 1 + exp\left[-2(y-3)^2 - \frac{1}{2}x^2\right].$$

In Test 2, the Lamé coefficient to be identified is

$$\rho(x,y) = 1 + exp\left[-2(y-3)^2 - \frac{1}{2}x^2\right] + exp\left[-2(y-9)^2 - \frac{1}{2}x^2\right].$$

In Test 3, the Lamé coefficient to be identified is

$$\begin{aligned}\rho = {} & 3(1-x)^2 exp\left[-x^2 - (y-5)^2\right] - \frac{1}{3}exp\left[-(x+1)^2 - (y-6)^2\right] \\ & - 10\left[\frac{1}{5}x - x^3 - (y-6)^5\right] exp\left[-x^2 - (y-6)^2\right] + 8.\end{aligned}$$

To test the robustness of the algorithm, random noise is added to the observation data as $z^\delta = z\ (\ 1 + n_\epsilon\ rand(x,y)\ )$. In this paper, the noise level n_ϵ is chosen to be 0 or 1%.

We test our algorithm for different wave numbers, and the algorithm recovers the Lamé coefficient efficiently in all three test problems with wave numbers up to 15. The numerical solutions of ρ when k equals 8 are shown in Fig. 1. We also display the numerical solution of u for Test 3 when k equals 1, 8, or 15 in Fig. 2. This figure demonstrates that our algorithm recovers the Lamé coefficient well from observed data of low frequency, modest frequency, and high frequency.

To test the computing time and scalability of our algorithm, we define the problem on a fine mesh with $1{,}601 \times 1{,}601$ mesh points. The coarse mesh is chosen to be 81×81 or 101×101. When the observed data are only available on a coarse level, we interpolate the observed data to the fine mesh using the bilinear interpolation before we solve the inverse problems.

The number of Newton iterations, the average number of linear iterations, and the computing time are shown in Tables 1 and 2. Since the choice of the coarse mesh only affects the preconditioner, the number of Newton iterations is generally not changed. The number of average linear iterations rises slightly as the number of processors increases. The 101×101 coarse mesh provides more information than the 81×81 one and improves the preconditioner in our two-level algorithm, hereby saving almost 50% of the linear iterations. However, the increasing cost on the coarse level dramatically raises the computing cost per iteration. Therefore, total computing time is not saved and the scalability is worse.

The computing time and the strong scalability of our algorithm are shown in Fig. 3, where the observed data originally come from a 801×801 mesh. Linear and super-linear scalability are achieved for up to 400 processors. When the number of processors exceeds 900, we achieve over 75% scalability of the ideal case.

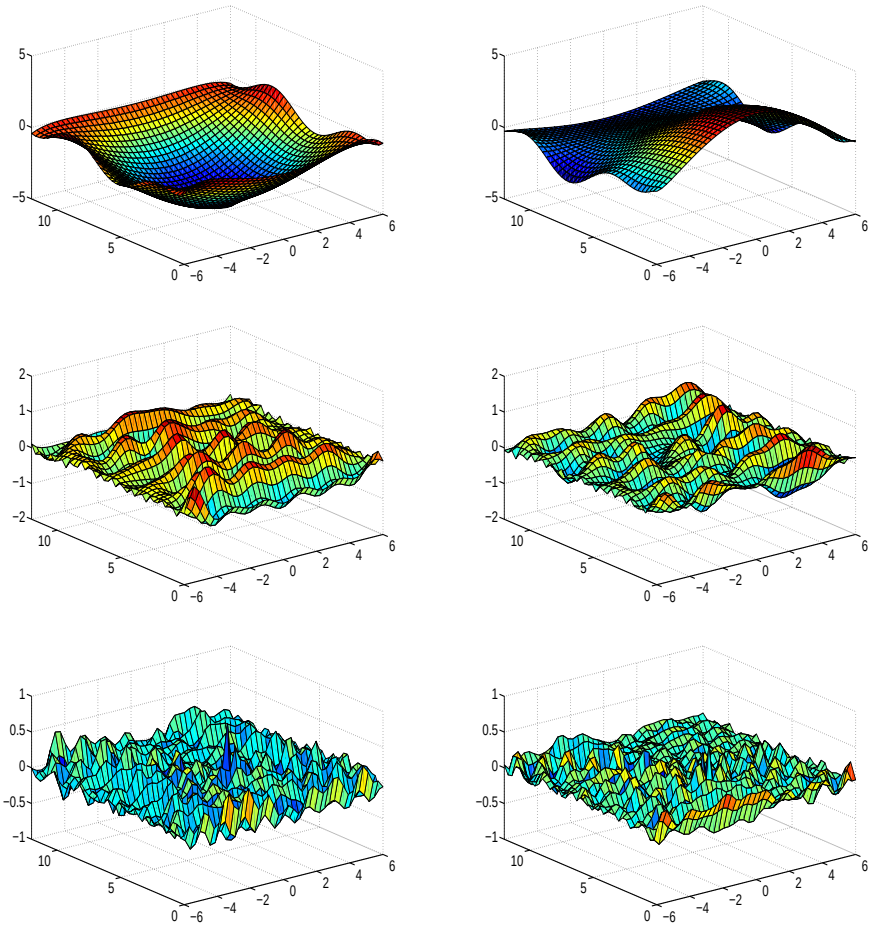


Fig. 2. This figure shows the numerical results of u in Test 3 for wave number k equal to 1, 8, and 15 from *top to bottom*. The *left column* shows the real part of the numerical solution of u, and the *right column* shows the imaginary part of the numerical solution of u.

Table 1. The table shows the numerical results of Test 3, when β equals 10^{-5} and n_ϵ equals 0%. The observed data are measured on a 801×801 mesh.

np	Newton	Average linear	Time(s)	Newton	Average linear	Time(s)
	Coarse mesh: 81×81			Coarse mesh: 101×101		
100	36	81.1	18,005.3	36	40.4	11,767.5
144	36	86.6	13,756.4	36	43.7	9,715.8
256	36	82.1	9,239.0	36	43.8	7,517.2
400	36	88.5	6,590.4	36	44.7	4,326.1
900	36	91.0	4,219.3	36	50.0	3,390.1
1,600	36	83.4	2,276.5	36	59.0	2,742.8

Table 2. The table shows the numerical results of Test 3, when β equals 10^{-5} and n_ϵ equals 1%. The observed data are measured on a 801×801 mesh.

np	Newton	Average linear	Time(s)	Newton	Average linear	Time(s)
	Coarse mesh: 81×81			Coarse mesh: 101×101		
100	38	74.2	17,730.9	38	39.0	12,077.8
144	38	68.9	12,014.1	38	39.6	9,856.3
256	38	70.3	8,573.7	38	41.7	7,664.3
400	38	77.8	6,245.4	38	43.0	4,449.3
900	38	72.8	3,697.6	38	47.4	3,453.4
1,600	38	71.2	2,137.4	38	52.5	2,679.8

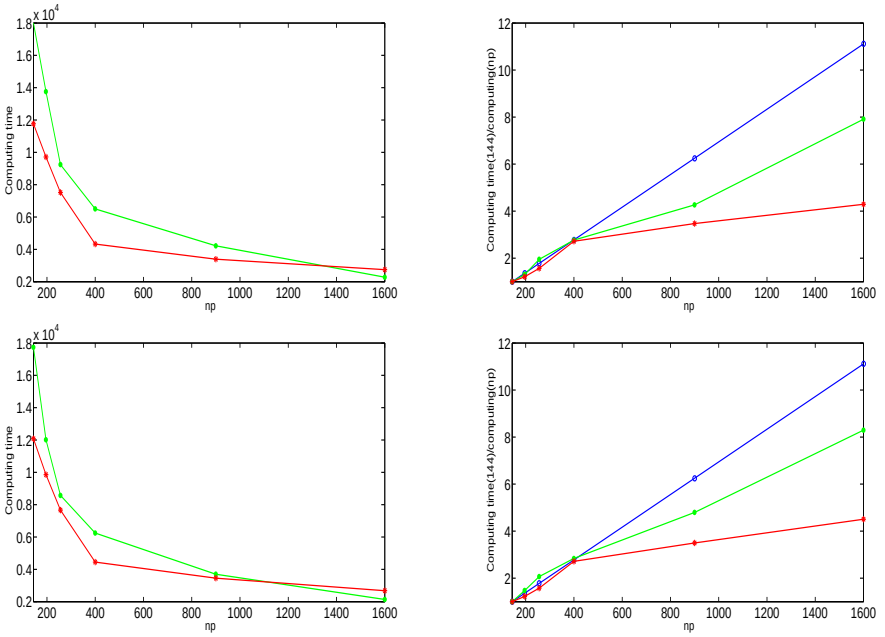


Fig. 3. This figure shows the computing time on the left and speedup curve on the right of Test 3. The noise level n_ϵ equals 0%(*top*) and 1%(*bottom*). $\circ$ represents the scalability of the ideal case. $\bullet$ and $*$ represent the results when the coarse level mesh is 81×81 and 101×101, respectively.

5 Concluding Remarks

In this paper, a two-level multiplicative domain decomposition algorithm is developed to solve this inverse problem of recovering the Lamé coefficient, which is usually difficult, expensive, and noise-sensitive. The algorithm can solve the inverse problem accurately and efficiently, when the observed data have random noise or are only available on a coarse mesh. The algorithm is fairly scalable considering the lin-

ear and nonlinear iteration numbers. Relatively scalable computing time is observed on supercomputers with up to $1{,}600$ processors.

References

1. X.-C. Cai, S. Liu, and J. Zou. Parallel overlapping domain decomposition methods for coupled inverse elliptic problems. *Commun. Appl. Math. Comput. Sci.*, 4:1–26, 2009.
2. S. Catheline, M. Tanter, F. Wu, and M. Fink. Diffraction field of a low frequency vibrator in soft tissues using transient elastography. IEEE Trans. Ultrason. Ferroelectn. Freq. Control, 46(4):1013–1019, 1999.
3. L. Ji and J. McLaughlin. Recovery of the Lamé parameter μ in biological tissues. *Inverse Probl.*, 20(1):1–24, 2004.
4. J. McLaughlin and D. Renzi. Shear wave speed recovery in transient elastography and supersonic imaging using propagating fronts. *Inverse Probl.*, 22(2):681–706, 2006.
5. L. Sandrin, M. Tanter, S. Catheline, and M. Fink. Shear modulus imaging with 2-d transient elastography. IEEE Trans. Ultrason. Ferroelectn. Freq. Control, 49(4):426–435, 2002.
6. A. Toselli and O. Widlund. *Domain Decomposition Methods—Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2005.

Robust Preconditioner for $\mathrm{H}(\mathrm{curl})$ Interface Problems

Jinchao Xu[1] and Yunrong Zhu[2]

[1] Department of Mathematics, Pennsylvania State University, University Park, PA 16802, USA, xu@math.psu.edu
[2] Department of Mathematics, University of California, San Diego (UCSD), La Jolla, CA 92093-0112, USA, zhu@math.ucsd.edu

Summary. In this paper, we construct an auxiliary space preconditioner for Maxwell's equations with interface, and generalize the HX preconditioner developed in [9] to the problem with strongly discontinuous coefficients. For the $\mathbf{H}(\mathbf{curl})$ interface problem, we show that the condition number of the HX preconditioned system is uniformly bounded with respect to the coefficients and meshsize.

Key words: HX preconditioner, AMG, $\mathbf{H}(\mathbf{curl})$ systems, Nédélec, interface

1 Introduction

The space $\mathbf{H}_0(\mathbf{curl})$ consists of square integrable vector fields with square integrable $\mathbf{curl}$ whose tangential component vanishes on $\partial\Omega$. In this paper, we try to develop robust and efficient preconditioners for the $\mathbf{H}(\mathbf{curl})$ interface problem:

$$\text{find } \mathbf{u} \in \mathbf{H}_0(\mathbf{curl}) : (\mu\mathbf{curl}\ \mathbf{u}, \mathbf{curl}\ \mathbf{v}) + (\sigma\mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v}),\ \forall \mathbf{v} \in \mathbf{H}_0(\mathbf{curl}). \quad (1)$$

Here, $\mathbf{f} \in \mathbf{L}^2(\Omega)$ is a vector field and the coefficients $\mu(x)$ and $\sigma(x)$ are assumed to be uniformly positive but may have large variations in a simply connected open polyhedral domain $\Omega \subset \mathbb{R}^3$.

This equation arises naturally from many engineering and physical applications based on Maxwell's equations. In some applications (see [12, 16] for example), the coefficients in (1) satisfy that $\mu(x)/\sigma(x) = c$ is the speed of light. In this case, Eq. (1) can be reduced to (2) by a simple scaling:

$$\text{find } \mathbf{u} \in \mathbf{H}_0(\mathbf{curl}) : (\omega\mathbf{curl}\ \mathbf{u}, \mathbf{curl}\ \mathbf{v}) + \tau(\omega\mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v}), \forall \mathbf{v} \in \mathbf{H}_0(\mathbf{curl}), \quad (2)$$

where $\tau \in (0, 1)$ is a constant, and $\omega > 0$ is piecewise constant but may possibly have large jump across the interfaces.

The finite element discretization of (2) reads:

$$\text{find } \mathbf{u}_h \in \mathbf{V}_h : (\omega \mathbf{curl}\ \mathbf{u}_h, \mathbf{curl}\ \mathbf{v}_h) + \tau(\omega \mathbf{u}_h, \mathbf{v}_h) = (\mathbf{f}, \mathbf{v}_h), \forall \mathbf{v}_h \in \mathbf{V}_h, \quad (3)$$

where $\mathbf{V}_h \subset \mathbf{H}_0(\mathbf{curl})$ is a conforming finite element space, e.g. Nedélec elements. It gives rise to the following linear system:

$$\mathbf{A}x = F, \quad (4)$$

where $\mathbf{A} = (a_{ij})$ is defined by $a_{ij} = \int_\Omega \omega \mathbf{curl}\ \mathbf{b}_j \cdot \mathbf{curl}\ \mathbf{b}_i + \tau \omega \mathbf{b}_j \cdot \mathbf{b}_i dx$ for any basis functions $\mathbf{b}_i, \mathbf{b}_j \in \mathbf{V}_h$. It is well-known that the operator curl has a large kernel, which should be taken into account in the development of efficient solvers. This kernel causes most existing AMG solvers for Poisson equations to fail; see [23] for a theoretical explanation. In order to deal with this issue, most work has been done for developing efficient solvers for (4) with constant coefficients; see [2, 8, 11, 15, 18, 19].

Recently, Hiptmair and Xu [9] proposed an innovative approach for solving $\mathbf{H}(\mathrm{curl})$ systems, known as the HX-preconditioner. It relies on a *regular decomposition* of $\mathbf{H}(\mathrm{curl})$ vector fields (see Sect. 2) and the framework of the auxiliary space method (cf. [22]). A related method, which is based on the compatible discretization framework, was introduced in [4]. Although the analysis in [9] is only for constant coefficients case, extensive numerical experiments (cf. [13, 14]) demonstrate that this preconditioner is also efficient and robust for general coefficients. It is the purpose of this paper to give an theoretical justification of the robustness of the HX-preconditioner for (3).

The remainder of this paper is organized as follows. In Sect. 2, we discuss the regular decompositions at the continuous level. In particular, we prove the regular decomposition in a weighted norm. Then in Sect. 3, we adapt the decomposition into a discrete form, develop the HX preconditioner, and prove its robustness.

2 Regular Decomposition

The theoretical foundation in the development of the HX preconditioner is the following theorem, which originates from [3, 7] for Maxwell's equations.

Theorem 1 ([10, 17]). *For any* $\mathbf{u} \in \mathbf{H}_0(\mathbf{curl})$ *there exist* $\mathbf{\Phi} \in \mathbf{H}_0^1(\Omega)$ *and* $p \in H_0^1(\Omega)$ *such that* $\mathbf{u} = \mathbf{\Phi} + \nabla p$, *which satisfy the following stability estimates:*

$$\|\mathbf{\Phi}\|_{1,\Omega} \lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega}, \text{ and } \|\nabla p\|_{0,\Omega} \lesssim \|\mathbf{u}\|_{\mathbf{H}(\mathbf{curl})}.$$

This theorem states that roughly speaking, the gap between $\mathbf{H}_0^1(\Omega)$ and $\mathbf{H}_0(\mathbf{curl})$ can be bridged by contributions from the kernel of $\mathbf{curl}$.

In some circumstances, the $\mathbf{H}(\mathbf{curl})$ systems are imposed with mixed boundary conditions. To deal with this situation, we consider the regular decomposition for the vector fields in the Hilbert space

$$\mathbf{H}_\Gamma(\mathbf{curl}) := \{\mathbf{u} \in \mathbf{H}(\mathbf{curl}) : \mathbf{u} \times \mathbf{n}|_\Gamma = 0, \ \text{ for } \Gamma \subset \partial\Omega\},$$

where $\Gamma \neq \emptyset$ is the Dirichlet boundary. We have a similar regular decomposition for $\mathbf{u} \in \mathbf{H}_\Gamma(\mathbf{curl})$ as follows:

Theorem 2. *For any* $\mathbf{u} \in \mathbf{H}_\Gamma(\mathbf{curl})$ *there exist* $\mathbf{\Phi} \in \mathbf{H}^1_\Gamma(\Omega)$ *and* $p \in H^1_\Gamma(\Omega)$ *such that*

$$\mathbf{u} = \mathbf{\Phi} + \nabla p.$$

This decomposition satisfies

$$\|\mathbf{\Phi}\|_{1,\Omega} \lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega}, \text{ and } \|\nabla p\|_{0,\Omega} \lesssim \|\mathbf{u}\|_{\mathbf{H}(\mathbf{curl})}.$$

Proof. We need to take special care of the boundary conditions. Without loss of generality, we assume that Γ is simply connected (otherwise, we just treat different connected components similarly). Let $\widetilde{\Omega}$ be a ball such that $\Omega \subset\subset \widetilde{\Omega}$, and $\widetilde{\Omega} = \Omega \cup O_\Gamma \cup O$ where O_Γ is the subdomain with $\partial O_\Gamma \cap \partial\Omega = \Gamma$, and $O = \widetilde{\Omega} \backslash (\Omega \cup O_\Gamma)$ (see Fig. 1). We extend $\mathbf{u}$ to $\bar{\mathbf{u}} \in \mathbf{H}_0(\mathbf{curl}, \widetilde{\Omega})$ defined by $\bar{\mathbf{u}}|_\Omega := \mathbf{u}$, $\bar{\mathbf{u}}|_{O_\Gamma} := 0$. On the subdomain O, we define $\bar{\mathbf{u}}$ as the $\mathbf{H}(\mathbf{curl})$ extension of $\mathbf{u}$ such that $\bar{\mathbf{u}}|_{\partial\Omega\backslash\Gamma} = \mathbf{u}|_{\partial\Omega\backslash\Gamma}$ and 0 on the remaining boundary of O. We refer to [1] for the existence of such an extension. The remainder of the proof is almost identical to that of Theorem

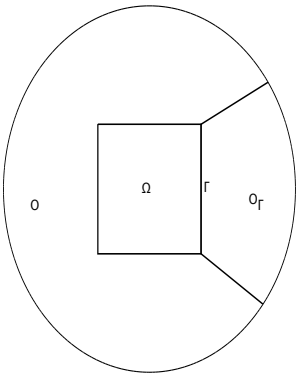


Fig. 1. Extension of $\mathbf{u} \in \mathbf{H}_\Gamma(\mathbf{curl}, \Omega)$ to $\bar{\mathbf{u}} \in \mathbf{H}_0(\mathbf{curl}, \widetilde{\Omega})$.

1 (see [17] for example). We omit the details.

Remark 1. For some other geometric structure of Γ, Theorem 2 still holds, for example if Γ is a closed surface, or a "screen" (see [6, 16]).

In order to deal with the interface problem (2), we consider the regular decomposition for $\mathbf{H}(\mathbf{curl})$ in the setting of the weighted norms, which are the natural norm to deal with the interface problems. More precisely, we denote

$$\|v\|^2_{0,\omega} = \int_\Omega \omega |v|^2 dx, \quad |v|^2_{1,\omega} = \int_\Omega \omega |\nabla v|^2 dx \text{ and } \|v\|^2_{1,\omega} = \|v\|^2_{0,\omega} + |v|^2_{1,\omega}.$$

For simplicity, let $\overline{\Omega} = \overline{\Omega}_1 \cup \overline{\Omega}_2$, where in Ω_1 and Ω_2 the equation has different constant coefficients ω_1, ω_2, respectively (see Fig. 2), with $\omega_1 \geq \omega_2 > 0$. The main

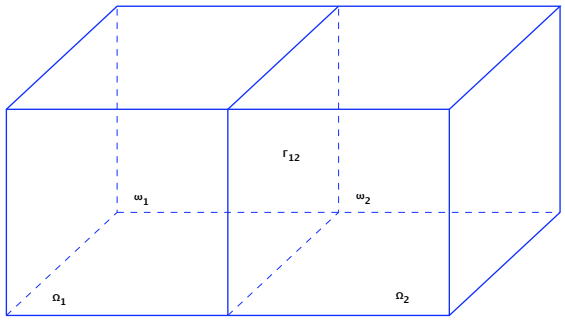


Fig. 2. Two domains with $\omega_1 \geq \omega_2 > 0$.

result of this section is the following decomposition. The idea of the proof is similar to the one used in [12] for proving a weighted Helmholtz decomposition.

Theorem 3. *For any* $\mathbf{u} \in \mathbf{H}_0(\mathbf{curl})$, *we have* $\mathbf{u} = \mathbf{\Phi} + \nabla p$, *where* $\mathbf{\Phi} \in \mathbf{H}_0^1(\Omega)$ *and* $p \in H_0^1(\Omega)$ *such that*

$$\|\mathbf{\Phi}\|_{1,\omega}^2 \lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\omega}^2 \text{ and } \|\nabla p\|_{0,\omega}^2 \lesssim \|\mathbf{u}\|_{0,\omega}^2 + \|\mathbf{curl}\ \mathbf{u}\|_{0,\omega}^2.$$

Proof. First we apply Theorem 2 on Ω_1 with the Dirichlet boundary $\Gamma_1 = \partial\Omega \cap \partial\Omega_1$. For given $\mathbf{u} \in \mathbf{H}_0(\mathbf{curl})$, we have, $\mathbf{u}|_{\Omega_1} = \mathbf{\Phi}_1 + \nabla p_1$ with $\mathbf{\Phi}_1 \in \mathbf{H}_{\Gamma_1}^1(\Omega_1)$ and $p_1 \in H_{\Gamma_1}^1(\Omega_1)$ such that

$$\|\mathbf{\Phi}_1\|_{1,\Omega_1} \lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_1} \quad \text{and} \quad \|\nabla p_1\|_{0,\Omega_1} \lesssim \|\mathbf{u}\|_{\mathbf{H}(\mathbf{curl},\Omega_1)}. \tag{5}$$

Let $\Gamma_{12} = \partial\Omega_1 \cap \partial\Omega_2$ be the interface. We then extend $\mathbf{\Phi}_1$ and p_1 to harmonic functions on Ω_2, and denote these extensions by $\widetilde{\mathbf{\Phi}}_1$ and $\widetilde{p}_1$. By the properties of harmonic extension (cf. [20]), the trace theorem and (5), we obtain

$$\begin{aligned}
\|\widetilde{\mathbf{\Phi}}_1\|_{1,\Omega_2} &\lesssim \|\mathbf{\Phi}_1\|_{\frac{1}{2},\Gamma_{12}} \lesssim \|\mathbf{\Phi}_1\|_{1,\Omega_1} \lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_1}, \\
\|\widetilde{p}_1\|_{1,\Omega_2} &\lesssim \|p_1\|_{\frac{1}{2},\Gamma_{12}} \lesssim \|p_1\|_{1,\Omega_1} \lesssim \|\mathbf{u}\|_{\mathbf{H}(\mathbf{curl},\Omega_1)}.
\end{aligned}$$

Now notice that on Ω_2, we have $\mathbf{u}_2^0 = \mathbf{u}|_{\Omega_2} - (\widetilde{\mathbf{\Phi}}_1 + \nabla\widetilde{p}_1)|_{\Omega_2} \in \mathbf{H}_0(\mathbf{curl}, \Omega_2)$. Then by Theorem 1 we get the decomposition $\mathbf{u}_2^0 = \mathbf{\Phi}_2^0 + \nabla p_2^0$ with $\mathbf{\Phi}_2^0 \in \mathbf{H}_0^1(\Omega_2)$ and $p_2^0 \in H_0^1(\Omega_2)$. This decomposition of $\mathbf{u}_2^0$ satisfies:

$$\begin{aligned}
\|\mathbf{\Phi}_2^0\|_{1,\Omega_2} &\lesssim \|\mathbf{curl}\ \mathbf{u}_2^0\|_{0,\Omega_2} \leq \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_2} + \|\mathbf{curl}\ \widetilde{\mathbf{\Phi}}_1\|_{0,\Omega_2} \\
&\leq \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_2} + \|\widetilde{\mathbf{\Phi}}_1\|_{1,\Omega_2} \lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_2} + \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_1},
\end{aligned}$$

and similarly $\|\nabla p_2^0\|_{0,\Omega_2} \lesssim \|\mathbf{u}\|_{\mathbf{H}(\mathbf{curl},\Omega)}$. Let the decomposition of $\mathbf{u}$ in the whole domain be $\mathbf{u} = \mathbf{\Phi} + \nabla p$ where

$$\mathbf{\Phi} = \begin{cases} \mathbf{\Phi}_1 & \text{in } \Omega_1 \\ \mathbf{\Phi}_2^0 + \widetilde{\mathbf{\Phi}}_1 & \text{in } \Omega_2 \end{cases} \quad \text{and} \quad p = \begin{cases} p_1 & \text{in } \Omega_1 \\ p_2^0 + \widetilde{p}_1 & \text{in } \Omega_2 \end{cases}.$$

Recalling that $\omega_1 \geq \omega_2 > 0$, this decomposition satisfies

$$\begin{aligned} \|\mathbf{\Phi}\|_{1,\omega}^2 &\leq \omega_1 \|\mathbf{\Phi}_1\|_{1,\Omega_1}^2 + \omega_2 \|\mathbf{\Phi}_2^0\|_{1,\Omega_2}^2 + \omega_2 \|\widetilde{\mathbf{\Phi}}_1\|_{1,\Omega_2}^2 \\ &\lesssim \omega_1 \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_1}^2 + \omega_2 \left(\|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_2}^2 + \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_1}^2 \right) + \omega_2 \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_1}^2 \\ &= \left(1 + \frac{2\omega_2}{\omega_1} \right) \omega_1 \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_1}^2 + \omega_2 \|\mathbf{curl}\ \mathbf{u}\|_{0,\Omega_2}^2 \\ &\lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\omega}, \end{aligned}$$

and similarly,

$$\begin{aligned} \|\nabla p\|_{0,\omega}^2 &\leq \omega_1 \|\nabla p_1\|_{0,\Omega_1}^2 + \omega_2 \|\nabla p_2^0\|_{0,\Omega_2}^2 + \omega_2 \|\nabla \widetilde{p}_1\|_{0,\Omega_2}^2 \\ &\lesssim \omega_1 \|\mathbf{u}\|_{\mathbf{H}(\mathbf{curl},\Omega_1)}^2 + \omega_2 \|u\|_{\mathbf{H}(\mathbf{curl},\Omega)}^2 \\ &\lesssim \|\mathbf{curl}\ \mathbf{u}\|_{0,\omega}^2 + \|\mathbf{u}\|_{0,\omega}^2. \end{aligned}$$

This completes the proof.

Remark 2. The above result can be generalized to more general interface problems. For example, to cases where the subdomains have no "cross edge", that is, there is no edge which belongs to more than two subdomains. In these cases, the same conclusion holds because the coefficients satisfy a certain monotonicity.

3 Auxiliary Space Preconditioners

To realize the preconditioners for the finite element discretization of the model equations (1), the decomposition discussed in the previous section should be adapted to the discrete setting.

The degrees of freedom specified for $\mathbf{V}_h$ determine the *nodal interpolation operator* $\mathbf{\Pi}_h$, defined by $\mathbf{\Pi}_h \mathbf{v} = \sum_{e \in \mathcal{E}_h} \left(\int_e \mathbf{v} \cdot dl \right) \mathbf{b}_e$, where $\mathcal{E}_h$ is the set of (interior) edges and $\mathbf{b}_e$ is the edge element basis function associated with the edge e. In the sequel, we let $S_h \subset H_0^1(\Omega)$ be the standard nodal finite element space and $\mathbf{S}_h \subset \mathbf{H}_0^1(\Omega)$ be the vector counterpart of S_h. Due to the local approximation property of $\mathbf{\Pi}_h$, we have the following standard estimate.

Lemma 1. *For any* $\mathbf{\Phi} \in \mathbf{H}_0^1(\Omega)$ *such that* $\mathbf{curl}\ \mathbf{\Phi} \in \mathbf{curl}\ \mathbf{V}_h$, *the interpolation operator* $\mathbf{\Pi}_h$ *satisfies*

$$\mathbf{curl}\ (\mathbf{\Pi}_h \mathbf{\Phi}) = \mathbf{curl}\ \mathbf{\Phi} \text{ and } \|h^{-1} (I - \mathbf{\Pi}_h) \mathbf{\Phi}\|_{0,\omega} \lesssim \|\mathbf{\Phi}\|_{1,\omega}.$$

Based on Theorem 3 and Lemma 1, we obtain the following main result.

Theorem 4. *For any $\mathbf{v}_h \in \mathbf{V}_h$ there exist $\mathbf{\Phi}_h \in \mathbf{S}_h$, $p_h \in S_h$ and $\widetilde{\mathbf{v}}_h \in \mathbf{V}_h$ such that $v_h = \widetilde{\mathbf{v}}_h + \mathbf{\Pi}_h \mathbf{\Phi}_h + \nabla p_h$, and for any constant $\tau \in (0,1)$*

$$\|(h^{-1} + \tau^{\frac{1}{2}})\widetilde{\mathbf{v}}_h\|_{0,\omega}^2 + |\!|\!|\mathbf{\Phi}_h|\!|\!|_\tau^2 + \tau\,|p_h|_{1,\omega}^2 \lesssim \|\mathbf{v}_h\|_A^2, \tag{6}$$

where $\|\mathbf{v}_h\|_A^2 = \int_\Omega \omega|\mathbf{curl}\ \mathbf{v}|^2 + \tau\omega|\mathbf{v}|^2 dx$ and $|\!|\!|\mathbf{w}|\!|\!|_\tau^2 = \int_\Omega \omega|\nabla \mathbf{w}|^2 + \tau\omega|\mathbf{w}|^2 dx$.

Proof. Notice that if $\mathbf{v}_h \in \mathbf{V}_h$, by Theorem 3 and Lemma 1 there exists a $\mathbf{\Phi} \in \mathbf{H}_0^1(\Omega)$ such that $\mathbf{curl}\ \mathbf{v}_h = \mathbf{curl}\ \mathbf{\Phi} = \mathbf{curl}\ \mathbf{\Pi}_h \mathbf{\Phi}$. That is, $\mathbf{v}_h - \mathbf{\Pi}_h \mathbf{\Phi}$ is in the kernel of $\mathbf{curl}$. Therefore, there exists a $p_h \in S_h$ such that $\nabla p_h = \mathbf{v}_h - \mathbf{\Pi}_h \mathbf{\Phi}$. It satisfies

$$\begin{aligned}\|\nabla p_h\|_{0,\omega} &\le \|\mathbf{v}_h\|_{0,\omega} + \|\mathbf{\Pi}_h \mathbf{\Phi}\|_{0,\omega} \\ &\le \|\mathbf{v}_h\|_{0,\omega} + \|(I - \mathbf{\Pi}_h)\mathbf{\Phi}\|_{0,\omega} + \|\mathbf{\Phi}\|_{0,\omega} \\ &\lesssim \|\mathbf{v}_h\|_{0,\omega} + \|\mathbf{curl}\ \mathbf{v}_h\|_{0,\omega}.\end{aligned}$$

In the last inequality, we used Lemma 1, the inverse inequality, and Theorem 3. We then define the other two terms in the decomposition in the theorem as

$$\widetilde{\mathbf{v}}_h := \mathbf{\Pi}_h\,(\mathbf{\Phi} - Q_h^\omega \mathbf{\Phi}) \in \mathbf{V}_h, \quad \mathbf{\Phi}_h := Q_h^\omega \mathbf{\Phi} \in \mathbf{S}_h,$$

where Q_h^ω is the weighted L^2 projection introduced in [5]. Note that in our setting of the interface problem, Q_h^ω satisfies

$$\|(I - Q_h^\omega)v\|_{0,\omega} \lesssim |v|_{1,\omega} \text{ and } |Q_h^\omega v|_{1,\omega} \lesssim |v|_{1,\omega}, \quad \forall v \in H_0^1(\Omega).$$

Hence, we have $|\!|\!|\mathbf{\Phi}_h|\!|\!|_\tau \lesssim \|\mathbf{\Phi}\|_{1,\omega} \lesssim \|\mathbf{curl}\ \mathbf{v}_h\|_{0,\omega} \le \|\mathbf{v}_h\|_A$. Moreover, we have

$$\begin{aligned}\|h^{-1}\widetilde{\mathbf{v}}_h\|_{0,\omega} &\le \|h^{-1}(I - \mathbf{\Pi}_h)(I - Q_h^\omega)\mathbf{\Phi}\|_{0,\omega} + \|h^{-1}(I - Q_h^\omega)\mathbf{\Phi}\|_{0,\omega} \\ &\lesssim \|\mathbf{\Phi}\|_{1,\omega} \lesssim \|\mathbf{curl}\ \mathbf{v}_h\|_{0,\omega} \lesssim \|\mathbf{v}_h\|_A.\end{aligned}$$

This completes the proof.

The resulting HX preconditioner for Eq. (2) reads

$$\mathbf{B} := \mathbf{D}_A^{-1} + \mathbf{P}_h(\mathbf{L}(\omega) + \tau\mathbf{M}(\omega))^{-1}\mathbf{P}_h{}^T + \tau^{-1}\mathbf{G}L(\omega)^{-1}\mathbf{G}^T, \tag{7}$$

where $\mathbf{D}_A$ is the diagonal of $\mathbf{A}$; $\mathbf{P}_h$ is the matrix representation of $\mathbf{\Pi}_h$; $\mathbf{L}(\omega) + \tau\mathbf{M}(\omega)$ is the matrix associated with the bilinear form $(\omega\nabla\mathbf{\Phi}, \nabla\mathbf{\Psi}) + \tau(\omega\mathbf{\Phi}, \mathbf{\Psi})$ on $\mathbf{S}_h$; $L(\omega)$ is the matrix associated with $(\omega\nabla\phi, \nabla\psi)$ on S_h; and $\mathbf{G}$ is the discrete gradient matrix. Standard multilevel preconditioners are robust for solving the H^1-interface problems $\mathbf{L}(\omega) + \tau\mathbf{M}(\omega)$ and $L(\omega)$ (see [21] for the theoretical justifications). In practical implementation, we can also replace $(\mathbf{L}(\omega) + \tau\mathbf{M}(\omega))^{-1}$ by an AMG solver for $\mathbf{P}_h{}^T\mathbf{A}\mathbf{P}_h$, and replace $L(\omega)^{-1}$ by an AMG solver for $\mathbf{G}^T\mathbf{A}\mathbf{G}$.

Based on Theorem 4 and the framework developed in [9], the HX preconditioner (7) is robust with respect to the coefficients and meshsize. More precisely, we have the following theorem:

Theorem 5. *The condition number $\kappa(\mathbf{BA}) \le C$, where the constant C is independent of the coefficients and the mesh size.*

4 Conclusions

In this paper, we have developed HX-preconditioners for the $\mathbf{H}(\mathbf{curl})$ interface problems. We have shown the robustness of the preconditioner by showing that the condition number of the preconditioned system is uniformly bounded with respect to the coefficients and the meshsize.

Acknowledgement. The first author was supported in part by NSF DMS-0609727, NSFC-10528102 and Alexander von Humboldt Research Award for Senior US Scientists. The second author would like to thank his postdoctoral advisor Professor Michael Holst for his encouragement and support through NSF Awards 0715146 and 0411723.

References

1. A. Alonso and A. Valli. Some remarks on the characterization of the space of tangential traces of $H(\mathrm{rot};\Omega)$ and the construction of an extension operator. *Manuscripta Math.*, 89(2):159–178, 1996. ISSN 0025-2611.
2. D.N. Arnold, R.S. Falk, and R. Winther. Multigrid in $H(\mathrm{div})$ and $H(\mathrm{curl})$. *Numer. Math.*, 85:197–218, 2000.
3. M.Sh. Birman and M.Z. Solomyak. L^2-theory of the Maxwell operator in arbitrary domains. *Russian Math. Surv.*, 42(6):75–96, 1987.
4. P.B. Bochev, J.J. Hu, C.M. Siefert, and R.S. Tuminaro. An algebraic multigrid approach based on a compatible gauge reformulation of Maxwell's equations. Technical Report SAND2007-1633J, Sandia National Laboratory, 2007.
5. J.H. Bramble and J. Xu. Some estimates for a weighted L^2 projection. *Math. Comput.*, 56:463–476, 1991.
6. Z. Chen, L. Wang, and W. Zheng. An adaptive multilevel method for time-harmonic Maxwell equations with singularities. *SIAM J. Sci. Comput.*, 29(118), 2007.
7. A.S.B.B. Dhia, C. Hazard, and S. Lohrengel. A singular field method for the solution of Maxwell's equations in polyhedral domains. *SIAM J. Appl. Math.*, 59(6):2028–2044 (electronic), 1999. ISSN 0036-1399.
8. R. Hiptmair. Multigrid method for Maxwell's equations. *SIAM J. Numer. Anal.*, 36(1): 204–225, 1998. URL http://link.aip.org/link/?SNA/36/204/1.
9. R. Hiptmair and J. Xu. Nodal auxiliary space preconditioning in $H(\mathrm{curl})$ and $H(\mathrm{div})$ spaces. *SIAM J. Numer. Anal.*, 45:2483–2509, 2007.
10. R. Hiptmair. Finite elements in computational electromagnetism. *Acta Numer.*, 11:237–339, 2002.
11. Q. Hu and J. Zou. A nonoverlapping domain decomposition method for Maxwell's equations in three dimensions. *SIAM J. Numer. Anal.*, 41(5):1682–1708, 2003. URL http://link.aip.org/link/?SNA/41/1682/1.
12. Q. Hu and J. Zou. A weighted Helmholtz decomposition and application to domain decomposition for saddle-point Maxwell systems. Technical Report 2007-15 (355), CUHK, 2007.
13. T.V. Kolev and P.S. Vassilevski. Some experience with a H^1-based auxiliary space AMG for H (curl) problems. Technical Report UCRL-TR-221841, Lawrence Livermore Nat. Lab., 2006.

14. T.V. Kolev and P.S. Vassilevski. Parallel auxiliary space AMG for H (curl) problems. *J. Comput. Math.*, 27(5):604–623, 2009.
15. T.V. Kolev, J.E. Pasciak, and P.S. Vassilevski. $H(\mathrm{curl})$ auxiliary mesh preconditioning. *Numer. Linear Algebra Appl.*, 15(5):455–471, 2008. URL http://dx.doi.org/10.1002/nla.534.
16. P. Monk. *Finite Element Methods for Maxwell's Equations*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, NY, 2003. ISBN 0-19-850888-3. URL http://dx.doi.org/10.1093/acprof:oso/9780198508885.001.0001.
17. J.E. Pasciak and J. Zhao. Overlapping Schwarz methods in $H(\mathrm{curl})$ on polyhedral domains. *J. Numer. Math.*, 10(3):221–234, 2002. ISSN 1570-2820.
18. S. Reitzinger and J. Schöberl. An algebraic multigrid method for finite element discretizations with edge elements. *Numer. Linear Algebra Appl.*, 9(3):223–238, 2002. URL http://dx.doi.org/10.1002/nla.271.
19. A. Toselli. Overlapping Schwarz methods for Maxwell's equations in three dimensions. *Numer. Math.*, V86(4):733–752, 2000. URL http://www.springerlink.com/content/4h0txq7wpw5wrubq.
20. A. Toselli and O. Widlund. *Domain Decomposition Methods—Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2005. ISBN 3-540-20696-5.
21. J. Xu and Y. Zhu. Uniform convergent multigrid methods for elliptic problems with strongly discontinuous coefficients. *Math. Models Methods Appl. Sci.*, 18(1):77 –105, 2008.
22. J. Xu. The auxiliary space method and optimal multigrid preconditioning techniques for unstructured meshes. *Computing*, 56:215–235, 1996.
23. L.T. Zikatanov. Two-sided bounds on the convergence rate of two-level methods. *Numer. Linear Algebra Appl.*, 15(5):439–454, 2008.

Mixed Multiscale Finite Element Analysis for Wave Equations Using Global Information

Lijian Jiang[1] and Yalchin Efendiev[2]

[1] Institute for Mathematics and its Applications,University of Minnesota, Minneapolis, MN, USA, lijiang@ima.umn.edu, corresponding author
[2] Department of Mathematics, Texas A&M University, College Station, TX, USA, efendiev@math.tamu.edu

Summary. A mixed multiscale finite element method (MsFEM) for wave equations is presented. Global information is used in the mixed MsFEM to construct multiscale basis functions. The solution of the wave equation smoothly depends on the global information. We investigate the relation between the smoothness of the global information and convergence rate of the mixed MsFEM.

1 Introduction

Over the past few decades, there has been growing interest in wave propagation in heterogeneous media. Many important problems such as earthquake motions, oceanography, medical and material sciences, and the morphology of oil and gas deposits can be understood through some use of mathematical and numerical modelings of wave propagation in heterogeneous media. In addition to heterogeneity, wave propagation is also a challenging multiscale problem. Among typical length scales present in wave propagation are wave length, propagation distance, and correlation length. In some problems such as in reflection seismology, the wave can propagate over a distance significantly larger than the wave length.

Consideration for accuracy suggests that the heterogeneity of media has to be sufficiently resolved when numerically simulating wave propagation, which can easily results in very expensive computations. While much more efficient and inexpensive in practice, standard upscaling techniques and multiscale methods employing some local information often fail to accurately transfer the fine scale information in media to the coarse formulation. Previous investigations (see e.g., [8]) indicate that appropriately taking into account some type of global information can potentially improve the accuracy of the multiscale methods. The importance of global information has been illustrated in porous media flow within the context of upscaling procedures [2] and also in multiscale finite (volume) elements [1]. The information is determined by some global fields that the solution of equations smoothly depends on. In the context of weak formulation, this global field is imbedded in the (multiscale) basis functions

which in turn is used to represent the solution. Our objective is to develop a mixed MsFEM using global information that can capture the solution of wave equations in multiscale heterogeneous media and to make a priori error estimates for the mixed MsFEM.

The rest of the paper is organized as follows. In Sect. 2, we present some preliminaries. In Sect. 3, we present a mixed MsFEM for a model wave equation using global information and derive the error estimates. Some conclusions are drawn finally.

2 Preliminaries

In this section, we describe a model wave equation and some notations of function spaces.

Define D_{tt} to be the second order partial derivative operator with respect to t. Let $a(x)$ and $u(t,x)$ represent the density of the material and the unknown pressure, respectively. We define a time-space domain $\Omega_T := (0,T] \times \Omega$. Then a model wave equation reads as following:

$$\begin{aligned} D_{tt}u(t,x) - \nabla \cdot a(x)\nabla u(t,x) &= f(t,x) \quad \text{in} \quad \Omega_T \\ u(t,x) &= 0 \quad \text{on} \quad [0,T] \times \partial\Omega \\ u(x,0) &= g_0(x) \quad \text{in } \Omega \\ D_t u(x,0) &= g_1(x) \quad \text{in } \Omega. \end{aligned} \tag{1}$$

Here we assume that $a(x)$ is uniformly positive, symmetric and bounded in Ω. We assume that $f(t,x)$, $g_0(x)$ and $g_1(x)$ are smooth and do not have multiscale structures. This equation arises from geophysics and seismology. It is frequently observed that the spatial scales inherent in $a(x)$ cannot be clearly separated.

We introduce some notation which are used in the following sections. The usual Lebesgue and Sobolev spaces are denoted by $L^p(D)$, $W^{k,p}(D)$. In particular, $H^k(D) := W^{k,2}(D)$. Define $H(div,D) := \{f | f \in [L^2(D)]^d \quad \text{and} \quad \nabla \cdot f \in L^2(D)\}$. The vector-valued Sobolev space is equipped with the norm

$$\|u\|_{W^{m,p}(0,T;X)} := \left(\int_0^T \sum_{0 \le k \le m} \|D_t^k u\|_X^p dt \right)^{\frac{1}{p}},$$

when X is a normed space. If $p = 2$, we use $H^m(0,T;X)$ instead. When no ambiguity occurs, we use $W^{m,p}(X)$ to denote $W^{m,p}(0,T;X)$.

Without loss of generality, our discussion is concentrated on problems in $\Omega \subset \mathbb{R}^2$. We denote by K a generic coarse element with $h = \text{diam}(K)$, and τ_h a quasi-uniform family of coarse elements K. We shall not write the variables x and t for simplicity of presentation.

3 Mixed MsFEM Analysis

In this section, we first present a mixed MsFEM for the wave equation (1) using multiple global information, and then derive a priori error estimates in pressure and velocity.

3.1 Mixed MsFEM Formulation

Let velocity $\sigma = a\nabla u$. Then the mixed formulation for (1) is to find $\{u, \sigma\} : [0,T] \longrightarrow L^2(\Omega) \times H(div, \Omega)$ such that

$$\begin{aligned} (D_{tt}u, w) - (\nabla \cdot \sigma, w) &= (f, w) \quad \forall w \in L^2(\Omega) \\ (a^{-1}\sigma, \chi) + (u, \nabla \cdot \chi) &= 0 \quad \forall \chi \in H(div, \Omega) \\ (u(0), w) &= (g_0, w) \quad \forall w \in L^2(\Omega) \\ ((D_t u)(0), w) &= (g_1, w) \quad \forall w \in L^2(\Omega) \\ (a^{-1}\sigma(0), \chi) &= (\nabla g_0, \chi) \quad \forall \chi \in H(div, \Omega). \end{aligned} \tag{2}$$

We use global fields σ_i $(i = 1, \cdots, N)$ to build velocity basis function. We formulate an assumption for the global fields as following.

Assumption 1 *There exist functions $\sigma_1, \cdots, \sigma_N$ and $A_1(t,x), \cdots, A_N(t,x)$ such that*

$$\sigma = \sum_{i=1}^{N} A_i(t,x)\sigma_i,$$

where $A_i(t,x)$'s are smooth functions (we specify their smoothness later) and $\sigma_i = a(x)\nabla p_i$ $(i = 1, ..., N)$ solves an elliptic equation $\nabla \cdot a(x)\nabla p_i = 0$ with appropriate boundary conditions.

Remark 1. As an example in 2D, let u_1 and u_2 be the solution of the following equations

$$\begin{aligned} \nabla a \cdot \nabla u_i &= 0 \quad \text{in} \quad \Omega \\ u_i &= x_i \quad \text{on} \quad \partial\Omega, \quad i = 1, 2. \end{aligned} \tag{3}$$

set $u = u(t, u_1, u_2)$, then

$$\sigma = a\nabla u = \sum_{i=1}^{2} \frac{\partial u}{\partial u_i} a\nabla u_i := \sum_{i=1}^{2} A_i(t,x)\sigma_i,$$

where $A_i(t,x) = \frac{\partial u}{\partial u_i}$. Here $\sigma_i = a\nabla u_i$ are the global fields. Provided that $f \in L^\infty(L^p(\Omega)) \cap H^1(L^p(\Omega))$, $g_1 \in W^{1,p}(\Omega)$ and $D_{tt}u(0) \in L^p(\Omega)$, then the proof Theorem 1.1 in [8] implies that $A_i(t,x) = \frac{\partial u}{\partial u_i} \in L^\infty(W^{1,p}(\Omega))$. Consequently $A_i(t,x) \in L^2(C^{1-\frac{2}{p}}(\Omega))$ if $p > 2$ by using the Sobolev embedding theorem.

To numerically approximate the mixed problem (2), we construct the basis function for the velocity σ,

$$\begin{aligned} \nabla \cdot (a(x)\nabla\phi_{ij}^K) &= \frac{1}{|K|} \quad \text{in} \quad K \\ a(x)\nabla\phi_{ij}^K \cdot n_{e_l}^K &= \delta_{jl}\frac{\sigma_i \cdot n_{e_l}^K}{\int_{e_l}\sigma_i \cdot n_{e_l} ds} \quad \text{on} \quad \partial K, \end{aligned} \tag{4}$$

where $i = 1, ..., N$ and j is the index of the edges of the coarse block K (a triangle or rectangle), and

$$\delta_{jj} = 1, \;\; \delta_{jl} = 0 \;\text{ if }\; j \neq l.$$

Here e_l denotes an edge of the coarse block. We shall omit the subscript e_l in n if the integral is taken along the edge. Note that for each edge, we have N basis functions and we assume that σ_1,..., σ_N are linearly independent in order to guarantee that the basis functions are linearly independent. To avoid the possibility that $\int_{e_l}\sigma_i \cdot n ds$ is zero or unbounded, we make the following assumption for the convergence analysis. If $\int_{e_l}\sigma_i \cdot n ds = 0$ on some e_l, we can use the local mixed MsFEM basis function proposed in [3], i.e., replace $\frac{\sigma_i \cdot n_{e_l}^K}{\int_{e_l}\sigma_i \cdot n_{e_l} ds}$ with $\frac{1}{|e_l|}$ in (4).

Assumption 2

$$\int_{e_l}|\sigma_i \cdot n| ds \leq Ch^{\beta_1} \;\text{ and }\; \|\frac{\sigma_i \cdot n}{\int_{e_l}\sigma_i \cdot n ds}\|_{L^r(e_l)} \leq Ch^{-\beta_2+\frac{1}{r}-1}$$

uniformly for all edges e_l, *where* $\beta_1 \leq 1$, $\beta_2 \geq 0$ *and* $r \geq 1$.

We would like to note that *Assumption 2* is used to define the boundary data for the velocity basis equations well and to bound the velocity basis function ψ_{ij}^K. In fact, *Assumption 2* implies that $\|\psi_{ij}^K\|_{0,K} \leq Ch^{-\beta_2}$ (see [1]). If σ_i are bounded in $L^\infty(e_l)$ for all e_l and $|\int_{e_l}\sigma_i \cdot n ds|$ remains positive uniformly for all e_l, then $\beta_1 = 1$ and $\beta_2 = 0$. The index r is only related to the L^r norm that appeared in Assumption 2 and has nothing to do with the convergence rate. We would like to note that local mixed MsFE basis function introduced in [3] is a special case defined in (4). To do this, one just needs to replace σ_1 in (4) by a constant vector.

We define $\psi_{ij}^K = a(x)\nabla\phi_{ij}^K$ and

$$\Sigma_h = \bigoplus_K \{\psi_{ij}^K\} \subset H(div, \Omega).$$

Let $Q_h = \bigoplus_K \mathrm{P}_0(K) \subset L^2(\Omega)$, i.e., piecewise constants, be the basis functions approximating u. For $t > 0$, we define

$$\Pi_h|_K\sigma(t) = \sum_{i,j}(\int_{e_j} A_i(t,x)\sigma_i \cdot n dx)\psi_{ij}^K$$

The numerical mixed formulation is to find $\{u_h, \sigma_h\} : [0, T] \longrightarrow Q_h \times \Sigma_h$ such that

$$\begin{aligned}
(D_{tt}u_h, w) - (\nabla\cdot\sigma_h, w) &= (f, w) \quad \forall w \in Q_h\\
(a^{-1}\sigma_h, \chi) + (u_h, \nabla\cdot\chi) &= 0 \quad \forall \chi \in \Sigma_h\\
(u_h(0), w) &= (g_0, w) \quad \forall w \in Q_h\\
((D_t u_h)(0), w) &= (g_1, w) \quad \forall w \in Q_h\\
(\sigma_h(0), \chi) &= (\sigma(0), \chi) \quad \forall \chi \in \Sigma_h.
\end{aligned} \tag{5}$$

3.2 A Priori Error Estimates for Continuous Time

Before we proceed with the convergence analysis of the mixed MsFEM for the wave equation, we recall some properties for the basis defined in (4). By Lemma 3.1 in [1], it follows that

$$\sigma_i|_K = \sum_j \beta_{ij}^K \psi_{ij}^K, \tag{6}$$

where $\beta_{ij}^K = \int_{e_j} \delta_{ij}\sigma_i \cdot n dx$. For the interpolator Π_h, Lemma 3.2 in [1] claims

$$(\nabla\cdot(\sigma - \Pi_h\sigma), w) = 0 \quad w \in Q_h. \tag{7}$$

Let P_h be $L^2(\Omega)$ orthogonal projection onto Q_h. We define

$$\|\sigma\|^2_{L^2_a(\Omega)} = \int_\Omega \sigma^t \cdot a^{-1}(x)\sigma dx, \quad \|\sigma\|^2_{L^2(0,T;L^2_a(\Omega))} = \int_{\Omega_T} \sigma^t \cdot a^{-1}(x)\sigma dx ds.$$

By using (7) and standard estimate techniques (e.g., Schwarz inequality, Grownall's inequality, Jensen's inequality and triangle inequality), we can obtain the following lemma.

Lemma 1. *[6] Let $\{u, \sigma\}$ and $\{u_h, \sigma_h\}$ be respectively solution of (2) and (5). Then*

$$\begin{aligned}
&\|u - u_h\|^2_{L^\infty(L^2(\Omega))} + \sup_t \|\int_0^t (\sigma(s) - \sigma_h(s))ds\|^2_{L^2_a(\Omega)}\\
&\le C(\|P_h u - u\|^2_{L^\infty(L^2(\Omega))} + \|\Pi_h\sigma - \sigma\|^2_{L^2(L^2_a(\Omega))}).
\end{aligned} \tag{8}$$

By Lemma 1, we get an priori error estimate for the scheme defined in (5).

Theorem 1. *Suppose that $f \in L^2(L^2(\Omega))$, $g_0 \in H^1(\Omega)$ and $g_1 \in L^2(\Omega)$. Let $\{u, \sigma\}$ and $\{u_h, \sigma_h\}$ be solution of (2) and (5), respectively. If Assumption 1 and Assumption 2 hold and $A_i(t, x) \in L^2(C^\alpha(\Omega))$ for $i = 1, \cdots, N$, then for $\alpha + \beta_1 - \beta_2 - 1 > 0$,*

$$\|u - u_h\|_{L^\infty(L^2(\Omega))} + \sup_t \|\int_0^t (\sigma(s) - \sigma_h(s))ds\|_{L^2_a(\Omega)} \le Ch^{\min(1, \alpha+\beta_1-\beta_2-1)}.$$

Proof. If the source term $f \in L^2(L^2(\Omega))$, the initial conditions $g_0 \in H^1(\Omega)$ and $g_1 \in L^2(\Omega)$, then $u \in L^\infty(H^1(\Omega))$ (see [5]). Thanks to the fact that P_h is the $L^2(\Omega)$ projection onto Q_h,

$$\|u - P_h u\|_{L^\infty(L^2(\Omega))} \leq Ch|u|_{L^\infty(H^1(\Omega))}, \tag{9}$$

which estimates the first term of right hand side in (8). Next we estimate the term $\|\sigma - \Pi_h\sigma\|^2_{L^2(L^2_a(\Omega))}$. Define

$$A^K_{ij}(t) = \int_{e_j} A_i(t,s)\sigma_i \cdot n ds$$

on each element K. With $\bar{A}^j_i$ the average $A_i(x)$ along e_j, then

$$\begin{aligned} |A^K_{ij} - \bar{A}^j_i \beta^K_{ij}| &= |\int_{e_j} A_i\sigma_i \cdot n ds - \bar{A}^j_i \int_{e_j} \sigma_i \cdot n ds| \\ &\leq Ch^{\alpha+\beta_1}\|A_i(t)\|_{C^\alpha(\Omega)}, \end{aligned} \tag{10}$$

where we have used the *Assumption 2*.

Invoking *Assumption 1*, (6) and $\|\psi^K_{ij}\|_{0,K} \leq Ch^{-\beta_2}$, see [1], we have in each element K

$$\begin{aligned} &\|\sigma - \Pi_h\sigma\|^2_{L^2(0,T;L^2_a(K))} = \\ &\int_0^T \int_K \sum_{i,j}(A_i(t,x)\beta^K_{ij} - A^K_{ij}(t))\psi^K_{ij} \cdot a^{-1} \sum_{i,j}(A_i(t,x)\beta^K_{ij} - A^K_{ij}(t))\psi^K_{ij} dxdt \\ &\leq C\int_0^T \int_K (\sum_{i,j}(A_i(t,x)\beta^K_{ij} - A^K_{ij}(t))\psi^K_{ij})^2 dxdt \\ &= C\|\sum_{i,j}(A_i(t,x)\beta^K_{ij} - A^K_{ij}(t))\psi^K_{ij}\|^2_{L^2(0,T;L^2(K))} \\ &\leq C\|\sum_{i,j}(A_i(t,x) - \bar{A}^j_i(t))\beta^K_{ij}\psi^K_{ij}\|^2_{L^2(0,T;L^2(K))} \\ &+ C\|\sum_{i,j}(\bar{A}^j_i(t)\beta^K_{ij} - A^K_{ij}(t))\psi^K_{ij}\|^2_{L^2(0,T;L^2(K))} \\ &\leq Ch^{2(\alpha+\beta_1)}(\sum_i \|A_i\|^2_{L^2(0,T;C^\alpha(K))}) \sum_{ij} \|\psi^K_{ij}\|^2_{0,K} \\ &\leq Ch^{2(\alpha+\beta_1-\beta_2)}(\sum_i \|A_i\|^2_{L^2(0,T;C^\alpha(K))}), \end{aligned} \tag{11}$$

where we have used the facts that $A_i \in L^2(0,T;C^\alpha(\Omega))$ and (10). After making summation over all K in (11), we have

$$\|\sigma - \Pi_h\sigma\|_{L^2(0,T;L^2_a(\Omega))} \leq Ch^{\alpha+\beta_1-\beta_2-1}. \tag{12}$$

Taking into account (9) , (12) and (8), the proof is complete.

If the functions $A_i(t,x)$ $(i = 1, \cdots, N)$ in *Assumption 1* have better regularity with respect to time t, we can obtain an convergence rate as follows:

Theorem 2. *Let $\{u, \sigma\}$ and $\{u_h, \sigma_h\}$ be the solution of (2), respectively and (5). If Assumption 1, 2 hold and $A_i(t,x) \in L^\infty(C^\alpha(\Omega)) \cap H^1(C^\alpha(\Omega))$ for $i = 1, \cdots, N$, then for $\alpha + \beta_1 - \beta_2 - 1 > 0$,*

$$\|u - u_h\|_{L^\infty(L^2(\Omega))} + \|\sigma - \sigma_h\|_{L^\infty(L^2_a(\Omega))} \leq Ch^{\min(1,\alpha+\beta_1-\beta_2-1)}.$$

The proof can be found in [6].

Remark 2. If global fields $u_i (i = 1, 2)$ are defined in Remark 1, then $\sigma = \sum_{i=1}^{2} A_i(t,x)\sigma_i$, where $A_i(t,x) = \frac{\partial u}{\partial u_i}$ and $\sigma_i = a\nabla u_i$. Provided that $f \in W^{1,\infty}(L^p(\Omega)) \cap W^{2,p}(L^p(\Omega))$, $D_{tt}u(0) \in W^{1,p}(\Omega)$ and $D_{ttt}u(0) \in L^p(\Omega)$, then the proof Lemma 2.6 in [8] implies that $A_i(t,x) = \frac{\partial u}{\partial u_i} \in W^{1,\infty}(W^{1,p}(\Omega))$. Consequently $A_i(t,x) \in H^1(C^{1-\frac{2}{p}}(\Omega))$ if $p > 2$ by using Sobolev embedding theorem.

3.3 A Priori Error Estimate for Discrete Time

We introduce the following notation for time-discetization,

$$D_t u^{\frac{1}{2}} = \frac{u^1 - u^0}{\Delta t}, \quad D_{tt}u^n = \frac{u^{n+1} - 2u^n + u^{n-1}}{\Delta t^2}.$$

Because we assume that the media has only spatial multiscales, we use the mixed MsFEM for the space discretization and use conventional finite difference schemes to discretize the temporal variables. As an explicit-in-time scheme, the fully mixed formulation is to find $\{u_h^{n+1}, \sigma_h^{n+1}\} \in Q_h \times \Sigma_h$ such that

$$\begin{aligned}
(D_{tt}u_h^n, w) - (\nabla \cdot \sigma_h^n, w) &= (f^n, w) \quad \forall w \in Q_h \\
(a^{-1}\sigma_h^{n+1}, \chi) + (u_h^{n+1}, \nabla \cdot \chi) &= 0 \quad \forall \chi \in \Sigma_h \\
(u_h^0, w) &= (g_0, w) \quad \forall w \in Q_h \\
(\frac{2}{\Delta t} D_t u_h^{\frac{1}{2}}, w) - (\nabla \cdot \sigma_h^0, w) &= (f^0 + \frac{2}{\Delta t} g_1, w) \quad \forall w \in Q_h \\
(\sigma_h^0, \chi) &= (\sigma(0), \chi) \quad \forall \chi \in \Sigma_h.
\end{aligned} \tag{13}$$

It is known that the scheme in (13) is conditional stable (refer to [4]) and that the time consistence error is $O(\Delta t^2)$ if $u(t,x)$ is sufficiently smooth with respect to t. Consequently, we can use Theorem 2 and follow the proof of Theorem 5.2 in [4] to obtain the following estimate.

Theorem 3. *Let $\{u, \sigma\}$ and $\{u_h, \sigma_h\}$ be solution of (2) and (13), respectively. If $u(t,x)$ is sufficiently smooth with respect to t and the assumptions in Theorem 2 are satisfied, then*

$$\sup_{t_n} \|u - u_h^n\|_{L^2(\Omega)} + \sup_{t_n} \|\sigma - \sigma_h^n\|_{L^2_a(\Omega)} \leq C(h^{\min(1,\alpha+\beta_1-\beta_2-1)} + \Delta t^2).$$

We would like to note that an implicit-in-time scheme for the wave equation is presented in [4].

4 Conclusions

In the paper, we present a mixed MsFEM for a wave equation using global information. The global information is described by global fields (velocity fields). For construction of velocity basis functions, the global fields are employed. A prior error estimates are derived for the wave equation by the mixed MsFEM. The numerical results in some recent works (e.g., [6, 7, 8]) demonstrate that using global fields can capture non-local effects in simulations and significant improve accuracy and efficiency when the media are heterogenous and their scales are non-separable.

References

1. J.E. Aarnes, Y. Efendiev, and L. Jiang. Mixed multiscale finite element methods using limited global information. *Multiscale Model. Simul.*, 7(2):655–676, 2008. ISSN 1540-3459.
2. Y. Chen and L.J. Durlofsky. Adaptive local-global upscaling for general flow scenarios in heterogeneous formations. *Transp. Porous Media*, 62(2):157–185, 2006. ISSN 0169-3913.
3. Z. Chen and T.Y. Hou. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comput.*, 72(242):541–576 (electronic), 2003.
4. L.C. Cowsar, T.F. Dupont, and M.F. Wheeler. A priori estimates for mixed finite element approximations of second-order hyperbolic equations with absorbing boundary conditions. *SIAM J. Numer. Anal.*, 33(2):492–504, 1996. ISSN 0036-1429.
5. L.C. Evans. *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 1998. ISBN 0-8218-0772-2.
6. L. Jiang. *Multiscale Numerical Methods for Partial Differential Equations Using Limited Global Information and Their Applications*. PhD thesis, Texas A&M University, College Station, TX, 2008.
7. L. Jiang, Y. Efendiev, and V. Ginting. Analysis of global multiscale finite element methods for wave equations with continuum spatial scales. *Appl. Numer. Math.*, 60(8):862–876, 2010.
8. H. Owhadi and L. Zhang. Numerical homogenization of the acoustic wave equations with a continuum of scales. *Comput. Methods Appl. Mech. Eng.*, 198(3–4):397–406, 2008.

A Domain Decomposition Preconditioner for Multiscale High-Contrast Problems

Yalchin Efendiev and Juan Galvis

Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA, efendiev@math.tamu.edu

1 Summary

We present a new class of coarse spaces for two-level additive Schwarz preconditioners that yield condition number bound independent of the contrast in the media properties. These coarse spaces are an extension of the spaces discussed in [3]. Second order elliptic equations are considered. We present theoretical and numerical results. Detailed description of the results and numerical studies will be presented elsewhere.

2 Introduction

Many problems in applied sciences occur in media that contains multiple scales and has high contrast in the properties. For example, it is very common to have several orders of magnitude of variations in the permeability field in natural porous formations. Domain decomposition preconditioners are often used to solve the fine-scale system that arises from the discretization of partial differential equations. The number of iterations required by domain decomposition preconditioners is typically affected by the contrast in the media properties that are within each coarse-grid block (e.g., [3, 4]; see also [2] for the approximation on a coarse grid). It is known that if high and low conductivity regions can be encompassed within coarse-grid blocks such that the variation of the conductivity within each coarse region is bounded, domain decomposition preconditioners result to a system with the condition number independent of the contrast (e.g., [5]). Because of complex geometry of fine-scale features, it is often impossible to separate low and high conductivity regions into different coarse-grid blocks. Thus, the contrast will adversely affect the number of iterations required by domain decomposition preconditioners.

The design and analysis of preconditioners that converge independent of the contrast is important for many applications, such as porous media flows where flow problems are solved multiple times. In [3], we introduce a coarse space based on local spectral problems (see also [1]). These spaces are motivated by weighted Poincaré

estimates that arise in the proofs of L^2 approximation property of the coarse interpolation, see [3, 6, 8]. In particular, the spectrum of local eigenvalue problem contains eigenvalues that are small and asymptotically vanish as the contrast increases, and thus, there is a gap in the spectrum. The eigenvectors corresponding to these small (asymptotically vanishing) eigenvalues represent the high-conducting features. The number of these eigenvectors is the same as the number of disconnected high-conductivity inclusions. The coarse space is constructed such that the basis functions span the eigenfunctions corresponding to these small (asymptotically vanishing) eigenvalues as well as some nodal multiscale basis functions. In [3], we prove that if the coarse space includes the basis functions associated to these eigenfunctions, then the condition number of the two level additive method is bounded independent of the contrast of the media.

In many applications where the flow equations are solved multiple times, it is important to choose a coarse space with a minimal dimension. The coarse spaces constructed in [3] represent both high-conductivity channels (high-conductivity inclusions that connect the boundaries of a coarse-grid block) and high-conductivity isolated inclusions. Consequently, these coarse spaces can have a large dimension. In [3], we note that one only needs to represent channels within coarse blocks and present a procedure for removing high-contrast isolated inclusions. In this paper, we present a more general approach that removes the inclusions. In fact, one can consider the proposed construction as an approach that *complements* the coarse spaces constructed using partition of unity functions. In particular, starting with an initial partition of unity functions, e.g., multiscale basis functions, one adds new basis functions by using eigenvectors of weighted eigenvalue problem. In this eigenvalue problem, the weight is computed using the gradient of the initial partition of unity functions, see (8) below. The eigenfunctions corresponding to small (asymptotically vanishing) eigenvalues are chosen and new basis functions that span these eigenfunctions are added to the coarse space. With a correct choice of partition of unity functions, one can remove the inclusions and obtain the coarse space with a small dimension. We present a theoretical result that states that the condition number of the preconditioned system is independent of contrast. Numerical results are presented to demonstrate our theoretical findings.

3 Problem Setting and Domain Decomposition Framework

Let $D \subset \mathbb{R}^2$ (or $\mathbb{R}^3$) be a polygonal domain which is the union of a disjoint polygonal subregions $\{D_i\}_{i=1}^N$. We consider the following problem. Find $u^* \in H_0^1(D)$ such that

$$a(u^*, v) = f(v) \quad \text{for all } v \in H_0^1(D). \tag{1}$$

Here the bilinear form a and f are defined by $a(u, v) = \int_D \kappa(x)\nabla u(x)\nabla v(x)dx$, and $f(v) = \int_D f(x)v(x)dx$, for all $u, v \in H_0^1(D)$.

We assume that $\{D_i\}_{i=1}^N$ form a quasiuniform triangulation of D and denote $H = \max_i \text{diam}(D_i)$. Let $\mathcal{T}^h$ be a fine triangulation which refine $\{D_i\}_{i=1}^N$. We denote by $V^h(D)$ the usual finite element discretization of piecewise linear continuous

functions with respect to the fine triangulation $\mathcal{T}^h$. Denote also by $V_0^h(D)$ the subset of $V^h(D)$ with vanishing values on ∂D. Similar notations, $V^h(\Omega)$ and $V_0^h(\Omega)$, are used for subdomains $\Omega \subset D$.

The Galerkin formulation of (1) is to find $u^* \in V_0^h(D)$ with $a(u^*, v) = f(v)$ for all $v \in V_0^h(D)$, or in matrix form

$$Au^* = b, \tag{2}$$

where for all $u, v \in V^h(D)$ we have $u^T A v = \int_D \kappa \nabla u \nabla v$, and $v^T b = \int_D fv$.

It is sufficient to consider the case of piecewise constant coefficient κ. From now on we will assume that κ is piecewise constant coefficient in $\mathcal{T}^h$ with value $\kappa = \kappa_e$ on each fine triangulation element $e \in \mathcal{T}^h$.

We denote by $\{D_i'\}_{i=1}^N$ the overlapping decomposition obtained from the original nonoverlapping decomposition $\{D_i\}_{i=1}^N$ by enlarging each subdomain D_i to $D_i' = D_i \cup \{x \in D, \text{dist}(x, D_i) < \delta_i\}$, $i = 1, \ldots, N$, where dist is some distance function and let $\delta = \max_{1 \le i \le N} \delta_i$. Let $V_0^i(D_i')$ be the set of finite element functions with support in D_i'. We also denote by $R_i^T : V_0^i(D_i') \to V^h$ the extension by zero operator.

We will use a partition of unity $\{\xi_i\}_{i=1}^N$ subordinated to the covering $\{D_i'\}_{i=1}^N$ such that

$$\sum_{i=1}^N \xi_i = 1, \quad \xi_i \in V^h, \quad \text{and} \quad \text{Supp}(\xi_i) \subset D_i', i = 1, \ldots, N, \tag{3}$$

where $\text{Supp}(\xi_i)$ stands for the support of the function ξ_i. This will be the partition of unity used to truncate global functions to local ones in the proof of the stable decomposition.

Given a coarse triangulation $\mathcal{T}^H$ we introduce N_c coarse basis functions $\{\Phi_i\}_{i=1}^{N_c}$. We define the coarse space by $V_0 = \text{span}\{\Phi_i\}_{i=1}^{N_c}$, and the coarse matrix $A_0 = R_0 A R_0^T$ where $R_0^T = [\Phi_1, \ldots, \Phi_{N_c}]$. We use a two level additive preconditioner of the form

$$B^{-1} = R_0^T A_0^{-1} R_0 + \sum_{i=1}^N R_i^T A_i^{-1} R_i, \tag{4}$$

where the local matrices are defined by $v^T A_i w = a(v, w)$ for all $v, w \in V^i = V_0^h(D_i')$, $i = 1, \ldots, N$. See [5].

We denote by $\{y_i\}_{i=1}^{N_v}$ the vertices of the coarse mesh $\mathcal{T}^H$ and define

$$\omega_i = \bigcup \{K \in \mathcal{T}^H; \quad y_i \in \overline{K}\}, \quad \omega_K = \bigcup \{\omega_j; \quad y_j \in \overline{K}\}. \tag{5}$$

We will use a partition of unity $\{\chi_i\}_{i=1}^{N_v}$ subordinated to the covering $\{\omega_i\}_i^{N_v}$ such that

$$\sum_{i=1}^{N_v} \chi_i = 1, \quad \chi_i \in V^h, \quad \text{and} \quad \text{Supp}(\chi_i) \subset \omega_i, i = 1, \ldots, N_c. \tag{6}$$

4 Coarse-Space-Completing Eigenvalue Problem and Stability Estimates

In this section we define the new local spectral multiscale coarse space using eigenvectors of high contrast eigenvalue problems. Fist we introduce the notation for eigenvalue problems. For any $\Omega \subset D$ define the matrix A^{Ω} and the *modified mass matrix* of same dimension M^{Ω} by

$$v^T A^{\Omega} w = \int_{\Omega} \kappa \nabla v \nabla w \quad \text{and} \quad v^T M^{\Omega} w = \int_{\Omega} \widetilde{\kappa} v w \quad \text{for all } v, w \in \widetilde{V}^h(\Omega), \tag{7}$$

where $\widetilde{V}_h = V_h(\Omega)$ if $\overline{\Omega} \cap \partial D = \emptyset$ and $\widetilde{V}_h = \{v \in V_h(\Omega) : v = 0 \text{ on } \partial\Omega \cap \partial D\}$ otherwise. Here $\widetilde{\kappa}$ in (7) is a weight derived from the high contrast coefficient κ and contains the relevant information we need for the construction of the coarse basis functions. Several possible choices for $\widetilde{\kappa}$ can be considered. We refer to [3] for the case $\widetilde{\kappa} = \kappa$. Here we will consider only the case of the piecewise constant $\widetilde{\kappa}$ given by

$$\widetilde{\kappa} = \max \left\{ \kappa \sum_{i=1}^{N} |\nabla \xi_i|^2, \kappa \sum_{j=1}^{N_v} |\nabla \chi_j|^2 \right\}, \tag{8}$$

where $\{\xi\}_{j=1}^{N}$ and $\{\chi_i\}_{i=1}^{N_v}$ are the partition of unity introduced in (3) and (6), respectively. From now on, we assume that overlapping decomposition is constructed from the coarse mesh and $\xi_i = \chi_i$ for all $i = 1, \ldots, N = N_v$. We consider the finite dimensional symmetric eigenvalue problem

$$A^{\Omega} \phi = \widetilde{\lambda} M^{\Omega} \phi \tag{9}$$

and denote its eigenvalues and eigenvectors by $\{\widetilde{\lambda}_{\ell}^{\Omega}\}$ and $\{\psi_{\ell}^{\Omega}\}$, respectively. Note that the eigenvectors $\{\psi_{\ell}^{\Omega}\}$ form an orthonormal basis of $\widetilde{V}^h(\Omega)$ with respect to the M^{Ω} inner product. Assume that $\widetilde{\lambda}_1^{\Omega} \leq \widetilde{\lambda}_2^{\Omega} \leq \cdots \leq \widetilde{\lambda}_i^{\Omega} \leq \ldots$, and note that $\widetilde{\lambda}_1^{\Omega} = 0$. In particular, $\psi_{\ell}^{\omega_i}$ denotes the ℓ-th eigenvector of the matrix associated to the neighborhood of y_i, $i = 1, \ldots, N_v$.

In general, when $\widetilde{\kappa} = \kappa$ and for the Neumann boundary case, if there are n inclusions and channels, then one can observe n small (asymptotically vanishing) eigenvalues. The eigenvectors corresponding to these eigenvalues will be used to construct the coarse space V_0. In this case, the term $\widetilde{\kappa} = \kappa$ on the right hand side of the eigenvalue problem results in eigenvectors that are nearly constant inside each high conductivity inclusion/channel. When $\widetilde{\kappa}$ is chosen based on (8), then the number of asymptotically small eigenvalues is the same as the number of high-conductivity inclusions in $\widetilde{\kappa}$. In particular, if the partition of unity functions are piecewise linear polynomials then $\widetilde{\kappa}$ and κ have the same high-contrast structure. We are interested in partition of unity functions that can "eliminate" isolated high-conductivity inclusions and thus reduce the size of the coarse space. This can be achieved by minimizing high-conductivity components in $\widetilde{\kappa}$. In particular, by choosing multiscale finite element basis functions or energy minimizing basis functions, we can eliminate all

isolated high-conductivity inclusions, while preserving the channels. This can be observed in our numerical experiments. In Fig. 1 (below) and Fig. 2 (on page 195), we depict κ (middle picture) and $\widetilde{\kappa}$ (right picture) using multiscale basis functions on the coarse grid. The coarse grid is depicted in the left pictures. One can observe that isolated inclusions are removed in $\widetilde{\kappa}$, and consequently, the coarse space contains only long channels that connect boundaries of the coarse grid.

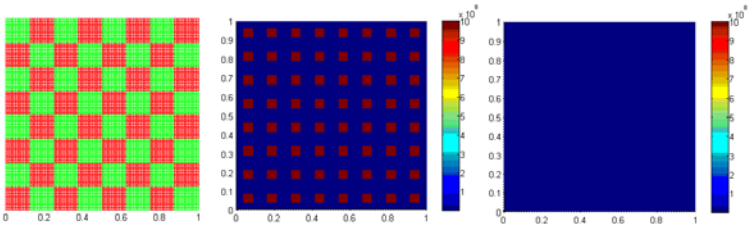


Fig. 1. *Left*: Coarse mesh. *Center*: Original coefficient. Here $\eta = 10^9$. *Right*: Coefficient $\widetilde{\kappa}$ computed as in (8) using (linear) multiscale basis functions.

We note that for the proposed methods, we only need to specify the eigenvectors based on the quantities $\{1/\widetilde{\lambda}_l^{\omega_i}\}$ in each ω_i, $i = 1, \ldots, N_v$. These eigenvectors are used to construct the coarse space.

We assume that the elements of $\mathcal{T}^h$ contained in Ω form a triangulation of Ω. Let $n_h(\Omega)$ denote the number of degrees of freedom in $\overline{\Omega}$. Given an integer L and $v \in V^h(\Omega)$ define

$$I_L^{\Omega} v = \sum_{\ell=1}^{L} \left(\int_{\Omega} \widetilde{\kappa} v \psi_{\ell}^{\Omega} \right) \psi_{\ell}^{\Omega}. \tag{10}$$

Let $\{\chi_i\}_{i=1}^{N_v}$ be a partition of unity (3). Define the coarse basis functions

$$\Phi_{i,\ell} = I^h(\chi_i \psi_{\ell}^{\omega_i}) \quad \text{for } 1 \leq i \leq N_v \text{ and } 1 \leq \ell \leq L_i, \tag{11}$$

where I^h is the fine-scale nodal value interpolation and L_i is an integer for each $i = 1, \ldots, N_v$. Denote by V_0 the *local spectral multiscale* space

$$V_0 = \text{span}\{\Phi_{i,\ell} : 1 \leq i \leq N_v \text{ and } 1 \leq \ell \leq L_i\}. \tag{12}$$

We note that in practice one only needs to computed the first L_i eigenvalues of (9). Hierarchical approximation with several triangulation can also be consider for the eigenvalues and eigenvectors.

Define also the coarse interpolation $I_0 : V^h(D) \to V_0$ by

$$I_0 v = \sum_{i=1}^{N_v} \sum_{\ell=1}^{L_i} \left(\int_{\omega_i} \widetilde{\kappa} v \psi_{\ell}^{\omega_i} \right) I^h(\chi_i \psi_{\ell}^{\omega_i}) = \sum_{i=1}^{N_v} I^h \left((I_{L_i}^{\omega_i} v) \chi_i \right), \tag{13}$$

where I^h is the fine-scale nodal value interpolation and $I_{L_i}^{\omega_i}$ is defined in (10).

We have the following weighted L^2 approximation and weighted H^1 stability properties.

Lemma 1. *For all coarse element K we have*

$$\int_K \widetilde{\kappa}(v - I_0 v)^2 \preceq \frac{1}{\widetilde{\lambda}_{K,L+1}} \int_{\omega_K} \kappa |\nabla v|^2 \tag{14}$$

$$\int_K \kappa |\nabla I_0 v|^2 \preceq \max\{1, \frac{1}{\widetilde{\lambda}_{K,L+1}}\} \int_{\omega_K} \kappa |\nabla v|^2 \tag{15}$$

where $\widetilde{\lambda}_{K,L+1} = \min_{y_i \in K} \widetilde{\lambda}_{L_i+1}^{\omega_i}$ and ω_K is the union of the elements that share common edge with K defined in (5).

The proof of this lemma follows from the results presented in [3] and will be presented elsewhere.

Using Lemma 1, we can estimate the condition number of the preconditioned operator $B^{-1}A$ with B^{-1} defined in (4) using the coarse space V_0 in (12). From the abstract domain decomposition theory we only need to prove the stable decomposition property; see [5]. From this stable decomposition property, one has the following Lemma.

Lemma 2. *The condition number of the preconditioned operator $B^{-1}A$ with B^{-1} defined in (4) is of order*

$$cond(B^{-1}A) \preceq C_0^2 \preceq 1 + \frac{1}{\widetilde{\lambda}_{L+1}}$$

where $\widetilde{\lambda}_{L+1} = \min\limits_{1 \le i \le N_v} \widetilde{\lambda}_{L_i+1}^{\omega_i}$.

It can be easily shown that the eigenvalues of the local problem scale as $O(1)$ assuming $\xi_i = \chi_i$, $i = 1, \ldots, N = N_v$, in (8). The dependency of the condition number of overlapping decomposition (δ) and coarse grid size (H) is controlled by the partition of unity $\{\xi_i\}$ and $\{\chi_i\}$ in (8), respectively. The condition number is independent of h and it is, in general, of order $O(H^2/\delta^2)$, see [3].

5 Numerical Results

In this section, we present representative $2D$ numerical results for the additive preconditioner (4) with the local spectral multiscale coarse space defined in (12). Numerical studies for the more interesting $3D$ case will be presented elsewhere. We take $D = [0,1] \times [0,1]$ that is divided into 8×8 equal square subdomains. Inside each subdomain we use a fine-scale triangulation where triangular elements constructed from 10×10 squares are used.

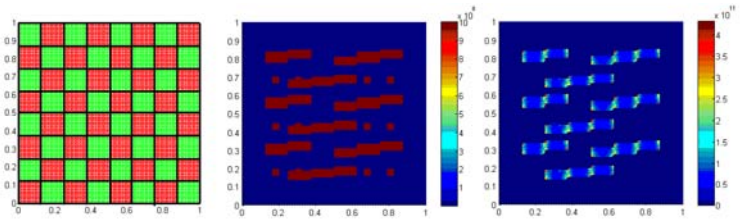


Fig. 2. *Left*: Coarse mesh. *Center*: Original coefficient. *Right*: Coefficient $\widetilde{\kappa}$ computed as in (8) using (linear) multiscale basis functions. See Table 1.

In our first numerical example, we choose a simple permeability field that only has isolated inclusions, see middle picture of Fig. 1. The coarse grid is demonstrated in the left picture. Multiscale finite element basis functions with linear boundary conditions are chosen as a partition of unity functions in (8). The purpose of this example is to demonstrate that $\widetilde{\kappa}$ does not have any high-conductivity components with this choice of partition of unity functions. As a result, we have only one eigenfunction (constant) per coarse grid. Thus, there is no need to complement the space of multiscale basis functions with linear boundary conditions. Note that if we use the eigenvalue problem with the weight function κ, then there will be four basis functions per node that represent inclusions. One can choose any $\widetilde{\kappa}$ that is larger than the one defined by (8). In our simulations, we add a positive constant to $\widetilde{\kappa}$ to avoid a numerical instability. In our numerical results, we observed that the number of iterations with the weight $\widetilde{\kappa} = \kappa$ and the weight $\widetilde{\kappa}$ defined in (8) (which results in the multiscale finite element basis functions) does not change for the contrast $\eta = 10^4, 10^5, 10^6, 10^7, 10^8$. The number of iterations is 22 iterations. Due to space limitation, we do not present detailed numerical results.

η	MS	EMF	LSM ($\widetilde{\kappa} = \kappa$)	LSM($\widetilde{\kappa}$ in (8))
10^4	98(2490.75)	62(257.86)	27(6.19)	28(7.34)
10^5	123(24866.24)	62(283.29)	28(6.19)	29(7.35)
10^6	144(248621.33)	62(286.12)	29(6.19)	29(7.35)
10^7	174(2486172.35)	63(286.41)	29(6.19)	30(7.35)
Dim	49	49	102	69

Table 1. Number of iterations until convergence and estimated condition number for the PCG and different values of the contrast η with the coefficient depicted in Fig. 2. We set the tolerance to 1e-10. Here $H = 1/8$ with $h = 1/80$. The notation MS stands for the (linear boundary condition) multiscale coarse space, EMF is the energy minimizing coarse space, see e.g., [7], and LSM is the local spectral multiscale coarse space defined in (12).

In the second example, we test our approach on a more complicated permeability field that contains inclusions and channels (see middle picture of Fig. 2). As before we use multiscale finite element basis functions as the initial partition of unity. From

the right picture of Fig. 2 we see that the modified weight $\widetilde{\kappa}$ does not contain any isolated inclusions and only contains long channels connecting boundaries of the coarse-grid block. This is achieved automatically from the choice of the partition of unity functions. There are fewer small (asymptotically vanishing) eigenvalues when local eigenvalue problem is solved with the modified weight $\widetilde{\kappa}$. Thus, with a good choice of partition of unity functions in (8), there are fewer new multiscale basis functions needed to achieve an optimal, in terms of the contrast, convergence. Numerical results are presented in Table 1. We observe that using the proposed coarse spaces, the number of iterations is independent of the contrast. In Table 1 we also show the dimension of the coarse spaces. The dimension of the local spectral coarse space is smaller if we use $\widetilde{\kappa}$ in (10) instead of $\widetilde{\kappa} = \kappa$ as in [3].

References

1. T. Chartier, R.D. Falgout, V.E. Henson, J. Jones, T. Manteuffel, S. McCormick, J. Ruge, and P.S. Vassilevski. Spectral AMGe (ρAMGe). *SIAM J. Sci. Comput.*, 25(1):1–26, 2003.
2. Y. Efendiev and T.Y. Hou. *Multiscale Finite Element Methods: Theory and applications*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer.
3. J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media. to appear in SIAM MMS.
4. I.G. Graham, P.O. Lechner, and R. Scheichl. Domain decomposition for multiscale PDEs. *Numer. Math.*, 106(4):589–626, 2007.
5. A. Toselli and O. Widlund. *Domain Decomposition Methods—Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2005.
6. J. Xu and Y. Zhu. Uniform convergent multigrid methods for elliptic problems with strongly discontinuous coefficients. *Math. Models Methods Appl. Sci.*, 18(1):77–105, 2008.
7. J. Xu and L. Zikatanov. On an energy minimizing basis for algebraic multigrid methods. *Comput. Vis. Sci.*, 7(3–4):121–127, 2004.
8. Y. Zhu. Domain decomposition preconditioners for elliptic equations with jump coefficients. *Numer. Linear Algebra Appl.*, 15(2–3):271–289, 2008.

Weighted Poincaré Inequalities and Applications in Domain Decomposition

Clemens Pechstein[1] and Robert Scheichl[2]

[1] Institute of Computational Mathematics, Johannes Kepler University Linz, 4040 Linz, Austria, Clemens.Pechstein@jku.at,
supported by the Austrian Sciences Fund (FWF) under grant P19255
[2] Department of Mathematical Sciences, University of Bath, Bath BA2 7AY, UK, R.Scheichl@bath.ac.uk

Summary. Poincaré type inequalities play a central role in the analysis of domain decomposition and multigrid methods for second-order elliptic problems. However, when the coefficient varies within a subdomain or within a coarse grid element, then standard condition number bounds for these methods may be overly pessimistic. In this short note we present new weighted Poincaré type inequalities for a class of piecewise constant coefficients that lead to sharper bounds independent of any possible large contrasts in the coefficients.

1 Introduction

Poincaré type inequalities play a central role in the analysis of domain decomposition (DD) methods for finite element discretisations of elliptic PDEs of the type

$$-\nabla \cdot (\alpha \nabla u) = f\,. \tag{1}$$

In many applications the coefficient α in (1) is discontinuous and varies over several orders of magnitude throughout the domain in a possibly very complicated way. Standard analyses of DD methods for (1) that use classical Poincaré type inequalities will often lead to pessimistic bounds. Two examples are the popular two-level overlapping Schwarz and FETI. If the subdomain partition can be chosen such that α is constant (or almost constant) on each subdomain as well as in each element of the coarse mesh (for two-level methods), then it is possible to prove bounds that are independent of the coefficient variation (cf. [2, 7, 14]). However, if this is not possible and the coefficient varies strongly within a subdomain, then the classical bounds depend on the local variation of the coefficient, which may be overly pessimistic in many cases. To obtain sharper bounds in some of these cases, it is possible to refine the standard analyses and use Poincaré inequalities on annulus type boundary layers of each subdomain [5, 8, 10, 12, 15], or weighted Poincaré type inequalities [4, 9, 13]. See also [1, 3, 6, 11, 16].

In this short note we want to collect and expand on the results in [4, 9] and present a new class of weighted Poincaré-type inequalities for a rather general class of piecewise constant coefficients. Due to space restrictions we have to refer the interested reader to [5, 8, 9, 13], to see where exactly these new inequalities can be used in the analysis of FETI and two-level Schwarz methods.

2 Weighted Poincaré Inequalities

Let D be a bounded domain in $\mathbb{R}^d$, $d = 2, 3$. For simplicity we only consider piecewise constant coefficient functions α with respect to a non-overlapping partitioning $\{Y_\ell : \ell = 1, \dots, n\}$ of D into open, connected Lipschitz polygons (polyhedra), i.e. $\alpha|_{Y_\ell} \equiv \alpha_\ell \equiv$ const. The results generalise straightforwardly to more general coefficients that vary mildly within each of the regions Y_ℓ.

Definition 1. *The region $P_{\ell_1,\ell_s} := (\overline{Y}_{\ell_1} \cup \overline{Y}_{\ell_2} \cup \cdots \cup \overline{Y}_{\ell_s})^\circ$ is called a type-m quasi-monotone path from Y_{ℓ_1} to Y_{ℓ_s}, if*
(i) for $i = 1, \dots, s-1$ the subregions $\overline{Y}_{\ell_i}$ and $\overline{Y}_{\ell_{i+1}}$ share a common m-dimensional manifold X_i
(ii) $\alpha_{\ell_1} \le \alpha_{\ell_2} \le \cdots \le \alpha_{\ell_s}$.

Definition 2. *Let $X^* \subset \overline{D}$ be a manifold of dimension m, with $0 \le m < d$. The coefficient distribution α is called type-m X^*-quasi-monotone on D, if for all $\ell = 1, \dots, n$ there exists an index k such that $X^* \subset \overline{Y}_k$ and such that there is a type-m quasi-monotone path $P_{\ell,k}$ from Y_ℓ to Y_k.*

Definition 3. *Let $\Gamma \subset \partial D$. The coefficient distribution α is called type-m Γ-quasi-monotone on D, if for all $\ell = 1, \dots, n$ there exists a manifold $X^*_\ell \subset \Gamma$ of dimension m and an index k such that $X^*_\ell \subset \partial Y_k$ and such that there is a type-m quasi-monotone path $P_{\ell,k}$ from Y_ℓ to Y_k.*

Note that the above definitions generalize the notion of quasi-monotone coefficients introduced in [2]. Definition 2 will be used to formulate weighted (discrete) Poincaré type inequalities, whereas Definition 3 will be used in weighted (discrete) Friedrichs inequalities. In Fig. 1 we give some examples of coefficient distributions that satisfy Definition 2.

To formulate our results we define for any $u \in H^1(D)$ the average

$$\overline{u}^{X^*} := \frac{1}{|X^*|} \int_{X^*} u \, ds \quad \text{if } m > 0, \qquad \overline{u}^{X^*} := u(X^*) \quad \text{if } m = 0,$$

as well as the weighted norm and seminorm

$$\|u\|_{L^2(D),\alpha} := \Big(\int_D \alpha \, |u|^2 \, dx \Big)^{1/2} \quad \text{and} \quad |u|_{H^1(D),\alpha} := \Big(\int_D \alpha \, |\nabla u|^2 \, dx \Big)^{1/2}.$$

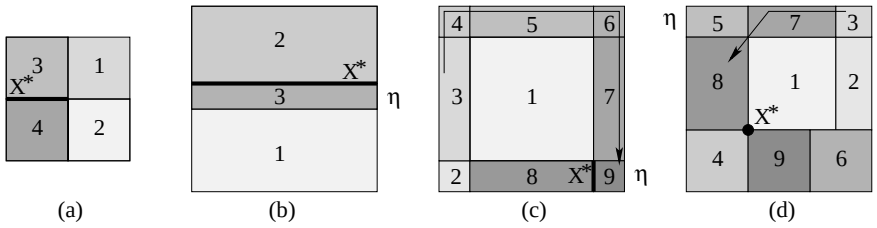


Fig. 1. Examples of quasi-monotone coefficients. The numbering of the regions is according to the relative size of the coefficients on these regions with the smallest coefficient in region Y_1. Note that the first case is quasi-monotone in the sense of [2], but the other three are not. The first three examples are type-1. The last example is type-0. The manifold X^* is shown in each case, together with a typical path in some of the cases.

Lemma 1 (weighted Poincaré inequality). *Let the coefficient α be type-$(d-1)$ X^*-quasi-monotone on D with the $(d-1)$-dimensional manifold X^*. For each index $\ell = 1, \ldots, n$, let $P_{\ell,k}$ be the path in Definition 2 with $X^* \subset \overline{Y_k}$, and let $C^P_{\ell,k} > 0$ be the best constant in the inequality*

$$\|u - \overline{u}^{X^*}\|^2_{L^2(Y_\ell)} \;\le\; C^P_{\ell,k}\,\mathrm{diam}(D)^2\,|u|^2_{H^1(P_{\ell,k})} \quad \text{for all } u \in H^1(P_{\ell,k}). \tag{2}$$

Then there exists a constant $C^P \le \sum_{\ell=1}^n C^P_{\ell,k}$ independent of α and $\mathrm{diam}(D)$ such that

$$\|u - \overline{u}^{X^*}\|^2_{L^2(D),\alpha} \;\le\; C^P\,\mathrm{diam}(D)^2\,|u|^2_{H^1(D),\alpha} \quad \text{for all } u \in H^1(D).$$

Proof. Let us fix one of the subregions Y_ℓ and suppose without loss of generality that $\int_{X^*} u\,ds = 0$ and that $\mathrm{diam}(D) = 1$. Due to the assumption on α, we have $\|u\|^2_{L^2(Y_\ell),\alpha} = \alpha_\ell \|u\|^2_{L^2(Y_\ell)}$. Combining this identity with inequality (2) and using that the coefficients are monotonically increasing in the path from Y_ℓ to Y_k, we obtain

$$\|u\|^2_{L^2(Y_\ell),\alpha} \;\le\; C^P_{\ell,k}\,\alpha_\ell\,|u|^2_{H^1(P_{\ell,k})} \;\le\; C^P_{\ell,k}\,|u|^2_{H^1(P_{\ell,k}),\alpha} \;\le\; C^P_{\ell,k}\,|u|^2_{H^1(D),\alpha}\,.$$

The proof is completed by adding up the above estimates for $\ell = 1, \ldots, n$.

Remark 1. Obviously, inequality (2) follows from the standard Poincaré type inequality $\|u - \overline{u}^{X^*}\|^2_{L^2(P_{\ell,k})} \le C\,|u|_{H^1(P_{\ell,k})}$ for all $u \in H^1(P_{\ell,k})$, with some constant C depending on $P_{\ell,k}$ and on X^*. However, this may lead to a sub-optimal constant. In general, the constants $C^P_{\ell,k}$ depend on the choice of the manifold X^*, as well as on the number, shape, and size of the subregions Y_ℓ. In Sect. 3, we give a bound of $C^P_{\ell,k}$ in terms of local Poincaré constants on the individual subregions Y_ℓ to make this dependency more explicit.

On the other hand, if X^* is a manifold of dimension less than $d-1$ (i.e. an edge or a point), inequality (2) does not hold for all functions $u \in H^1(D)$. However, there is a discrete analogue for finite element functions which holds under some geometric assumptions on the subregions Y_ℓ, cf. [14, Sect. 4.6].

Let $\{\mathcal{T}_h(D)\}$ be a family of quasi-uniform, simplicial triangulations of D with mesh width h. By $V^h(D)$ we denote the space of continuous piecewise linear functions with respect to the elements of $\mathcal{T}_h(D)$. Note that we do not prescribe any boundary conditions. We further assume that the fine mesh $\mathcal{T}_h(D)$ resolves the interfaces between the subregions Y_ℓ.

Assumption 1 (cf. [14, Assumption 4.3]) There exists a parameter η with $h \leq \eta \leq \operatorname{diam}(D)$ such that each subregion Y_ℓ is the union of a few simplices of diameter η, and the resulting coarse mesh is globally conforming on all of D.

Before stating the next lemma, we define the function

$$\sigma^\delta(x) := \begin{cases} (1+\log(x)) & \text{for } \delta = 2, \\ x & \text{for } \delta = 3. \end{cases} \tag{3}$$

Lemma 2 (weighted discrete Poincaré inequality). *Let Assumption 1 hold and let α be type-m X^*-quasi-monotone on D with the manifold X^* having dimension $m < d-1$. If $m = 1$, assume furthermore that X^* is an edge of the coarse triangulation in Assumption 1. For each $\ell = 1, \ldots, n$, let $P_{\ell,k}$ be the path in Definition 2 with $X^* \subset \overline{Y}_k$ and let $C_{\ell,k}^{P,m} > 0$ be the best constant independent of h such that*

$$\|u - \overline{u}^{X^*}\|^2_{L^2(Y_\ell)} \leq C_{\ell,k}^{P,m}\, \sigma^{d-m}\left(\tfrac{\eta}{h}\right) \operatorname{diam}(D)^2\, |u|^2_{H^1(P_{\ell,k})} \quad \text{for all } u \in V^h(P_{\ell,k}). \tag{4}$$

Then, there exists a constant $C^{P,m} \leq \sum_{\ell=1}^n C_{\ell,k}^{P,m}$, independent of h, of α, and of $\operatorname{diam}(D)$ such that

$$\|u - \overline{u}^{X^*}\|^2_{L^2(D),\alpha} \leq C^{P,m}\, \sigma^{d-m}\left(\tfrac{\eta}{h}\right) \operatorname{diam}(D)^2\, |u|^2_{H^1(D),\alpha} \quad \text{for all } u \in V^h(D).$$

Proof. The proof is analogous to that of Lemma 1, but uses (4) instead of (2).

We remark that the existence of the constants $C_{\ell,k}^{P,m}$ fulfilling inequality (4) will follow from the results summarized in [14, Sect. 4.6] and from our investigation in Sect. 3. For simplicity, let us also define $\sigma^1 \equiv 1$ and $C^{P,d-1} := C^P$.

We would like to mention that similar inequalities than those in Lemmas 1 and 2 can also be proved, if u vanishes on part of the boundary of D. Here, we just display the case $m = d-1$. The generalisation to $m < d-1$ is straightforward and follows Lemma 2.

Lemma 3 (weighted Friedrichs inequality). *Let $\Gamma \subset \partial D$ and let α be type-$(d-1)$ Γ-quasi-monotone on D (according to Definition 3). Then there exists a constant $C^F = C^{F,d-1}$ independent of α and of $\operatorname{diam}(D)$ such that*

$$\|u\|^2_{L^2(D),\alpha} \leq C^F \operatorname{diam}(D)^2\, |u|^2_{H^1(D),\alpha} \quad \text{for all } u \in H^1(D),\ u|_\Gamma = 0.$$

3 Explicit Dependence on Geometrical Parameters

In this section we will study the dependence of the constants $C_{\ell,k}^{P,m}$ (and consequently $C^{P,m}$) in the above lemmas on the choice of X^* and on the number, size

and shape of the regions Y_ℓ (in particular the ratio $\operatorname{diam}(D)/\eta$). In [9, §3] the dependence on the geometry of the subregions is made more explicit. The lemmas presented there are in fact special cases of Lemmas 1 and 2 here.

First, we show that bounds for the constants $C_{\ell,\,k}^{P,\,m}$ can be obtained from inequalities on the individual subregions Y_ℓ. Secondly, we will look at a series of examples in two dimensions. For further 3D examples see [9].

Lemma 4. *Let α be type-m X^*-quasi-monotone on D with $0 \le m \le d-1$, and let $P_{\ell_1,\,\ell_s}$ be any of the paths in Definition 2. If $m < d-1$, let Assumption 1 hold. If $m = 1$ and $d = 3$, assume additionally that X^* is an edge of the coarse triangulation. For each $i = 1, \dots, s$, let $C_{\ell_i}^{P,m}$ be the best constant, such that*

$$\|u - \overline{u}^X\|^2_{L^2(Y_{\ell_i})} \;\le\; C_{\ell_i}^{P,\,m}\, \sigma^{d-m}\!\left(\tfrac{\eta}{h}\right)\, \operatorname{diam}(Y_{\ell_i})^2\, |u|^2_{H^1(Y_{\ell_i})} \quad \text{for all } u \in V^h(Y_{\ell_i}), \tag{5}$$

where $X \subset \overline{Y}_{\ell_i}$ is any of the manifolds X_{i-1}, X_i or X^* in Definition 2 (as appropriate), cf. [14, Sect. 4.6]. Then

$$C_{\ell_1,\,\ell_s}^{P,\,m} \;\le\; 4\left\{\sum_{i=1}^{s} \frac{\operatorname{meas}(Y_{\ell_1})}{\operatorname{meas}(Y_{\ell_i})}\, \frac{\operatorname{diam}(Y_{\ell_i})^2}{\operatorname{diam}(D)^2}\, C_{\ell_i}^{P,\,m}\right\}.$$

If $m = d-1$ we can extend the result to the whole of H^1.

Proof. We give the proof for the case $m = d-1$. The other cases are analogous. For convenience let $X_s := X^*$. Then, telescoping yields

$$\|u - \overline{u}^{X^*}\|_{L^2(Y_{\ell_1})} \;\le\; \|u - \overline{u}^{X_1}\|_{L^2(Y_{\ell_1})} + \sum_{i=2}^{s} \sqrt{\operatorname{meas}(Y_{\ell_1})}\left|\overline{u}^{X_{i-1}} - \overline{u}^{X_i}\right|.$$

Due to (5), $\|u - \overline{u}^{X_1}\|_{L^2(Y_{\ell_1})} \le \sqrt{C_{\ell_1}^{P,\,m}}\, \operatorname{diam}(Y_{\ell_1})\, |u|_{H^1(Y_{\ell_1})}$, and for each i,

$$\begin{aligned}\left|\overline{u}^{X_{i-1}} - \overline{u}^{X_i}\right|^2 &\le \frac{2}{\operatorname{meas}(Y_{\ell_i})}\left(\|\overline{u}^{X_{i-1}} - u\|^2_{L^2(Y_{\ell_i})} + \|u - \overline{u}^{X_i}\|^2_{L^2(Y_{\ell_i})}\right)\\ &\le \frac{4}{\operatorname{meas}(Y_{\ell_i})}\, C_{\ell_i}^{P,\,m}\, \operatorname{diam}(Y_{\ell_i})^2\, |u|^2_{H^1(Y_{\ell_i})}.\end{aligned}$$

An application of Cauchy's inequality (in $\mathbb{R}^s$) yields the final result.

Let us look at some examples now. Firstly, if Assumption 1 holds with constant $\eta \gtrsim \operatorname{diam}(D)$ (e. g. in Fig. 1a), then $n = \mathcal{O}(1)$ and each path $P_{\ell,k}$ in Definition 2 contains $\mathcal{O}(1)$ subregions. If we choose X^* to be a vertex, edge or face of the coarse triangulation in Assumption 1, then by standard arguments $C_\ell^{P,\,m} = \mathcal{O}(1)$ in (5) for all $\ell = 1, \dots, n$. Hence, it follows from Lemma 4 that the constants $C^{P,\,m}$ in Lemmas 1–2 are all $\mathcal{O}(1)$.

Before we look to more complicated examples, which involve in particular long, thin regions, let us first derive two auxiliary results.

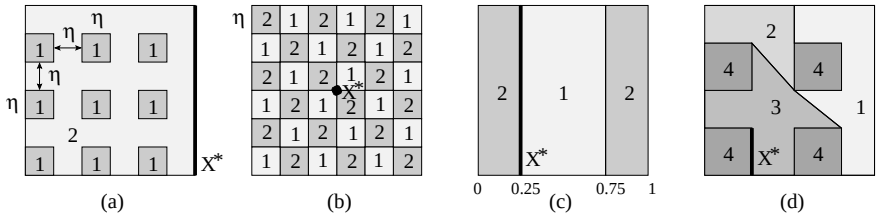


Fig. 2. More examples (with $\alpha_1 \ll \alpha_2$): The first two examples are quasi-monotone of type-1 and type-0, respectively. X^* is shown in each case. The examples in (**c**) and (**d**) are not quasi-monotone.

(i) The middle region Y_3 in Fig. 1b is long and thin if $\eta \ll \operatorname{diam}(Y_3)$. With X^* as given in the figure, one can show that (5) holds with $C_3^{P,1} = \mathcal{O}(1)$, independent of η and $\operatorname{diam}(Y_3)$. Note that $\operatorname{diam}(X^*) \simeq \operatorname{diam}(Y_3)$.

(ii) The region Y_8 in Fig. 1c has essentially the same shape, but here X^* has diameter $\eta \ll \operatorname{diam}(Y_8)$. Nevertheless, one can show that (5) holds with $X = X^*$ and $C_8^{P,1} = \mathcal{O}(1)$, independent of η and $\operatorname{diam}(Y_8)$. (This result can be obtained by sub-dividing Y_8 into small quadrilaterals of sidelength η and applying Lemma 4).

In Figs. 1 and 2, H denotes the sidelength of D (thus, $H \simeq \operatorname{diam}(D)$). We view η (if displayed) as a varying parameter $\leq H$, with the other parameters fixed.

Fig. 1b. As just discussed, $C_{3,3}^{P,1} = C_3^{P,1} = \mathcal{O}(1)$. Similarly, $C_{2,2}^{P,1} = C_2^{P,1} = \mathcal{O}(1)$. To obtain $C_{1,3}^{P,1} = \mathcal{O}(1)$ we use $\|u - \overline{u}^{X^*}\|^2_{L^2(Y_1)} \leq \|u - \overline{u}^{X^*}\|^2_{L^2(P_{1,3})}$ and apply a standard Poincaré inequality (rather than resorting to Lemma 4 which would yield a pessimistic bound). Hence, Lemma 1 holds with $C^{P,1} = \mathcal{O}(1)$.

Fig. 1c. Despite the fact that $C_1^{P,1} = \mathcal{O}(1)$ and $C_8^{P,1} = \mathcal{O}(1)$, the constant $C_{1,8}^{P,1}$ is not $\mathcal{O}(1)$: Since $\operatorname{diam}(Y_1) \sim H$, Lemma 4 yields

$$C_{1,8}^{P,1} \lesssim \frac{H^2}{H^2}\frac{H^2}{H^2} + \frac{H^2}{H\,\eta}\frac{H^2}{H^2} = \mathcal{O}\Big(\frac{H}{\eta}\Big).$$

We easily convince ourselves that this is the worst constant $C_{\ell,k}^{P,1}$, for all $\ell = 1, \ldots, 9$ (e. g., $C_{3,9}^{P,1} = \mathcal{O}(1)$), and so we obtain $C^{P,1} = \mathcal{O}(\frac{H}{\eta})$.

Fig. 1d. Here the coefficient is only type-0 quasi-monotone and so we cannot apply Lemma 1, but by applying Lemma 4 we find that $C_{7,8}^{P,0} = \mathcal{O}(1)$ and all the other constants are no worse. So in contrast to Case (c), we can show that the constant $C^{P,0}$ in Lemma 2 is $\mathcal{O}(1)$ in this case. The crucial difference is not that α is type-0 here, but that $\operatorname{diam}(Y_8) = \mathcal{O}(H)$ and $\operatorname{diam}(Y_9) = \mathcal{O}(H)$.

The examples in Fig. 2 are further, typical test cases used in the literature.

Fig. 2a. To obtain a sharp bound for $C^{P,1}$, it is better here to treat all the regions where $\alpha = \alpha_1$ as one single region Y_1, slightly modifying the proof of Lemma 1. Then $C_{1,2}^{P,1} = \mathcal{O}(1)$ (standard Poincaré on D). Due to a tricky overlapping argument that can be found in the Appendix of [8], $C_{2,2}^{P,1} = \mathcal{O}(1)$. Thus, $C^{P,1} = \mathcal{O}(1)$. Note that this is only possible if α takes the same values on all the inclusions. If there are

p distinct values in the inclusions, the constant $C^{P,1}$ depends (linearly) on p. This should be compared with one of the main results in [4], where a similar Poincaré inequality is proved with a constant depending on the number of inclusions.

Fig. 2b. For each region Y_ℓ we have $C_\ell^{P,0} = C_\square^{P,0} = \mathcal{O}(1)$. For a moment, let us restrict on the regions where the coefficient is α_1 and group them into $T := \frac{H}{2\eta}$ concentric layers starting from the two centre squares touching X^* where $\alpha = \alpha_1$. Obviously, for $t = 1, \ldots, T$, layer t contains $2t-2$ regions where $\alpha = \alpha_1$. Each region in layer t can be connected to one of the two centre squares by a type-0 quasi-monotone path of length t. By Lemma 4, $C_{\ell,k}^{P,0} \leq 4 \sum_{j=1}^{t} \frac{\eta^2}{H^2} C_\square^{P,0} = 4\,t\,\frac{\eta^2}{H^2} C_\square^{P,0}$ for all the regions Y_ℓ in layer t where $\alpha = \alpha_1$. The same bound holds for the regions where $\alpha = \alpha_2$. Summing up these bounds over all regions and all layers, we obtain

$$C^{P,0} \;\leq\; 2 \sum_{t=1}^{T} (2t-2)\, 4\, t\, \frac{\eta^2}{H^2}\, C_\square^{P,0} \;=\; 16\, \frac{\eta^2}{H^2}\, \frac{T^3 - T}{3} \;=\; \mathcal{O}\Big(\frac{H}{\eta}\Big).$$

Equivalently, as there are $n_\times = \mathcal{O}\big(\frac{H}{\eta}\big)^2$ crosspoints in this example, we have shown that $C^{P,0} = \mathcal{O}(\sqrt{n_\times})$. An enhanced bound of $\mathcal{O}\big((1 + \log(H/\eta))^2\big)$ for $C^{P,0}$ in this example can be obtained using a multilevel argument, and will be proved in an upcoming paper.

Fig. 2c. α is not quasi-monotone in this case, and indeed Lemmas 1–3 do not hold. For example, if we choose X^* as shown, then it suffices to choose u to be the continuous function that is equal to $2(x_1 - \frac{1}{4})$ for $\frac{1}{4} \leq x_1 \leq \frac{3}{4}$ and constant otherwise, to obtain a counter example in $V^h(D) \subset H^1(D)$ that satisfies $\overline{u}^{X^*} = 0$. We have $\|u\|^2_{L^2(D),\alpha} = \frac{\alpha_1}{6} + \frac{\alpha_2}{4}$ and $|u|^2_{H^1(D),\alpha} = 2\alpha_1$, and so the constant $C^{P,1}$ in Lemma 1 blows up with the contrast $\frac{\alpha_2}{\alpha_1}$. It is impossible to find X^* such that Lemma 2 holds.

Fig. 2d. Again α is not quasi-monotone and Lemmas 1–3 do not hold on all of the domain D. However, by choosing suitable (energy-minimising) coarse space basis functions in two-level Schwarz methods (cf. [5, 12, 15]), it often suffices to be able to apply Lemmas 1–3 on $D' := Y_1 \cup Y_2 \cup Y_3$. Since α is type-1 quasi-monotone on D', e.g. Lemma 1 holds for $u \in H^1(D')$ and it is easy to verify that $C^{P,1} = \mathcal{O}(1)$.

References

1. C.R. Dohrmann, A. Klawonn, and O.B. Widlund. Domain decomposition for less regular subdomains: Overlapping Schwarz in two dimensions. *SIAM J. Numer. Anal.*, 46:2153–2168, 2008.
2. M. Dryja, M.V. Sarkis, and O.B. Widlund. Multilevel Schwarz methods for elliptic problems with discontinuous coefficients in three dimensions. *Numer. Math.*, 72:313–348, 1996.
3. Y. Efendiev and J. Galvis. Domain decomposition preconditioners for multiscale problems. Preprint, Texas A&M University, 2009. Submitted.
4. J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media. Preprint, Texas A&M University, 2009. Submitted.

5. I.G. Graham, P. Lechner, and R. Scheichl. Domain decomposition for multiscale PDEs. *Numer. Math.*, 106:589–626, 2007.
6. A. Klawonn, O. Rheinbach, and O.B. Widlund. An analysis of a FETI-DP algorithm on irregular subdomains in the plane. *SIAM J. Numer. Anal.*, 46:2484–2504, 2008.
7. A. Klawonn and O.B. Widlund. FETI and Neumann–Neumann iterative substructuring methods: connections and new results. *Commun. Pure Appl. Math.*, 54:57–90, 2001.
8. C. Pechstein and R. Scheichl. Analysis of FETI methods for multiscale PDEs. *Numer. Math.*, 111:293–333, 2008.
9. C. Pechstein and R. Scheichl. Analysis of FETI methods for multiscale PDEs - Part II: Interface variation. BICS Preprint 7/09, University of Bath, 2009. Submitted.
10. C. Pechstein and R. Scheichl. Scaling up through domain decomposition. *Appl. Anal.*, 88 (10), 2009.
11. M. Sarkis. Nonstandard coarse spaces and Schwarz methods for elliptic problems with discontinuous coefficients using non-conforming elements. *Numer. Math.*, 77:383–406, 1997.
12. R. Scheichl and E. Vainikko. Additive Schwarz and aggregation-based coarsening for elliptic problems with highly variable coefficients. *Computing*, 80:319–343, 2007.
13. R. Scheichl, P.S. Vassilevski, and L. Zikatanov. Two-level Schwarz with non-matching coarse grids. Preprint, Lawrence Livermore National Labs, 2009.
14. A. Toselli and O. Widlund. *Domain Decomposition Methods – Algorithms and Theory*. Springer, Berlin, 2005.
15. J. Van lent, R. Scheichl, and I.G. Graham. Energy minimizing coarse spaces for two-level Schwarz methods for multiscale PDEs. *Numer. Linear Algebra Appl.*, 16:775–799, 2009.
16. Y. Zhu. Domain decomposition preconditioners for elliptic equations with jump coefficients. *Numer. Linear Algebra Appl.*, 15:271–289, 2008.

Technical Tools for Boundary Layers and Applications to Heterogeneous Coefficients

Maksymilian Dryja[1] and Marcus Sarkis[2,3]

[1] Department of Mathematics, Warsaw University, Banacha 2, 02-097 Warsaw, Poland. This work was supported in part by the Polish Sciences Foundation under grant NN201006933.
[2] Instituto Nacional de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Rio de Janeiro 22460-320, Brazil
[3] Department of Mathematical Sciences, Worcester Polytechnic Institute, Worcester, MA 01609, USA

1 Summary

We consider traces and discrete harmonic extensions on thin boundary layers. We introduce *sharp* estimates on how to control the $H^{1/2}-$ or $H_{00}^{1/2}-$ boundary norm of a finite element function by its energy in a thin layer and vice versa, how to control the energy of a discrete harmonic function in a layer by the $H^{1/2}$ or $H_{00}^{1/2}$ norm on the boundary. Such results play an important role in the analysis of domain decomposition methods in the presence of high-contrast media inclusions, small overlap and/or inexact solvers.

2 Introduction and Assumptions

Let Ω be a well-shaped polygonal domain of diameter $O(1)$ in $\mathbb{R}^2$. We assume that the substructures $\Omega_i, 1 \leq i \leq N$, are well-shaped polygonal domains of diameters $O(H_i)$, and also assume that the Ω_i form a geometrically conforming nonoverlapping partitioning of Ω. Let $\mathcal{T}^{h_i}(\Omega_i)$ be a conforming shape-regular simplicial triangulation of Ω_i where h_i denotes the smallest diameter of the simplices of $\mathcal{T}^{h_i}(\Omega_i)$. We assume that the union of the triangulations $\mathcal{T}^{h_i}(\Omega_i)$, which we denote by $\mathcal{T}^h(\Omega)$, forms a conforming triangulation for Ω.

For purpose of analysis, let us introduce an auxiliary conforming shape-regular simplicial triangulation $\mathcal{T}^{\eta_i}(\Omega_i)$ of Ω_i where η_i denotes the smallest diameter of its simplices of $\mathcal{T}^{\eta_i}(\Omega_i)$. We do not assume that the triangulations $\mathcal{T}^{\eta_i}(\Omega_i)$ and $\mathcal{T}^{h_i}(\Omega_i)$ are nested. Let us introduce the boundary layer $\Omega_{i,\eta_i} \subset \Omega_i$ of width $O(\eta_i)$ as the union of all simplices of $\mathcal{T}^{\eta_i}(\Omega_i)$ that touch $\partial\Omega_i$ in at least one point. We assume that the mesh parameter η_i is large enough compared to h_i in the sense that all simplices of $\mathcal{T}^{h_i}(\Omega_i)$ that touch $\partial\Omega_i$ must be contained in Ω_{i,η_i}. We also introduce

the boundary layer Ω'_{i,η_i} of width $O(\eta_i)$ as the union of all simplices of $\mathcal{T}^{h_i}(\Omega_i)$ which intersect Ω_{i,η_i}, hence, it is easy to see that $\Omega_{i,\eta_i} \subset \Omega'_{i,\eta_i}$. We denote by $\mathcal{T}^{\eta_i}(\Omega_{i,\eta_i})$ the triangulation of $\mathcal{T}^{\eta_i}(\Omega_i)$ restricted to Ω_{i,η_i}, and by $\mathcal{T}^{h_i}(\Omega'_{i,\eta_i})$ the triangulation of $\mathcal{T}^{h_i}(\Omega_i)$ restricted to Ω'_{i,η_i}. Throughout the paper, the notation $c \preceq d$ (for quantities c and d) means that c/d is bounded from above by a positive constant independently of h_i, H_i, η_i and ρ_i. Moreover, $c \asymp d$ means $c \preceq d$ and $d \preceq c$. We also use $c \leq d$ to stress that $c/d \leq 1$.

We study the following selfadjoint second order elliptic problem:

Find $u^* \in H_0^1(\Omega)$ such that

$$a_\rho(u^*, v) = f(v), \quad \forall v \in H_0^1(\Omega) \tag{1}$$

where

$$a_\rho(u^*, v) := \sum_{i=1}^{N} \int_{\Omega_i} \rho_i(x) \nabla u^* \cdot \nabla v \, dx \quad \text{and} \quad f(v) := \int_{\Omega_i} f v \, dx \quad \text{for} \quad f \in L^2(\Omega).$$

We assume that $0 < c_i \leq \rho_i(x) \leq C_i$ for any $x \in \Omega_i$. We note that the condition number estimates of the preconditioned systems considered in this paper do not depend on the constants c_i and C_i.

Definition We say that a coefficient ρ_i satisfies the *Boundary Layer Assumption* on Ω_i if $\rho_i(x)$ is equal to a constant $\bar{\rho}_i$ for any $x \in \Omega'_{i,\eta_i}$.

Definition We say that a coefficient ρ_i associated to a subdomain Ω_i is of the *Inclusion Hard* type or *Inclusion Soft* type if the *Boundary Layer Assumption* holds with $\rho_i(x) = \bar{\rho}_i$ on Ω'_{i,η_i}, and

- *Inclusion Hard* type: $\rho_i(x) \succeq \bar{\rho}_i$ for all $x \in \Omega_i \backslash \Omega'_{i,\eta_i}$,
- *Inclusion Soft* type: $\rho_i(x) \preceq \bar{\rho}_i$ for all $x \in \Omega_i \backslash \Omega'_{i,\eta_i}$.

We allow the coefficients $\{\bar{\rho}_i\}_{i=1}^N$ to have large jumps across the interface of the subdomains $\Gamma := (\cup_{i=1}^N \partial\Omega_i) \backslash \partial\Omega$. The results to be presented in this paper can be extended easily to moderate variations of the coefficients ρ_i on Ω'_{i,η_i}.

We point out that the extension of our results to problems where the coefficient ρ_i has large jumps inside Ω'_{i,η_i} is not trivial. We point out, however, that for certain distributions of coefficients ρ_i where weighted Poincaré type inequalities are explicitly given (see [7]), the technical tools introduced here can be applied to derive sharper analysis. For instance, in the case where a hard inclusion G crosses an edge $E_{ij} := \partial\Omega_i \cap \partial\Omega_j$, we can impose primal constraints to guarantee average continuity on each connected component of $G \cap E_{ij}$; see numerical experiments on [3]. See also the related work on energy minimizing coarse spaces [5] and on expensive and robust methods based on enhanced partition of unity coarse spaces based on eigenvalue problems [1, 8] on the diagonally scaled system, see Remark 4.1 of [1], or equivalently, using generalized eigenvalue problems on the original system [4].

3 Technical Tools for Layers

We now introduce technical tools that are essential for obtaining sharp bounds for certain domain decomposition methods. The next lemma shows how $|w|_{H^{1/2}(\partial\Omega_i)}$ can be controlled by the energy of w on Ω_{i,η_i}.

Lemma 1. *Let $w \in H^1(\Omega_{i,\eta_i})$. Then*

$$|w|^2_{H^{1/2}(\partial\Omega_i)} \preceq \frac{H_i}{\eta_i}|w|^2_{H^1(\Omega_{i,\eta_i})}. \tag{2}$$

Proof. Let $V^{\eta_i}(\Omega_{i,\eta_i}) \subset H^1(\Omega_{i,\eta_i})$ be the space of piecewise linear and continuous functions associated to $\mathcal{T}_{\eta_i}(\Omega_{i,\eta_i})$. Let Π^{η_i} be the Zhang–Scott–Clemént interpolation operator from $H^1(\Omega_{i,\eta_i})$ to $V^{\eta_i}(\Omega_{i,\eta_i})$. Using a triangular inequality we obtain

$$|w|^2_{H^{1/2}(\partial\Omega_i)} \leq 2\left(|w-\Pi^{\eta_i}w|^2_{H^{1/2}(\partial\Omega_i)} + |\Pi^{\eta_i}w|^2_{H^{1/2}(\partial\Omega_i)}\right). \tag{3}$$

We now estimate the first term of the right-hand side of (3). Let us first define the cut-off function θ_i on Ω_i which equals to one on $\partial\Omega_i$, equals to zero at all interior nodes of $\mathcal{T}^{\eta_i}(\Omega_i)$ and is linear in each element of $\mathcal{T}^{\eta_i}(\Omega_i)$. Note that $0 \leq \theta_i(x) \leq 1$ for $x \in \Omega_i$, $\theta_i(x) = 1$ for $x \in \partial\Omega_i$, $\theta_i(x) = 0$ for $x \in \Omega_i \backslash \Omega_{i,\eta_i}$, and $\|\theta_i\|_{W^{1,\infty}(\Omega_{i,\eta_i})} \preceq 1/\eta_i$. Denoting by $z = w - \Pi^{\eta_i}w$ on Ω_{i,η_i} and using trace and minimal energy arguments plus standard calculations we obtain

$$|z|^2_{H^{1/2}(\partial\Omega_i)} \preceq |\theta_i z|^2_{H^1(\Omega_{i,\eta_i})} \preceq |z|^2_{H^1(\Omega_{i,\eta_i})} + \frac{1}{\eta_i^2}\|z\|^2_{L^2(\Omega_{i,\eta_i})}. \tag{4}$$

The right-hand side of (4) can be bounded by $|w|^2_{H^1(\Omega_{i,\eta_i})}$ by using the $H^1(\Omega_{i,\eta_i})$-stability and the $L_2(\Omega_{i,\eta_i})$-approximation properties of the Zhang–Scott–Clemént interpolation operator Π^{η_i}. We note that the proofs of these properties are based only on local arguments, therefore, they hold also for domains like Ω_{i,η_i}.

We now estimate the second term of the right-hand side of (3). We first use scaling and embedding arguments to obtain

$$|\Pi^{\eta_i}w|^2_{H^{1/2}(\partial\Omega_i)} \preceq H_i|\Pi^{\eta_i}w|^2_{H^1(\partial\Omega_i)}. \tag{5}$$

To bound the right-hand side of (5), let us first introduce the subregion $\hat{\Omega}_{i,\eta_i} \subset \Omega_{i,\eta_i}$ as the union of elements of $\mathcal{T}^{\eta_i}(\Omega_{i,\eta_i})$ which have an edge on $\partial\Omega_i$. Using only properties of linear elements of $V^{\eta_i}(\Omega_{i,\eta_i})$ we have

$$H_i|\Pi^{\eta_i}w|^2_{H^1(\partial\Omega_i)} \preceq \frac{H_i}{\eta_i}|\Pi^{\eta_i}w|^2_{H^1(\hat{\Omega}_{i,\eta_i})} \leq \frac{H_i}{\eta_i}|\Pi^{\eta_i}w|^2_{H^1(\Omega_{i,\eta_i})}. \tag{6}$$

The lemma follows by using the $H^1(\Omega_{i,\eta_i})$-stability of the Zhang–Scott–Clemént interpolation operator.

3.1 Technical Tools for DDMs

In this section we present the technical tools necessary to establish sharp analysis for exact and inexact two-dimensional FETI-DP with vertex constraints. More general technical tools can also be extended to obtain sharp analysis for non-overlapping Schwarz methods such as FETI and FETI-DP with edge and vertex primal constraints [9], additive average Schwarz methods [2], inexact iterative substructuring methods and for three-dimensional problems; see [3].

Let $w \in V^{h_i}(\partial\Omega_i)$. Define the following discrete harmonic extensions:

(i) The $\mathcal{H}^{(i)}_{\rho_i} w \in V^{h_i}(\Omega_i)$ as the ρ_i-discrete harmonic extension of w inside Ω_i, i.e., $\mathcal{H}^{(i)}_{\rho_i} w = w$ on $\partial\Omega_i$ and

$$\int_{\Omega_i} \rho_i(x) \nabla \mathcal{H}^{(i)}_{\rho_i} w \cdot \nabla v \, dx = 0 \text{ for any } \; v \in V_0^{h_i}(\Omega_i). \tag{7}$$

Here, $V_0^{h_i}(\Omega_i)$ is the space of functions of $V^{h_i}(\Omega_i)$ which vanish on $\partial\Omega_i$.

(ii) The $\mathcal{H}^{(i)}_{\rho_i,\mathcal{D}} w \in V^{h_i}(\Omega'_{i,\eta_i})$ as the zero Dirichlet boundary layer harmonic extension of w inside Ω'_{i,η_i}, i.e., $\mathcal{H}^{(i)}_{\rho_i,\mathcal{D}} w = w$ on $\partial\Omega_i$ and $\mathcal{H}^{(i)}_{\rho_i,\mathcal{D}} w = 0$ on $\partial\Omega'_{i,\eta_i} \backslash \partial\Omega_i$, and

$$\int_{\Omega'_{i,\eta_i}} \rho_i(x) \nabla \mathcal{H}^{(i)}_{\rho_i,\mathcal{D}} w \cdot \nabla v \, dx = 0 \text{ for any } \; v \in V^{h_i}_{0,\mathcal{D}}(\Omega'_{i,\eta_i}).$$

Here, $V^{h_i}(\Omega'_{i,\eta_i})$ is the space of continuous piecewise linear finite elements on $\mathcal{T}^{h_i}(\Omega'_{i,\eta_i})$, and $V^{h_i}_{0,\mathcal{D}}(\Omega'_{i,\eta_i})$ is the space of functions of $V^{h_i}(\Omega'_{i,\eta_i})$ which vanish on $\partial\Omega'_{i,\eta_i}$.

(iii) The $\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w \in V^{h_i}(\Omega'_{i,\eta_i})$ as the zero Neumann boundary layer harmonic extension of w inside Ω'_{i,η_i}, i.e.,$\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w = w$ only on $\partial\Omega_i$ and

$$\int_{\Omega'_{i,\eta_i}} \rho_i(x) \nabla \mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w \cdot \nabla v \, dx = 0 \text{ for any } \; v \in V^{h_i}_{0,\mathcal{N}}(\Omega'_{i,\eta_i}).$$

Here, $V^{h_i}_{0,\mathcal{N}}(\Omega'_{i,\eta_i})$ is the space of functions of $V^{h_i}(\Omega'_{i,\eta_i})$ which vanish on $\partial\Omega_i$.

Lemma 2. *Let us assume that the Boundary Layer Assumption holds on Ω_i and let $w \in V^{h_i}(\partial\Omega_i)$. Then*

$$|\mathcal{H}^{(i)}_{\rho_i} w|^2_{H^1_{\rho_i}(\Omega_i)} \le |\mathcal{H}^{(i)}_{\rho_i,\mathcal{D}} w|^2_{H^1_{\rho_i}(\Omega'_{i,\eta_i})} \preceq |\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w|^2_{H^1_{\rho_i}(\Omega'_{i,\eta_i})} + \frac{\bar{\rho}_i}{\eta_i} \|w\|^2_{L^2(\partial\Omega_i)}. \tag{8}$$

When $\rho_i(x) \preceq \bar{\rho}_i$ (Inclusion Soft type) on Ω_i, then

$$|\mathcal{H}^{(i)}_{\rho_i} w|^2_{H^1_{\rho_i}(\Omega_i)} \preceq \bar{\rho}_i |w|^2_{H^{1/2}(\partial\Omega_i)} \preceq \frac{H_i}{\eta_i} |\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w|^2_{H^1_{\rho_i}(\Omega'_{i,\eta_i})}. \tag{9}$$

Proof. The result (8) follows from [6]; see also [3] for an alternative proof. The result (9) follows from Lemma 1 and the fact that $\Omega_{i,\eta_i} \subset \Omega'_{i,\eta_i}$.

Let E be an edge of $\partial\Omega_i$ and $I^{H_i} w : V^{h_i}(\partial\Omega_i) \to V^{H_i}(E)$ be the linear interpolation of w on E defined by the values of w on ∂E. Using some of the ideas shown in the proof of Lemma 1 (see [3] for details), it is possible to prove the following lemma:

Lemma 3. *Let us assume that the Boundary Layer Assumption holds on Ω_i and let $w \in V^{h_i}(\partial\Omega_i)$, $v_E := w - I^{H_i} w$ on E and $v_E := 0$ on $\partial\Omega_i \backslash E$. Then*

$$\bar{\rho}_i \|v_E\|^2_{H^{1/2}_{00}(E)} \preceq \left(\frac{H_i}{\eta_i} \left(1 + \log \frac{\eta_i}{h_i}\right) + \left(1 + \log \frac{\eta_i}{h_i}\right)^2 \right) |\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w|^2_{H^1_{\rho_i}(\Omega'_{i,\eta_i})}, \tag{10}$$

$$|\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} v_E|^2_{H^1_{\rho_i}(\Omega'_{i,\eta_i})} \preceq \left(1 + \log \frac{\eta_i}{h_i}\right)^2 |\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w|^2_{H^1_{\rho_i}(\Omega'_{i,\eta_i})}, \tag{11}$$

and

$$\frac{\bar{\rho}_i}{\eta_i} \|v_E\|^2_{L^2(E)} \preceq \frac{H_i^2}{\eta_i^2} \left(1 + \log \frac{\eta_i}{h_i}\right) |\mathcal{H}^{(i)}_{\rho_i,\mathcal{N}} w|_{H^1_{\rho_i}(\Omega'_{i,\eta_i})}. \tag{12}$$

When $\bar{\rho}_i \preceq \rho_i(x)$ (Inclusion Hard type) on Ω_i, then

$$\frac{\bar{\rho}_i}{\eta_i} \|v_E\|^2_{L^2(E)} \preceq \frac{H_i}{\eta_i} \left(1 + \log \frac{\eta_i}{h_i}\right) |\mathcal{H}^{(i)}_{\rho_i} w|^2_{H^1_{\rho_i}(\Omega_i)}. \tag{13}$$

4 Dual-Primal Formulation

The discrete problem associated to (1) will be formulated below in (17) as a saddle-point problem. We follow [9] for the description of the FETI-DP method.

Let $V^{h_i}(\Omega_i)$ be the space of continuous piecewise linear functions on $\mathcal{T}^{h_i}(\Omega_i)$ which vanish on $\partial\Omega_i \cap \partial\Omega$. The associated subdomain stiffness matrices $A^{(i)}$ and the load vectors $f^{(i)}$ from the contribution of the individual elements are given by

$$v^{(i)^T} A^{(i)} u^{(i)} := a_{\rho_i}(u^{(i)}, v^{(i)}) := \int_{\Omega_i} \rho_i \, \nabla u^{(i)} \cdot \nabla v^{(i)} \, dx, \quad \forall \, u^{(i)}, v^{(i)} \in V^{h_i}(\Omega_i)$$

and

$$v^{(i)^T} f^{(i)} := \int_{\Omega_i} f v^{(i)} \, dx, \quad \forall \, v^{(i)} \in V^{h_i}(\Omega_i).$$

Here and below we use the same notation to denote both finite element functions and their vector representations. We denote by $V^h(\Omega)$ the product space of the $V^{h_i}(\Omega_i)$ and represent a vector (or function) $u \in V^h(\Omega)$ as $u = \{u^{(i)}\}_{i=1}^N$ where $u^{(i)} \in V^{h_i}(\Omega_i)$.

Let the interface $\Gamma := (\cup_{i=1}^N \partial\Omega_i) \backslash \partial\Omega$ be the union of interior edges and vertices. The nodes of an edge are shared by exactly two subdomains, and the edges

are open subsets of Γ. The vertices are endpoints of the edges. For each subdomain $\overline{\Omega}_i$, let us partition the vector $u^{(i)}$ into a vector of primal variables $u_\Pi^{(i)}$ and a vector of nonprimal variables $u_\Sigma^{(i)}$. We choose only vertices as primal nodes since we are considering only two dimensional problems. Let us partition the nonprimal variables $u_\Sigma^{(i)}$ into a vector of interior variables $u_I^{(i)}$ and a vector of edge variables $u_\triangle^{(i)}$. We will enforce continuity of the solution in the primal unknowns of $u_\Pi^{(i)}$ by making them global; we subassemble the subdomain stiffness matrix $A^{(i)}$ with respect to this set of variables and denote the resulting matrix by $\tilde{A}$. For the remaining interfaces variables, i.e., the edge variables $u_\triangle := \{u_\triangle^{(i)}\}_{i=1}^N$, we will introduce Lagrange multipliers to enforce continuity. We also refer to the edge variables as dual variables.

Here we include more details: we partition the stiffness matrices according to the different sets of unknowns and obtain

$$A^{(i)} = \begin{bmatrix} A_{\Sigma\Sigma}^{(i)} & A_{\Pi\Sigma}^{(i)\,T} \\ A_{\Pi\Sigma}^{(i)} & A_{\Pi\Pi}^{(i)} \end{bmatrix}, \quad A_{\Sigma\Sigma}^{(i)} = \begin{bmatrix} A_{II}^{(i)} & A_{\triangle I}^{(i)\,T} \\ A_{\triangle I}^{(i)} & A_{\triangle\triangle}^{(i)} \end{bmatrix}, \tag{14}$$

and

$$f^{(i)} = [f_\Sigma^{(i)\,T} f_\Pi^{(i)\,T}]^T, \quad f_\Sigma^{(i)} = [f_I^{(i)} f_\triangle^{(i)}]^T.$$

Next we define the block diagonal matrices

$$A_{\Sigma\Sigma} = \text{diag}_{i=1}^N(A_{\Sigma\Sigma}^{(i)}), \quad A_{\Pi\Sigma} = \text{diag}_{i=1}^N(A_{\Pi\Sigma}^{(i)}), \quad A_{\Pi\Pi} = \text{diag}_{i=1}^N(A_{\Pi\Pi}^{(i)}),$$

and load vectors

$$f_\Sigma = \{f_\Sigma^{(i)}\}_{i=1}^N, \quad f_\Pi = \{f_\Pi^{(i)}\}_{i=1}^N.$$

Assembling the local subdomain matrices and load vectors with respect to the primal variables, we obtain the partially assembled global stiffness matrix $\tilde{A}$ and the load vector $\tilde{f}$,

$$\tilde{A} = \begin{bmatrix} A_{\Sigma\Sigma} & \tilde{A}_{\Pi\Sigma}^T \\ \tilde{A}_{\Pi\Sigma} & \tilde{A}_{\Pi\Pi} \end{bmatrix}, \quad \tilde{f} = \begin{bmatrix} f_\Sigma \\ \tilde{f}_\Pi \end{bmatrix}, \tag{15}$$

where a tilde refers an assembled quantity. It is easy to see that the matrix $\tilde{A}$ is positive definite.

To enforce the continuity of the dual variables $u_\triangle$, we introduce a jump matrix $B_\triangle$ with entries 0, -1 and 1 given by

$$B_\triangle = [B_\triangle^{(1)}, \cdots, B_\triangle^{(N)}], \tag{16}$$

where $B_\triangle^{(i)}$ consists of columns of $B_\triangle$ attributed to the i-th component of the dual variables. The space $\Lambda := range(B_\triangle)$ is used as the space for the Lagrange multipliers λ. The Dual-Primal saddle point problem is given by

$$\begin{bmatrix} A_{II} & A_{\triangle I}^T & \tilde{A}_{\Pi I}^T & 0 \\ A_{\triangle I} & A_{\triangle\triangle} & \tilde{A}_{\Pi\triangle}^T & B_\triangle^T \\ \tilde{A}_{\Pi I} & \tilde{A}_{\Pi\triangle} & \tilde{A}_{\Pi\Pi} & 0 \\ 0 & B_\triangle & 0 & 0 \end{bmatrix} \begin{bmatrix} u_I \\ u_\triangle \\ \tilde{u}_\Pi \\ \lambda \end{bmatrix} = \begin{bmatrix} f_I \\ f_\triangle \\ \tilde{f}_\Pi \\ \lambda \end{bmatrix} \tag{17}$$

where $A_{II} := \text{diag}_{i=1}^{N}(A_{II}^{(i)})$ and $\tilde{u}_{II}$ means the primal unknowns at the vertices of the substructures Ω_i. By eliminating $u_I := \{u_I^{(i)}\}_{i=1}^N$, $u_\triangle := \{u_\triangle^{(i)}\}_{i=1}^N$ and $\tilde{u}_{II}$ from (17), we obtain a system on the form

$$F\lambda = d \tag{18}$$

where

$$F = B_\Sigma \tilde{A}^{-1} B_\Sigma^T, \;\; d = B_\Sigma \tilde{A}^{-1}[f_\Sigma^T \; \tilde{f}_\Pi^T]^T \;\; \text{with} \;\; B_\Sigma = (0, B_\Delta).$$

5 FETI-DP Preconditioner

To define the FETI-DP preconditioner M for F, we need to introduce a scaled variant of the jump matrix $B_\triangle$, which we denote by

$$B_{D,\triangle} = [D_\triangle^{(1)} B_\triangle^{(1)}, \cdots, D_\triangle^{(N)} B_\triangle^{(N)}].$$

The diagonal scaling matrices $D_\triangle^{(i)}$ operates on the dual variables $u_\triangle^{(i)}$ and they are defined as follows. Let $\mathcal{J}_i$ be the indices of the substructures which share an edge with Ω_i. An edge shared by Ω_i and Ω_j is denoted by E_{ij}, and the set of dual nodes on $\mathcal{T}^{h_i}(\partial\Omega_i)$ on E_{ij} is denoted by $E_{ij,h}$. The diagonal matrix $D_\triangle^{(i)}$ is defined via $\delta_i^\dagger(x)$ where

$$\delta_i^\dagger(x) := \frac{\bar{\rho}_i}{\bar{\rho}_i + \bar{\rho}_j}(x) \quad x \in E_{ij,h} \;\; \text{and} \; j \in \mathcal{J}_i,$$

and let

$$P_\triangle := B_{D,\triangle}^T B_\triangle. \tag{19}$$

The FETI-DP preconditioner is defined by

$$M^{-1} = P_\triangle S_{\triangle\triangle} P_\triangle^T \quad \text{where}$$

$$S_{\triangle\triangle} := \text{diag}_{i=1}^{N} \langle S_{\triangle\triangle}^{(i)} \rangle, \;\; \langle S_{\triangle\triangle}^{(i)} w_\triangle^{(i)}, w_\triangle^{(i)} \rangle := \int_{\Omega_i} \rho_i \nabla \mathcal{H}_{\rho_i}^{(i)} w_\triangle^{(i)} \cdot \nabla \mathcal{H}_{\rho_i}^{(i)} w_\triangle^{(i)} \, dx,$$

where $w_\triangle^{(i)}$ is identified with a function on $V^{h_i}(\partial\Omega_i)$ which vanishes at the vertices of Ω_i. Using Lemma 2 and 3, it is possible to prove (see [3] for details) the following theorem:

Theorem 1. *Let us assume that the Boundary Layer Assumption holds for any substructures Ω_i. Then, for any $\lambda \in \Lambda$ we have:*

$$\langle M\lambda, \lambda \rangle \;\; \leq \;\; \langle F\lambda, \lambda \rangle \leq \lambda_{\max} \langle M\lambda, \lambda \rangle$$

where

$$\lambda_{\max} \preceq \max_{i=1}^{N} \frac{H_i^2}{\eta_i^2} \, (1 + \log \frac{\eta_i}{h_i}).$$

When the coefficients ρ_i, $1 \leq i \leq N$, are simultaneously of the Inclusion Hard type, or are simultaneously of the Inclusion Soft type, then:

$$\lambda_{\max} \preceq \max_{i=1}^{N} \left\{ \frac{H_i}{\eta_i} \, (1 + \log \frac{\eta_i}{h_i}) + (1 + \log \frac{\eta_i}{h_i})^2 \right\}.$$

The linear dependence result on H_i/η_i for Inclusion Soft type coefficients is the first one given in the literature. The bounds in Theorem 1 hold also for the FETI method and are sharper than $O(\frac{H_i}{\eta_i} \, (1 + \log \frac{H_i}{h_i})^2)$ obtained in [6] for Inclusion Hard type coefficients.

References

1. M. Brezina, C. Heberton, J. Mandel, and P. Vanek. An iterative method with convergence rate chosen a priori. UCD/CCM Report 140, 1999.
2. M. Dryja and M. Sarkis. Additive average Schwarz methods for discretization of elliptic problems with highly discontinuous coefficients. To appear in Comput. Methods Appl. Math., 2010.
3. M. Dryja and M. Sarkis. Boundary layer technical tools for domain decompostion methods. In preparation, 2010.
4. Y. Efendiev and J. Galvis. Domain decomposition preconditioners for multiscale problems. Texas A & M, Preprint, 2009.
5. V.J. Lent, R. Scheichl, and I. Graham. Energy minimizing coarse spaces for two-level Schwarz methods for multiscale PDEs. *Numer. Linear Algebra Appl.*, 16:775–779, 2009.
6. C. Pechstein and R. Scheichl. Analysis of FETI methods for multiscale PDEs. *Numer. Math.*, 111(2):293–333, 2008. ISSN 0029-599X. URL http://dx.doi.org/10.1007/s00211-008-0186-2.
7. C. Pechstein and R. Scheichl. Analysis of FETI methods for multiscale PDEs – Part II: Interface variation. Bath Institute for Complex Systems, University of Bath, Preprint 7/09, 2009.
8. M. Sarkis. Partition of unity coarse spaces: enhanced versions, discontinuous coefficients and applications to elasticity. In *Domain Decomposition Methods in Science and Engineering*, pp. 149–158. Natl. Auton. Univ. Mex., México, 2003.
9. A. Toselli and B.O. Widlund. *Domain Decomposition Methods – Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2005. ISBN 3-540-20696-5.

Coarse Spaces over the Ages

Jan Mandel[1] * and Bedřich Sousedík[1,2] †

[1] Department of Mathematical and Statistical Sciences,
University of Colorado Denver, Denver, CO 80217, USA
[2] Institute of Thermomechanics, Academy of Sciences of the Czech Republic, 182 00
Prague 8, Czech Republic, jan.mandel@ucdenver.edu,
bedrich.sousedik@ucdenver.edu

1 Introduction

The objective of this paper is to explain the principles of the design of a coarse space in a simplified way and by pictures. The focus is on ideas rather than on a more historically complete presentation. That can be found, e.g., in [28]. Also, space limitation does not allow us even to mention many important methods and papers that should be rightfully included.

The coarse space facilitates a global exchange of information in multigrid and domain decomposition methods for elliptic problems. This exchange is necessary, because the solution is non-local: its value at any point depends on the right-hand-side at any other point. Both multigrid and domain decomposition combine a global correction in the coarse space with local corrections, called smoothing in multigrid and subdomain solves in domain decomposition. In multigrid the coarse space is large (typically, the mesh ratio is 2 or 3 at most) and the local solvers are not very powerful (usually, relaxation). In domain decomposition, the coarse space is small (just one or a few degrees of freedom per subdomain), and the local solvers are powerful (direct solvers on subdomain). But the mathematics is more or less the same.

2 Local Nullspace and Bounded Energy Conditions

Consider the variational problem

$$u \in V : \qquad a(u,v) = f(v) \qquad \forall v \in V, \tag{1}$$

where a is symmetric positive definite and V is a finite dimensional space. Most, if not all, multigrid, domain decomposition, and substructuring methods for (1) can be

* Supported by National Science Foundation under grant DMS-0713876.
† Partially supported by National Science Foundation under grant DMS-0713876 and by the Grant Agency of the Czech Republic under grant 106/08/0403.

cast as variants of the additive Schwarz method (ASM), which is the preconditioning by the approximate solver

$$M : r \mapsto \sum_{i=0}^{N} u_i \tag{2}$$

where u_i are solutions of the subproblems

$$u_i \in V_i : \qquad a(u_i, v_i) = r(v_i) \qquad \forall v_i \in V_i \tag{3}$$

where

$$V = V_0 + V_1 + \cdots + V_N \tag{4}$$

The resulting condition number of the preconditioned problem is then bounded by nC_0, where $n \leq N + 1$ is the maximal number of the subspaces $V_0, V_1, \ldots, V_N$ that have nontrivial intersections, and C_0 is the constant from the bounded energy decomposition property

$$\forall v \in V \exists v_i \in V_i : v = \sum_{i=0}^{N} v_i, \quad \sum_{i=0}^{N} a\left(v_i, v_i\right) \leq C_0 a\left(v, v\right), \tag{5}$$

cf., [1, 9, 29].

Variants of ASM include the multiplicative use of the subspace correction in [16, 18], and the use of other forms in place of a in subproblems (3), cf., [9, 26].

Now consider V to be a space of functions on a domain Ω. The subspaces V_i range from the span of one basis vector in multigrid (for the simplest case, Jacobi iteration) to spaces of functions on large overlapping subdomains Ω_i. When the domain Ω is the union of non-overlapping subdomains Ω_j, $j = 1, \ldots, M$, the spaces V_i are defined as certain subspaces of the space $W = W_1 \times \cdots \times W_M$, where W_j is a space of functions on Ω_j. The natural splitting of the bilinear form $a\left(\cdot, \cdot\right)$ into integrals over Ω_j is then $a\left(u, v\right) = \int_\Omega \nabla u \cdot \nabla v = \sum_{j=1}^{N} a_j\left(u, v\right)$, where the local bilinear forms

$$a_j\left(u, v\right) = \int_{\Omega_j} \nabla u \cdot \nabla v \tag{6}$$

are used on W_j instead of the bilinear form $a\left(\cdot, \cdot\right)$.

The space V_0 is the coarse space, and the rest of this paper deals with its construction. It had been long understood and then formulated explicitly in [14] that for condition numbers to be independent of the number of subdomains, the coarse space needs to contain the nullspace of the local bilinear forms $a_j\left(\cdot, \cdot\right)$. For the scalar problem as in (6), this means constant functions, while for elasticity, the coarse space needs to contain the rigid body modes of every substructure. Much of the development of the coarse space in domain decomposition has been driven by the need for the coarse space to satisfy this *local nullspace condition* at the same time as the *bounded energy condition* $a\left(v_0, v_0\right) \leq C_0 a\left(v, v\right)$, required as a part of (5).

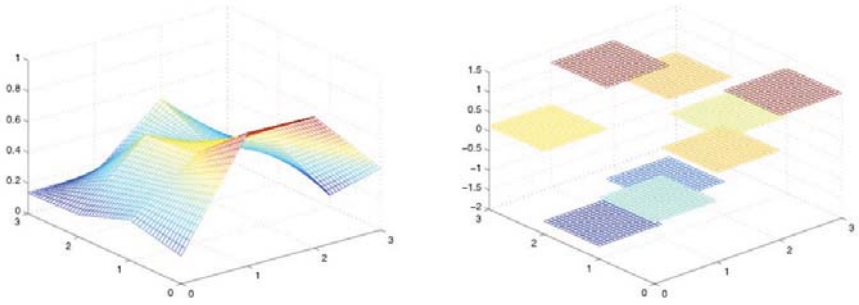


Fig. 1. *Left*: Piecewise bilinear coarse space function. *Right*: piecewise constant functions.

3 Some Early Domain Decomposition Methods

By taking v_0 in (5) first, we see that the design objective of the coarse space is that there should exist a mapping $v \in V \longmapsto v_0 \in V_0$ such that (i) the energy of v_0 is not too large, and (ii) the remainder $v - v_0$ can be decomposed in the spaces V_i, $i = 1, \ldots, N$, without increasing the energy too much. Definition of v_0 by linear or bilinear interpolation is the natural first choice (Fig. 1 left). Because of the discrete Sobolev inequality, this works fine in 2D: values of v at interpolation nodes are bounded by the energy of v up to a logarithmic factor in the mesh size h. The remainder $v - v_0$ is tied to zero by its zero values at the interpolation nodes, and it turns out it can be decomposed into v_i's with bounded energy (up to a logarithmic factor). In 3D, however, the pointwise values of v for constant energy of v can grow quickly as $h \to 0$, so interpolation can no longer be used. Overlapping methods ([8]) use decomposition into v_i's by a partition of unity on overlapping subdomains Ω_i, and they carry over to 3D; only the interpolation from the values of v needs to be replaced by a method that is energy stable in 3D, such as interpolation from averages or L^2 projection. In some non-overlapping methods, however, the functions v_i are defined in such way that they are zero at the nodes that define the values of v_0, e.g., [2]. Then a straightforward extension of the method to 3D forces v_0 to be linear interpolant from pointwise values of v. [3] resolved this problem by redefining the coarse bilinear form a_0 so that $a_0(u, u) = \sum_{i=1}^{N} \min_{c_i} \int_\Omega |\nabla u - c_i|^2$; cf., [15] for a generalization to elasticity and an algebraic explanation. The coarse space degrees of freedom are one number c_i per substructure, thus the coarse space can be thought of as piecewise constant (Fig. 1 right). Piecewise constant coarse space used with the original bilinear form $a(\cdot, \cdot)$ results in aggregation methods [27]. Reference [7] defined the interpolant by discrete harmonic functions, which have lower energy than piecewise linear functions.

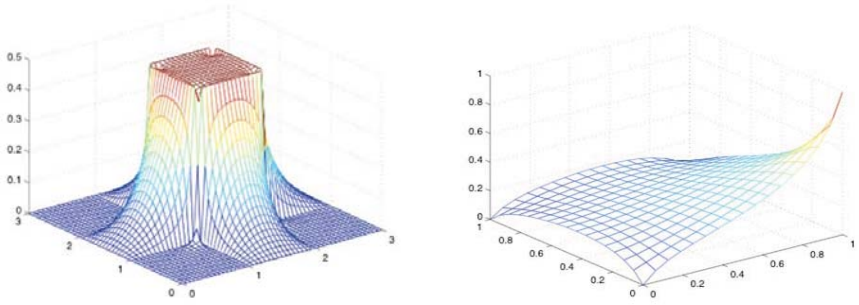


Fig. 2. *Left*: BDD coarse basis function, with support on one substructure and adjacent ones. *Right*: Coarse function on one substructure of BDD for plates, and BDDC (reproduced from [20]).

4 Balancing Domain Decomposition (BDD) and FETI

The BDD method was created by [16] by adding a special coarse space to the Neumann–Neumann (NN) method from [5]. The NN method uses the additive preconditioner with the local forms a_i from (6) and no coarse space. In the NN method, the local forms are generally singular and the local problems (3) are not consistent. The BDD preconditioner applies multiplicatively a coarse correction based on a known superspace Z_i of the local nullspace and designed so that the right-hand side in (3) is orthogonal to Z_i. Since the nullspace of a_i is contained in Z_i, (3) is now guaranteed to be consistent. The coarse space is obtained by averaging between adjacent substructures and extending the functions from the substructure boundaries in the interior with minimal energy (i.e., as discrete harmonic). A basis function of the resulting coarse space is in Fig. 2 right. Of course, for elasticity, rigid body modes are used rather than constants, giving 6 coarse degrees of freedom per substructure in 3D.

BDD is completely *algebraic*. It can be implemented only by calls to subdomain matrix-vector multiplication and by access to a basis of the local space Z_i (such as the rigid body modes written in terms of the degrees of freedom). This made possible a black-box type application of BDD to mixed finite elements in [4]: the substructure matrix-vector multiply becomes the mapping of pressure on substructure faces to the velocity in the normal direction. (Some components of other methods can be generated algebraically also; e.g., overlapping Schwarz methods are used as smoothers in adaptive algebraic multigrid in [24].)

BDD with the spaces Z_i given by constants or rigid body modes is not suitable for 4th order problems (such as plate bending), because the tearing at corners has high energy – the trace norm associated with 4th order problem is the Sobolev norm $H^{3/2}$. But empowering BDD by enriching the coarse space was envisioned already in [16], and all that was needed was to enlarge the spaces Z_i so that after the coarse

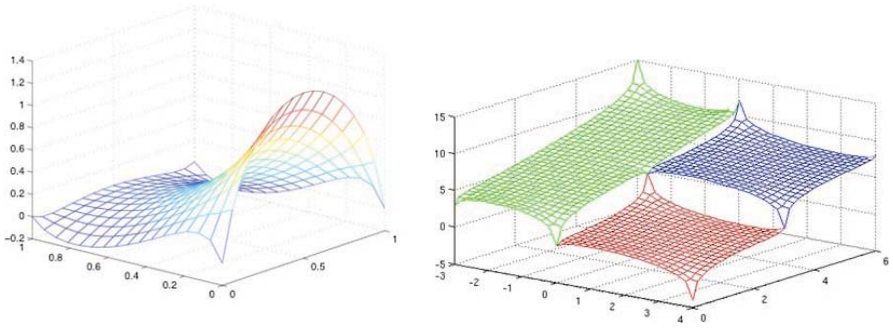


Fig. 3. *Left*: coarse function in BDDC for edge average degree of freedom on one substructure. *Right*: BDD for plates and BDDC coarse space with vertex degrees of freedom on several substructures (courtesy of Marta Čertíková and Jakub Šístek).

correction, the error is zero at corners, thus the tears across the corners do not matter. In [13], such Z_i consists of functions determined by their values at the corners of the substructure, and by having minimal energy (Fig. 2 right).

The FETI method by [11] runs in the dual space of Lagrange multipliers and it uses a coarse space constructed from the exact nullspace of the local problems (3). In the scalar case, this is the space of discontinuous piecewise constant functions (Fig. 1 right), and of piecewise rigid body modes for elasticity. Since the dual space (after elimination of the interior) is equipped with the $H^{-1/2}$ norm, jumps between subdomains do not cause a large energy increase. Like BDD, FETI is completely algebraic, which is why the two methods have become popular in practice. [23] generalized FETI to deal with 4th order problems analogously as in BDD, but the resulting method, called FETI-2, was quite complicated. Since the basic algebra of FETI relies on the exact nullspace of the local problems, the added coarse functions had to be in a new coarse space of their own, with the additional components of the coarse correction wrapped around the original FETI method. Eventually, FETI-2 was superseded by FETI-DP.

A Neumann–Neumann method, also called balancing but somewhat different from BDD, was developed in [9]. This method uses the same coarse space as BDD, but additively, and it takes care of the singularity in the local problems by adding small numbers to the diagonal. To guarantee optimal condition bounds, a modification of the form a_0 is needed. This method is not algebraic in the same sense as BDD or FETI, i.e., relying on the matrices only.

5 BDDC and FETI-DP

A satisfactory extension of FETI and BDD to 4th order problems came only with FETI-DP by [10] and BDDC by [6]. These methods are based on identical components and have the same spectrum. except possibly for the eigenvalues equal to zero

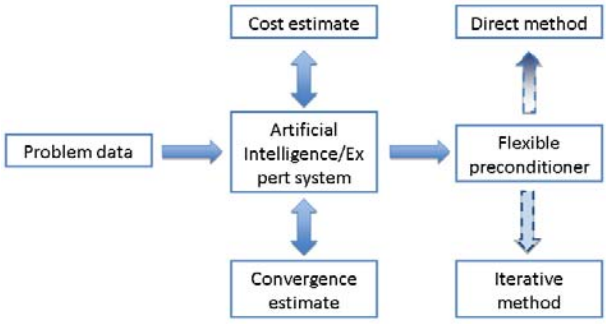


Fig. 4. Intelligent iterative method. Adapted from [17].

and one [21], so we can discuss BDDC only. The coarse space consists of functions given by their values of coarse degrees of freedom and energy minimal on every substructure independently. For coarse degrees of freedom given by values on substructure corners, this is the same coarse space as in BDD for plates in [13] (Fig. 2 right, Fig. 3 right), and the substructure spaces W_i are also the same. The new feature of BDDC is that the coarse correction is additive, not multiplicative, resulting in a sparser coarse matrix [20]. In 3D, FETI-DP and BDDC require additional degrees of freedom for optimal convergence, namely averages over faces or edges [10, 12], cf., Fig. 3 left for a visualization in 2D.

6 Adaptive Methods by Enriching the Coarse Space

Enlarging the coarse space is a powerful but expensive tool. When the coarse space is the whole space, domain decomposition turns into a direct solver. So, adding suitable functions to the coarse space adaptively can yield a robust method, which is fast on easy problems, but does not fail on hard ones (Fig. 4). In [19], the coarse space in the p-version finite element method consists of linear functions when all is good, quadratic functions when things get worse, all function in one direction in the case of strong anisotropy, up to all functions when the heuristic gives up. In [24], a similar methodology was applied in algebraic multigrid. In [22] and in the companion paper [25] in this volume, the coarse space in BDDC is enriched by adaptively selected linear combinations of basis functions on substructure faces.

References

1. P.E. Bjørstad and J. Mandel. On the spectra of sums of orthogonal projections with applications to parallel computing. *BIT*, 31(1):76–88, 1991.
2. J.H. Bramble, J.E. Pasciak, and A.H. Schatz. The construction of preconditioners for elliptic problems by substructuring. I. *Math. Comput.*, 47(175):103–134, 1986.

3. J.H. Bramble, J.E. Pasciak, and A.H. Schatz. The construction of preconditioners for elliptic problems by substructuring. IV. *Math. Comput.*, 53(187):1–24, 1989.
4. L.C. Cowsar, J. Mandel, and M.F. Wheeler. Balancing domain decomposition for mixed finite elements. *Math. Comput.*, 64(211):989–1015, 1995.
5. Y.-H. De Roeck and P. Le Tallec. Analysis and test of a local domain-decomposition preconditioner. In *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations (Moscow, 1990)*, pp. 112–128, SIAM, Philadelphia, PA, 1991.
6. C.R. Dohrmann. A preconditioner for substructuring based on constrained energy minimization. *SIAM J. Sci. Comput.*, 25(1):246–258, 2003.
7. M. Dryja. A method of domain decomposition for 3-D finite element problems. In *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pp. 43–61, SIAM, Philadelphia, PA, 1988.
8. M. Dryja and O.B. Widlund. Domain decomposition algorithms with small overlap. *SIAM J. Sci. Comput.*, 15(3):604–620, 1994. ISSN 1064-8275.
9. M. Dryja and O.B. Widlund. Schwarz methods of Neumann–Neumann type for three-dimensional elliptic finite element problems. *Commun. Pure Appl. Math.*, 48(2):121–155, 1995.
10. C. Farhat, M. Lesoinne, and K. Pierson. A scalable dual-primal domain decomposition method. *Numer. Linear Algebra Appl.*, 7:687–714, 2000.
11. C. Farhat and F.-X. Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *Int. J. Numer. Methods Eng.*, 32:1205–1227, 1991.
12. A. Klawonn, O.B. Widlund, and M. Dryja. Dual-primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients. *SIAM J. Numer. Anal.*, 40(1):159–179, 2002. ISSN 1095-7170.
13. P. Le Tallec, J. Mandel, and M. Vidrascu. A Neumann–Neumann domain decomposition algorithm for solving plate and shell problems. *SIAM J. Numer. Anal.*, 35(2):836–867 (electronic), 1998.
14. J. Mandel. Iterative solvers by substructuring for the p-version finite element method. *Comput. Methods Appl. Mech. Eng.*, 80(1–3):117–128, 1990.
15. J. Mandel. Two-level domain decomposition preconditioning for the p-version finite element method in three dimensions. *Int. J. Numer. Methods Eng.*, 29(5):1095–1108, 1990.
16. J. Mandel. Balancing domain decomposition. *Commun. Numer. Methods Eng.*, 9(3): 233–241, 1993.
17. J. Mandel. Intelligent block iterative methods. In J. Robinson, editor, *FEM Today and the Future*, pp. 471–477. Robinson and Associates, Okehampton, 1993. Proceedings of the Seventh World Congress on Finite Elements, Monte Carlo, November 1993.
18. J. Mandel. Hybrid domain decomposition with unstructured subdomains. In *Proceedings of the 6th International Symposium on Domain Decomposition Methods, Como, Italy, 1992*, volume 157 of *Contemporary Mathematics*, pp. 103–112. AMS, Providence, RI, 1994.
19. J. Mandel. Iterative methods for p-version finite elements: preconditioning thin solids. *Comput. Methods Appl. Mech. Eng.*, 133(3-4):247–257, 1996.
20. J. Mandel and C.R. Dohrmann. Convergence of a balancing domain decomposition by constraints and energy minimization. *Numer. Linear Algebra Appl.*, 10(7):639–659, 2003.
21. J. Mandel, C.R. Dohrmann, and R. Tezaur. An algebraic theory for primal and dual substructuring methods by constraints. *Appl. Numer. Math.*, 54(2):167–193, 2005.

22. J. Mandel and B. Sousedík. Adaptive selection of face coarse degrees of freedom in the BDDC and the FETI-DP iterative substructuring methods. *Comput. Methods Appl. Mech. Eng.*, 196(8):1389–1399, 2007.
23. J. Mandel, R. Tezaur, and C. Farhat. A scalable substructuring method by Lagrange multipliers for plate bending problems. *SIAM J. Numer. Anal.*, 36(5):1370–1391, 1999.
24. G. Poole, Y.-C. Liu, and J. Mandel. Advancing analysis capabilities in ANSYS through solver technology. *Electron. Trans. Numer. Anal.*, 15:106–121 (electronic), 2003.
25. H. Yanping, R. Kornhuber, O. Widlund, and J. Xu (eds.). Domain Decomposition in Science and Engineering XIX, Springer Verlag, Berlin Heidelberg, p. 213–228, 2011.
26. A. Toselli and O. Widlund. *Domain decomposition methods—algorithms and theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2005.
27. P. Vaněk, J. Mandel, and M. Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56(3):179–196, 1996.
28. O. Widlund. The development of coarse spaces for domain decomposition algorithms. In *Domain Decomposition methods in science and engineering XVIII*, volume 70 of *Lecture Notes in Computational Science and Engineering*, pp. 241–248. Springer, Heidelberg, 2009.
29. O.B. Widlund. Iterative substructuring methods: algorithms and theory for elliptic problems in the plane. In *First International Symposium on Domain Decomposition Methods for Partial Differential Equations (Paris, 1987)*, pp. 113–128. SIAM, Philadelphia, PA, 1988.

FETI-DP for Stokes-Mortar-Darcy Systems

Juan Galvis[1] and Marcus Sarkis[2,3]

[1] Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA
[2] Instituto Nacional de Matemática Pura e Aplicada, Rio de Janeiro 22460-320, Brazil
[3] Department of Mathematical Sciences, Worcester Polytechnic Institute, Worcester, MA 01609, USA

1 Introduction and Problem Setting

We consider the coupling across an interface of a fluid flow and a porous media flow. The differential equations involve Stokes equations in the fluid region, Darcy equations in the porous region, plus a coupling through an interface with Beaver-Joseph-Saffman transmission conditions, see [1, 2, 6, 8]. The discretization consists of $P2$-$P0$ finite elements in the fluid region, the lowest order triangular Raviart-Thomas finite elements in the porous region, and the mortar piecewise constant Lagrange multipliers on the interface. Due to the small values of the permeability parameter κ of the porous medium, the resulting discrete symmetric saddle point system is very ill conditioned. Preconditioning is needed in order to efficiently solve the resulting discrete system. The purpose of this work is to present some preliminary results on the extension of the modular FETI type preconditioner proposed in [5, 7] to the multidomain FETI-DP case.

Let $\Omega^f,\ \Omega^p \subset \mathbb{R}^n$ be polyhedral subdomains, define $\Omega = \text{int}(\overline{\Omega}^f \cup \overline{\Omega}^p)$ and $\Gamma = \partial\Omega^f \cap \partial\Omega^p$, with outward unit normal vectors $\boldsymbol{\eta}^i$ on $\partial\Omega^i$, $i = f, p$. The tangent vectors on Γ are denoted by $\boldsymbol{\tau}_1$ ($n = 2$), or $\boldsymbol{\tau}_l$, $l = 1, 2$ ($n = 3$). The exterior boundaries are $\Sigma^i := \partial\Omega^i \setminus \Gamma$, $i = f, p$. Fluid velocities are denoted by $\boldsymbol{u}^i : \Omega^i \to \mathbb{R}^n$, $i = f, p$, and pressures by $p^i : \Omega^i \to \mathbb{R}$, $i = f, p$.

We consider Stokes equations in the fluid region Ω^f and Darcy equations for the filtration velocity in the porous medium Ω^p.

$$\begin{array}{cc} \text{Stokes equations} & \text{Darcy equations} \\ \left\{\begin{array}{rl} -\nabla\cdot T(\boldsymbol{u}^f, p^f) = \boldsymbol{f}^f & \text{in } \Omega^f \\ \nabla\cdot \boldsymbol{u}^f = g^f & \text{in } \Omega^f \\ \boldsymbol{u}^f = \boldsymbol{h}^f & \text{on } \Sigma^f \end{array}\right. & \left\{\begin{array}{rl} \boldsymbol{u}^p = -\frac{\kappa}{\nu}\nabla p^p & \text{in } \Omega^p \\ \nabla\cdot \boldsymbol{u}^p = g^p & \text{in } \Omega^p \\ \boldsymbol{u}^p\cdot\boldsymbol{\eta}^p = h^p & \text{on } \Sigma^p. \end{array}\right. \end{array} \tag{1}$$

Here $T(\boldsymbol{v}, p) := -pI + 2\nu\boldsymbol{D}\boldsymbol{v}$, where ν is the fluid viscosity, $\boldsymbol{D}\boldsymbol{v} := \frac{1}{2}(\nabla\boldsymbol{v} + \nabla\boldsymbol{v}^T)$ is the linearized strain tensor and κ denotes the rock permeability. We assume that

κ is a real positive constant. We impose the following interface matching conditions across Γ (see [1, 2, 6, 8] and references therein):

(i) Conservation of mass across Γ: $\boldsymbol{u}^f \cdot \boldsymbol{\eta}^f + \boldsymbol{u}^p \cdot \boldsymbol{\eta}^p = 0$ on Γ.
(ii) Balance of normal forces across Γ: $p^f - 2\nu \boldsymbol{\eta}^{fT} \boldsymbol{D}(\boldsymbol{u}^f)\boldsymbol{\eta}^f = p^p$ on Γ.
(iii) Beavers-Joseph-Saffman condition: $\boldsymbol{u}^f \cdot \boldsymbol{\tau}_l = -\frac{\sqrt{\kappa}}{\alpha^f} 2\boldsymbol{\eta}^{fT} \boldsymbol{D}(\boldsymbol{u}^f)\boldsymbol{\tau}_l, l = 1, \cdots, n-1$ on Γ.

We require that $\langle g^f, 1\rangle_{\Omega^f} + \langle g^p, 1\rangle_{\Omega^p} - \langle \boldsymbol{h}^f \cdot \boldsymbol{\eta}^f, 1\rangle_{\Sigma^f} - \langle h^p, 1\rangle_{\Sigma^p} = 0$ which is the compatibility condition (see [6]).

2 Weak Formulation

In this section we present the weak version of the coupled system of partial differential equations introduced above. Without loss of generality, we consider $\boldsymbol{h}^f = \boldsymbol{0}$, $g^f = 0$, $h^p = 0$ and $g^p = 0$ in (1); see [6]. The weak problem is formulated as: *Find* $(\boldsymbol{u}, p, \lambda) \in \boldsymbol{X} \times M_0 \times \Lambda$ *such that for all* $(\boldsymbol{v}, q, \mu) \in \boldsymbol{X} \times M_0 \times \Lambda$ *we have*

$$\begin{cases} a(\boldsymbol{u}, \boldsymbol{v}) + b(\boldsymbol{v}, p) + b_\Gamma(\boldsymbol{v}, \lambda) = f(\boldsymbol{v}) \\ b(\boldsymbol{u}, q) = 0 \\ b_\Gamma(\boldsymbol{u}, \mu) = 0, \end{cases} \tag{2}$$

where $\boldsymbol{X} = \boldsymbol{X}^f \times \boldsymbol{X}^p := H_0^1(\Omega^f, \Sigma^f)^n \times \boldsymbol{H}_0(\text{div}, \Omega^p, \Sigma^p)$ and M_0 is the subset of $M := M^f \times M^p := L^2(\Omega^f) \times L^2(\Omega^p) \equiv L^2(\Omega)$ of pressures with a zero average value in Ω. Here $H_0^1(\Omega^f, \Sigma^f)$ denotes the subspace of $H^1(\Omega^f)$ of functions that vanish on Σ^f. The space $\boldsymbol{H}_0(\text{div}, \Omega^p, \Sigma^p)$ consists of vector functions in $\boldsymbol{H}(\text{div}, \Omega^p)$ with zero normal trace on Σ^p, where $\boldsymbol{H}(\text{div}, \Omega^p) := \left\{\boldsymbol{v} \in L^2(\Omega^p)^n : \text{div}\, \boldsymbol{v} \in L^2(\Omega^p)\right\}$. For the Lagrange multiplier space we consider $\Lambda := H^{1/2}(\Gamma)$. See [6, 8] for well posedness results. The global bilinear forms are given by

$$a(\boldsymbol{u}, \boldsymbol{v}) := a^f_{\alpha^f}(\boldsymbol{u}^f, \boldsymbol{v}^f) + a^p(\boldsymbol{u}^p, \boldsymbol{v}^p) \quad \text{and} \quad b(\boldsymbol{v}, p) := b^f(\boldsymbol{v}^f, p^f) + b^p(\boldsymbol{v}^p, p^p),$$

with local forms $a^f_{\alpha^f}, b^f$ and b^p defined for $\boldsymbol{u}^f, \boldsymbol{v}^i \in \boldsymbol{X}^i$, $p^i, q^i \in M^i$ by

$$a^f_{\alpha^f}(\boldsymbol{u}^f, \boldsymbol{v}^f) := 2\nu(\boldsymbol{D}\boldsymbol{u}^f, \boldsymbol{D}\boldsymbol{v}^f)_{\Omega^f} + \sum_{\ell=1}^{n-1} \frac{\nu\alpha^f}{\sqrt{\kappa}} \langle \boldsymbol{u}^f \cdot \boldsymbol{\tau}_\ell, \boldsymbol{v}^f \cdot \boldsymbol{\tau}_\ell\rangle_\Gamma, \tag{3}$$

$$a^p(\boldsymbol{u}^p, \boldsymbol{v}^p) := (\frac{\nu}{\kappa}\boldsymbol{u}^p, \boldsymbol{v}^p)_{\Omega^p}, \tag{4}$$

$$b^f(\boldsymbol{v}^f, q^f) := -(q^f, \nabla \cdot \boldsymbol{v}^f)_{\Omega^f}, \quad \text{and} \quad b^p(\boldsymbol{v}^p, p^p) := -(p^p, \nabla \cdot \boldsymbol{v}^p)_{\Omega^p}. \tag{5}$$

The weak conservation of mass bilinear form is defined by

$$b_\Gamma(\boldsymbol{v},\mu) := \langle \boldsymbol{v}^f \cdot \boldsymbol{\eta}^f, \mu\rangle_\Gamma + \langle \boldsymbol{v}^p \cdot \boldsymbol{\eta}^p, \mu\rangle_\Gamma, \quad \boldsymbol{v} = (\boldsymbol{v}^f, \boldsymbol{v}^p) \in \boldsymbol{X}, \mu \in \Lambda. \tag{6}$$

The second duality pairing of (6) is interpreted as $\langle \boldsymbol{v}^p \cdot \boldsymbol{\eta}^p, E_{\boldsymbol{\eta}^p}(\mu)\rangle_{\partial\Omega^p}$. Here $E_{\boldsymbol{\eta}^p}$ is any continuous lifting operator from $H^{1/2}(\Gamma)$ to $H^{1/2}(\partial\Omega^p)$; recall that $\Gamma \subset \partial\Omega^p$ and that $\boldsymbol{v} \in \boldsymbol{H}_0(\mathrm{div}, \Omega^p, \Sigma^p)$, see [6]. The functional f in the right-hand side of (2) is defined by $f(\boldsymbol{v}) := f^f(\boldsymbol{v}^f) + f^p(\boldsymbol{v}^p)$, for all $\boldsymbol{v} = (\boldsymbol{v}^f, \boldsymbol{v}^p) \in \boldsymbol{X}$, where $f^i(\boldsymbol{v}^i) := (\boldsymbol{f}^i, \boldsymbol{v}^i)_{L^2(\Omega^i)}$ for $i = f, p$.

The bilinear forms $a^f_{\alpha^f}, b^f$ are associated to the Stokes equations, and the bilinear forms a^p, b^p to the Darcy law. The bilinear form $a^f_{\alpha^f}$ includes interface matching conditions $1.b$ and $1.c$ above. The bilinear form b_Γ is used to impose the weak version of the interface matching condition $1.a$ above.

3 Discretization and Decomposition

From now on we consider only the two-dimensional case. The ideas developed below can be extended to the case of three-dimensional subdomains. We assume that Ω^i, $i = f, p$, are polygonal subdomains. For the fluid region, let $\boldsymbol{X}^{h,f}$ and $M^{h,f}$ be $P2/P0$ triangular finite elements. For the porous region, let $\boldsymbol{X}^{h,p}$ and $M^{h,p}$ be the lowest order Raviart-Thomas finite elements based on triangles. Define $\boldsymbol{X}_h := \boldsymbol{X}^{h,f} \times \boldsymbol{X}^{h,p} \subset \boldsymbol{X}$ and $M_h := M^{h,f} \times M^{h,p} \subset M_0$. We assume that the boundary conditions are included in the definition of the finite element spaces, i.e., for $\boldsymbol{v}^f \in \boldsymbol{X}^{h,f}$ we have $\boldsymbol{v}^f = \boldsymbol{0}$ on the exterior fluid boundary Σ^f and for $\boldsymbol{v}^p \in \boldsymbol{X}^{h,p}$ we have that $\boldsymbol{v}^p_h \cdot \boldsymbol{\eta}^p = 0$ on the porous exterior boundary Σ^p.

With the discretization chosen above, we obtain the following symmetric saddle point linear system

$$\left[\begin{array}{ccc|c} K^f & 0 & M^{fT} & \\ 0 & K^p & M^{pT} & \\ M^f & M^p & 0 & \end{array}\right] \left[\begin{array}{c} \boldsymbol{u}^f \\ p^f \\ \boldsymbol{u}^p \\ p^p \\ \lambda \end{array}\right] = \left[\begin{array}{cccc|c} A^f & B^{fT} & 0 & 0 & C^{fT} \\ B^f & 0 & 0 & 0 & 0 \\ 0 & 0 & A^p & B^{pT} & -C^{pT} \\ 0 & 0 & B^p & 0 & 0 \\ C^f & 0 & -C^p & 0 & 0 \end{array}\right] \left[\begin{array}{c} \boldsymbol{u}^f \\ p^f \\ \boldsymbol{u}^p \\ p^p \\ \lambda \end{array}\right] = \left[\begin{array}{c} \boldsymbol{f}^f \\ g^f \\ \boldsymbol{f}^p \\ g^p \\ 0 \end{array}\right] \tag{7}$$

with matrices A^i, B^i, C^i defined by

$$a^i(\boldsymbol{u}^i, \boldsymbol{v}^i) = \boldsymbol{v}^{iT} A^i \boldsymbol{u}^i, \qquad b^i(\boldsymbol{u}^i, q^i) = q^{iT} B^i \boldsymbol{u}^i, \quad (\boldsymbol{u}^i \cdot \boldsymbol{\eta}^f, \mu)_\Gamma = \mu^T C^i \boldsymbol{u}^i,$$

and vectors $\boldsymbol{f}^i, g^i$ given by $f^i(\boldsymbol{v}^i) = \boldsymbol{v}^{iT} \boldsymbol{f}^i$, $g^i(q^i) = q^{iT} g^i$, $i = f, p$. Matrix A^f corresponds to ν times the discrete version of the linearized stress tensor on Ω^f. Note that in the case $\alpha^f > 0$, the bilinear form $a^f_{\alpha^f}$ in (3) includes a boundary term. The matrix A^p corresponds to ν/κ times a discrete L^2-norm on Ω^p. Matrix $-B^i$ is the discrete divergence in Ω^i, $i = f, p$, and matrices C^f and C^p correspond to the matrix form of the discrete conservation of mass on Γ. Note that ν can be viewed as

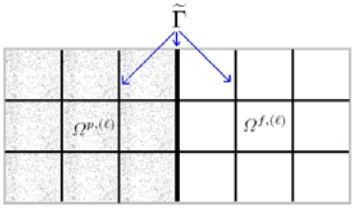


Fig. 1. Global interface $\widetilde{\Gamma}$ that includes all local interfaces and the Stokes/Darcy interface Γ.

a scaling factor since it appears in both matrices A^f and A^p, therefore, ν plays no role for the preconditioning.

Let $\{\Omega^{i,(\ell)}\}_{\ell=1}^{N^i}$ be geometrically conforming substructures of Ω^i, $i = f, p$. We also assume that $\{\Omega^{f,(\ell)}\}_{\ell=1}^{N^f} \cup \{\Omega^{p,(\ell)}\}_{\ell=1}^{N^p}$ forms a geometrically conforming decomposition of Ω, hence, the two decompositions are aligned on the Stokes/Darcy interface Γ, see Fig. 1. We define the local inner interfaces as $\Gamma^{i,(\ell)} = \partial\Omega^{i,(\ell)} \setminus \partial\Omega^i$, $\ell = 1, \ldots, N^i$, $i = f, p$. We also define the global interface

$$\widetilde{\Gamma} = \left(\bigcup_{\ell=1}^{N^f} \Gamma^{f,(\ell)}\right) \cup \left(\bigcup_{\ell=1}^{N^p} \Gamma^{p,(\ell)}\right) \cup \Gamma \equiv (\Gamma^f) \cup (\Gamma^p) \cup \Gamma.$$

In the Stokes region $\Omega^{f,(\ell)}$, we consider the following partition of the degrees of freedom,

$$\begin{bmatrix} \boldsymbol{u}_I^{f,(\ell)} \\ p_I^{f,(\ell)} \\ u_{\widetilde{\Gamma}}^{f,(\ell)} \\ \bar{p}^{f,(\ell)} \end{bmatrix} \begin{array}{l} \text{Interior velocities in } \Omega^{f,(\ell)} + \text{tangential velocities on } \partial\Omega^{f,(\ell)} \setminus \Gamma, \\ \text{Interior pressures with zero average in } \Omega^{f,(\ell)}, \\ \text{Interface velocities on } \Gamma^{f,(\ell)} + \text{normal velocities on } \partial\Omega^{f,(\ell)} \cap \Gamma, \\ \text{Constant pressure in } \Omega^{f,(\ell)}. \end{array}$$

Analogously, in the Darcy region $\Omega^{p,(\ell)}$ we use,

$$\begin{bmatrix} \boldsymbol{u}_I^{p,(\ell)} \\ p_I^{p,(\ell)} \\ u_{\widetilde{\Gamma}}^{p,(\ell)} \\ \bar{p}^{p,(\ell)} \end{bmatrix} \begin{array}{l} \text{Interior velocities in } \Omega^{p,(\ell)}, \\ \text{Interior pressures with zero average in } \Omega^{p,(\ell)}, \\ \text{Normal velocities on } \Gamma^{p,(\ell)} + \text{normal velocities on } \partial\Omega^{p,(\ell)} \cap \Gamma, \\ \text{Constant pressure in } \Omega^{p,(\ell)}. \end{array}$$

Then, for $i = f, p$, we have the block structure:

$$A^i = \begin{bmatrix} A_{II}^i & A_{\Gamma I}^{iT} \\ A_{\Gamma I}^i & A_{\Gamma\Gamma}^i \end{bmatrix}, \quad B^i = \begin{bmatrix} B_{II}^i & B_{\Gamma I}^{iT} \\ 0 & \widetilde{B}^{iT} \end{bmatrix} \text{ and } C^i = \begin{bmatrix} 0\,0\,\widetilde{C}^i\,0 \end{bmatrix}.$$

The $(2,1)$ entry of B^i corresponds to integrating an interior velocity against a constant pressure, therefore, it vanishes due to the divergence theorem.

Following [9] we choose the following matrix representation in each subdomain $\Omega^{i,(\ell)}$, $i = f, p$,

$$K^{i,(\ell)} = \left[\begin{array}{cc|cc} A_{II}^{i,(\ell)} & B_{II}^{i,(\ell)T} & A_{\Gamma I}^{i,(\ell)T} & 0 \\ B_{II}^{i} & 0 & B_{I\Gamma}^{i,(\ell)} & 0 \\ \hline A_{\Gamma I}^{i,(\ell)} & B_{I\Gamma}^{i,(\ell)T} & A_{\Gamma\Gamma}^{i,(\ell)} & \bar{B}^{i,(\ell)T} \\ 0 & 0 & \bar{B}^{i,(\ell)} & 0 \end{array}\right] = \left[\begin{array}{c|c} K_{II}^{i,(\ell)} & K_{\Gamma I}^{i,(\ell)T} \\ \hline K_{\Gamma I}^{i,(\ell)} & K_{\Gamma\Gamma}^{i,(\ell)} \end{array}\right]. \tag{8}$$

4 Dual Formulation

In order to simplify the notation and since there is no danger of confusion, we will denote the finite element functions and the corresponding vector representation by the same symbols. Let $\boldsymbol{X}^{i,(\ell)}$, $M^{i,(\ell)}$ be the finite element spaces $\boldsymbol{X}_h$ and M_h restricted to subdomain $\Omega^{i,(\ell)}$, $i = f, p$, $\ell = 1, \ldots, N^i$. Define the product spaces,

$$\boldsymbol{W} = \boldsymbol{W}^f \otimes \boldsymbol{W}^p = \bigotimes_i \bigotimes_\ell \boldsymbol{X}^{i,(\ell)}$$

and $Q = M^f \otimes M^p = \bigotimes_i \bigotimes_\ell M^{i,(\ell)}$. Functions in $\boldsymbol{W}$ do not satisfy any continuity requirement at the subdomain corners or edges. In particular they do not satisfy continuity on Stokes/Stokes edges, or continuity of normal component on Darcy/Darcy edges, neither discrete continuity of normal fluxes on Stokes/Darcy edges. The linear operator $K = \text{diag}(K^f, K^p)$ in (7) defined on the pair of spaces $(\boldsymbol{X}_h, M_h)$, can be extended to the pair $(\boldsymbol{W}, Q)$ defined above. The resulting matrix will be a block diagonal.

Primal degrees of freedom and definition of $\widetilde{\boldsymbol{W}}$: we now introduce our primal degrees of freedom, as is usual in the constructions of FETI-DP [4] and BDDC methods [3]. The primal degrees of freedom are selected accordingly for Stokes and Darcy substructures. On the fluid side, the primal degrees of freedom are given by the fluid velocity field at the substructure corners and by the mean value of both components over each Stokes/Stokes edge on Γ^f; see [9, 10]. For the porous side, the primal degrees of freedom consist of the mean value of the normal flux on each Darcy/Darcy edge on Γ^p; see [11]. For the Stokes/Darcy interface Γ, the primal degrees of freedom consist of the mean value of the normal (either Stokes or Darcy velocity) flux on each Stokes/Darcy edge on Γ; see [7]. The $\widetilde{\boldsymbol{W}}$ is the subspace of $\boldsymbol{W}$ made of functions that are continuous on the primal degrees of freedom described above.

Once the linear operator $K = \text{diag}(K^f, K^p)$ in (7) is extended to $(\boldsymbol{W}, Q)$, it can be restricted to an operator $\widetilde{K}$ acting on $(\widetilde{\boldsymbol{W}}, Q)$. The matrix form of $\widetilde{K}$ is no longer block diagonal but it will have a block structure with small interaction between blocks associated to different subdomains; see [9]. In the FETI-DP method,

we will need the inverse action of $\widetilde{K}$. This inverse action can be obtained by solving a small coarse problem and a (either Darcy or Stokes) local problems for each subdomains.

Functions in $\widetilde{\boldsymbol{W}}$ do not satisfy the dual continuity requirements on $\widetilde{\Gamma}$. The dual continuity requirements can be enforced using additional FETI-Lagrange multipliers μ on $\widetilde{\Gamma} \setminus \Gamma$ and the Stokes-Mortar-Darcy-Lagrange multipliers on Γ just as before. We obtain the linear system

$$\begin{bmatrix} \widetilde{K} & \widetilde{B}^T \\ \widetilde{B} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{w} \\ \widetilde{\lambda} \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix} \tag{9}$$

where the vector $\widetilde{\lambda}$ includes all Lagrange multiplier degrees of freedom. The matrix $\widetilde{B}$ has entries $+1, -1, 0$ for the degrees of freedom associated Γ^f and Γ^p. On the Stokes/Darcy interface Γ, we ensure that the flux continuity across Stokes/Darcy edges on Γ coincides with the last equation of (7). For that, we use the same Lagrange multipliers, up to the constant functions, as for the Stokes-Mortar-Darcy system (7). We now eliminate all degrees of freedom but the ones associated to the Lagrange multipliers to obtain a dual formulation,

$$\widetilde{B}\widetilde{K}^{-1}\widetilde{B}^T\widetilde{\lambda} = \widetilde{F}\widetilde{\lambda} = b = \widetilde{B}\widetilde{K}^{-1}b \tag{10}$$

where $\widetilde{\lambda} \in \text{Rank}(\widetilde{B})$. Note that applying $\widetilde{K}^{-1}$ requires the solution of a Stokes/Darcy problem with a block structure and very little coupling between blocks; see [9].

4.1 Dirichlet Preconditioner

Let us define

$$S^D_{\widetilde{\Gamma}} := \text{diag}(S^f_{\widetilde{\Gamma}}, S^p_{\widetilde{\Gamma}}) \quad \text{where} \quad S^i_{\widetilde{\Gamma}} =: \sum_{\ell=1}^{N^i} R^{i,(\ell)T} D_1^{i,(\ell)} S^{i,(\ell)}_{\widetilde{\Gamma}} D_1^{i,(\ell)} R^{i,(\ell)} \tag{11}$$

and $S^{i,(\ell)}_{\widetilde{\Gamma}}$ is defined from (8) via

$$S^{i,(\ell)} = \begin{bmatrix} S^{i,(\ell)}_{\widetilde{\Gamma}} & \bar{B}^{i,(\ell)T} \\ \bar{B}^{i,(\ell)} & 0 \end{bmatrix} := K^{i,(\ell)}_{\Gamma\Gamma} - K^{i,(\ell)}_{\Gamma I} \left(K^{i,(\ell)}_{II}\right)^{-1} K^{i,(\ell)T}_{\Gamma I},$$

$$I^D_{\widetilde{\Gamma}} := \text{diag}(I^f_{\widetilde{\Gamma}}, I^p_{\widetilde{\Gamma}}) \quad \text{where} \quad I^i_{\widetilde{\Gamma}} =: \sum_{\ell=1}^{N^i} R^{i,(\ell)T} D_2^{i,(\ell)} I^{i,(\ell)}_{\widetilde{\Gamma}} D_2^{i,(\ell)} R^{i,(\ell)} \tag{12}$$

and $I^{i,(\ell)}_{\widetilde{\Gamma}}$ is an identity matrix. We propose the following preconditioners:

$$\widetilde{B}(S^D_{\widetilde{\Gamma}} + I^D_{\widetilde{\Gamma}})\widetilde{B}^T. \tag{13}$$

In (11) we choose the diagonal matrix $D_1^{i,(\ell)}$ with entries $1/2$ on both sides of Stokes/Stokes and Darcy/Darcy edges, the value zero at the Stokes corners, and the values γ_1^f (Stokes side) and γ_1^p (Darcy side) on the Stokes/Darcy edges. In (12) we choose the diagonal matrix $D_2^{i,(\ell)}$ entries equal to γ_2^f (Stokes side) and γ_2^p (Darcy side) on the Stokes/Darcy edges, and entries equal zero elsewhere.

5 Numerical Results

In this section we present representative numerical results concerning the performance of the FETI-DP methods introduced before. We consider $\Omega^f = (1,2)\times(0,1)$ and $\Omega^p = (0,1)\times(0,1)$. We set $\mu = 1$. See [6] for examples of exact solutions and compatible divergence and boundary data. We use Conjugate Gradient (CG) and Preconditioned Conjugate Gradient (PCG) with the Dirichlet preconditioner (13) to solve the linear system (10). In our test problems we run (CG) PCG until the initial residual is reduced by a factor of 10^{-6}.

Table 1. Right: PCG iteration number for different number of subdomains. CG iteration number in parenthesis. Here $\frac{H}{h} = 4$, $H^f = H^p = H = \frac{1}{N}$, $\gamma_1^f = 0$, $\gamma_1^p = 1$, $\gamma_2^f = 0$, $\gamma_2^p = 0$. Left: $\frac{H}{h} = 8$.

$\kappa \downarrow N \rightarrow$	2×2	4×4	8×8
1	5(27)	7(57)	8(66)
10^{-2}	7(13)	8(22)	8(36)
10^{-4}	11(47)	19(52)	15(33)
10^{-6}	18(74)	34(131)	43(157)

$\kappa \downarrow N \rightarrow$	2×2	4×4	8×8
1	6(62)	9(98)	10(104)
10^{-2}	8(23)	10(40)	10(64)
10^{-4}	20(70)	20(61)	16(36)
10^{-6}	29(150)	60(259)	79(275)

Table 2. Top: PCG iteration and condition number for different number of subdomains. $\frac{H}{h} = 4$, $H^f = H^p = H = \frac{1}{N}$, $\gamma_1^f = 0$, $\gamma_1^p = 0$, $\gamma_2^f = 1$, $\gamma_2^p = 1 + H/h$. Bottom: $\frac{H}{h} = 8$

$\kappa \downarrow N \rightarrow$	2×2	4×4	8×8
1	9(4.4e+2)	15 (1.8e+3)	22 (7.0e+3)
10^{-2}	7(5.5e+0)	12 (1.9e+1)	16 (7.1e+1)
10^{-4}	7(3.2e+0)	8 (4.6e+0)	8 (4.6e+0)
10^{-6}	7(3.4e+0)	9 (5.7e+0)	10 (6.7e+0)

$\kappa \downarrow N \rightarrow$	2×2	4×4	8×8
1	18(3.2e+3)	32(1.3e+4)	40(5.2e+4)
10^{-2}	14(3.3e+1)	24(1.3e+2)	30(5.2e+2)
10^{-4}	10(8.3e+0)	12(1.3e+1)	14(1.7e+1)
10^{-6}	11(8.3e+0)	13(1.2e+1)	15(1.5e+1)

In our first experiment we fix $H/h = 4$ or $H/h = 8$ and run CG and PCG for different values of $H = H^f = H^p$ and different values of κ. See Table 1 for the

FETI-DP method with and without a preconditioner. We observe the preconditioned FETI-DP method with $\gamma_1^f = 0$, $\gamma_1^p = 1$, $\gamma_2^p = 0$ and $\gamma_2^f = 0$ is robust with respect to the number of subdomains and size of the subdomains when the κ is not very small. We repeat the experiment above with $\gamma_1^f = 0$, $\gamma_1^p = 0$, $\gamma_2^f = 0$ and $\gamma_2^p = 1 + H/h$ and present the number of iterations and estimate condition numbers in Table 2. With this choice of paramenters we obtain a robust preconditioner for κ small. Analysis of the FETI-DP methods presented here as well as the design of more sophisticated FETI-DP solvers are currently being studied by the authors.

References

1. T. Arbogast and D.S. Brunson. A computational method for approximating a Darcy-Stokes system governing a vuggy porous medium. *Comput. Geosci.*, 11(3):207–218, 2007.
2. M. Discacciati, A. Quarteroni, and A. Valli. Robin–Robin domain decomposition methods for the Stokes-Darcy coupling. *SIAM J. Numer. Anal.*, 45(3):1246–1268 (electronic), 2007.
3. C.R. Dohrmann. A preconditioner for substructuring based on constrained energy minimization. *SIAM J. Sci. Comput.*, 25(1):246–258 (electronic), 2003.
4. C. Farhat, M. Lesoinne, and K. Pierson. A scalable dual-primal domain decomposition method. *Numer. Linear Algebra Appl.*, 7(7–8):687–714, 2000. Preconditioning techniques for large sparse matrix problems in industrial applications (Minneapolis, MN, 1999).
5. J. Galvis and M. Sarkis. Balancing domain decomposition methods for mortar coupling Stokes-Darcy systems. In D. Keyes and O.B. Widlund, editors, *Domain Decomposition Methods in Science and Engineering XVI*, volume 55 of *Lecture Notes in Computational Science and Engineering*, pp. 373–380. Springer Berlin, Heidelberg, New York, 2006.
6. J. Galvis and M. Sarkis. Non-matching mortar discretization analysis for the coupling Stokes-Darcy equations. *Electron. Trans. Numer. Anal.*, 26:350–384, 2007.
7. J. Galvis and M. Sarkis. FETI and BDD preconditioners for Stokes-Mortar-Darcy systems. *Commun. Appl. Math. Comput. Sci.*, 5(1):1–30, 2010.
8. W.J. Layton, F. Schieweck, and I. Yotov. Coupling fluid flow with porous media flow. *SIAM J. Numer. Anal.*, 40(6):2195–2218 (2003), 2002.
9. J. Li. A dual-primal FETI method for incompressible Stokes equations. *Numer. Math.*, 102(2):257–275, 2005.
10. L.F. Pavarino and O.B. Widlund. Balancing Neumann–Neumann methods for incompressible Stokes equations. *Commun. Pure Appl. Math.*, 55(3):302–335, 2002.
11. X. Tu. A BDDC algorithm for a mixed formulation of flow in porous media. *Electron. Trans. Numer. Anal.*, 20:164–179, 2005.

Multigrid Methods for Elliptic Obstacle Problems on 2D Bisection Grids

Long Chen[1], Ricardo H. Nochetto[2], and Chen-Song Zhang[3]

[1] Department of Mathematics, University of California at Irvine, CA 92697, USA, chenlong@math.uci.edu
[2] Department of Mathematics, University of Maryland, College Park, MD, USA, rhn@math.umd.edu
[3] Department of Mathematics, The Pennsylvania State University, University Park, PA 16802, USA, Corresponding author. zhangcs@psu.edu

1 Introduction

In this paper, we develop and analyze an efficient multigrid method to solve the finite element systems from elliptic obstacle problems on two dimensional adaptive meshes. Adaptive finite element methods (AFEMs) based on local mesh refinement are an important and efficient approach when the solution is non-smooth. An optimality theory on AFEM for linear elliptic equations can be found in [8]. To achieve optimal complexity, an efficient solver for the discretization is indispensable.

The classical projected successive over-relaxation method by [5] converges but the convergence rate degenerates quickly as the mesh size approaches zero. To speed up the convergence, different multigrid and domain decomposition techniques have been developed, see the monograph [7] and the recent review [6]. In particular, the constraint decomposition method by [10] is proved to be convergent linearly with a rate which is *almost* robust with respect to the mesh size in $\mathbb{R}^2$; but the result is restricted to uniformly refined grids.

We shall extend the algorithm and theoretical results by [10] to an important class of adaptive grids obtained by newest vertex bisections; thereafter we call them *bisection grids* for short. This is new according to [6]: the existing work assumes quasi-uniformity of the underlying meshes. Based on a decomposition of bisection grids due to [3], we present an efficient constraint decomposition method on bisection grids and prove an almost uniform convergence

$$\mathcal{J}(u^k) - \mathcal{J}(u^*) \leq C\Big(1 - \frac{1}{1 + |\log h_{\min}|^2}\Big)^k, \tag{1}$$

where $\mathcal{J}(u) = \int_\Omega (\frac{1}{2}|\nabla u|^2 - fu)\,\mathrm{d}x$ is the objective energy functional, u^k is the k-th iteration and u^* is the exact solution of the constrained minimization problem, $h_{\min} = \min_{\tau\in\mathcal{T}} \operatorname{diam}(\tau)$ and the grid $\mathcal{T}$ is obtained by bisections from a suitable initial triangulation $\mathcal{T}_0$.

2 Constraint Decomposition Methods

The subspace correction framework [14] has been extended to nonlinear convex minimization problems [12]. This technique has also been applied to develop domain decomposition and multigrid methods for obstacle problems in [1, 11]. Furthermore, a constraint decomposition method (CDM) was introduced and proved to have a contraction factor which is almost independent of mesh size [10]. In this section, we briefly review the CDM for obstacle problems.

Let $\mathbb{V} \subset H_0^1$ be a finite dimensional Hilbert space and $\mathcal{J} : \mathbb{K} \to \mathbb{R}$ be a convex functional defined over the convex set $\mathbb{K} \subset \mathbb{V}$. We consider the energy minimization problem

$$\min_{v \in \mathbb{K}} \mathcal{J}(v). \tag{2}$$

In this paper, for simplicity, we only consider the case

$$\mathcal{J}(u) := \int_\Omega (\frac{1}{2}|\nabla u|^2 - fu)\,\mathrm{d}x \quad \text{and} \quad \mathbb{K} := \{v \in \mathbb{V} \,|\, v \geq 0\}, \tag{3}$$

where $\Omega \subset \mathbb{R}^2$ is a polygonal domain, $\mathcal{T}$ is a conforming triangulation of Ω, $\mathbb{V} = \mathbb{V}(\mathcal{T})$ is the continuous piecewise linear finite element space over $\mathcal{T}$. Let $|\!|\!|\cdot|\!|\!|$ be the norm associated to the energy $\mathcal{J}$. For our choice of $\mathcal{J}$ in (3), the energy norm is $|\!|\!|u|\!|\!| = \|\nabla u\|$. The algorithm discussed in this paper can be generalized to problems with more general energies and obstacles.

We decompose the space $\mathbb{V}$ into a sum of subspaces $\mathbb{V}_i \subset \mathbb{V}, i = 1, \ldots, m$:

$$\mathbb{V} = \mathbb{V}_1 + \cdots + \mathbb{V}_m = \textstyle\sum_{i=1}^m \mathbb{V}_i, \tag{4}$$

and further decompose the convex set $\mathbb{K}$ as follows

$$\mathbb{K} = \mathbb{K}_1 + \cdots + \mathbb{K}_m = \textstyle\sum_{i=1}^m \mathbb{K}_i \quad \text{with} \quad \mathbb{K}_i \subset \mathbb{V}_i \ (i = 1, \ldots, m), \tag{5}$$

where $\mathbb{K}_i$ are convex and closed in $\mathbb{V}_i$. Then we have the following abstract algorithm of successive subspace correction type.

Algorithm 1 (CDM) Given an initial guess $u^0 \in \mathbb{K}$.
For $k = 0, 1, \ldots,$ till convergence
  Decompose $u^k = \sum_{i=1}^m u_i$, such that $u_i \in \mathbb{K}_i$; and let $w^0 = u^k$.
  For $i = 1 : m$
    $w^i = w^{i-1} + \text{argmin}_{d_i} \left\{\mathcal{J}(w^{i-1} + d_i) \,|\, d_i \in \mathbb{V}_i \text{ and } u_i + d_i \in \mathbb{K}_i\right\}$.
  End For
  Let $u^{k+1} = w^m$.
End For

It is clear that each iteration w^i $(i = 1, \ldots, m)$ stays in the feasible set $\mathbb{K}$ due to (5). A linear convergence rate of Algorithm 1 has been established in [10] under the following assumptions:

Assumption 1 (Assumptions on Decomposition) (i) *Nonlinear Stability*: For any $u, v \in \mathbb{K}$, there exist a constant $C_1 > 0$ and decompositions $u = \sum_{i=1}^{m} u_i, v = \sum_{i=1}^{m} v_i$ with $u_i, v_i \in \mathbb{K}_i$ such that

$$\Big(\sum_{i=1}^{m} |\!|\!| u_i - v_i |\!|\!|^2 \Big)^{\frac{1}{2}} \le C_1 \, |\!|\!| u - v |\!|\!| \, ;$$

(ii) *Nonlinear Strengthened Cauchy–Schwarz*: There exists $C_2 > 0$ such that

$$\sum_{i,j=1}^{m} | \langle \mathcal{J}'(w_{ij} + v_i) - \mathcal{J}'(w_{ij}), \widetilde{v}_j \rangle | \le C_2 \Big(\sum_{i=1}^{m} |\!|\!| v_i |\!|\!|^2 \Big)^{\frac{1}{2}} \Big(\sum_{j=1}^{m} |\!|\!| \widetilde{v}_j |\!|\!|^2 \Big)^{\frac{1}{2}},$$

for any $w_{ij} \in \mathbb{V}$, $v_i \in \mathbb{V}_i$, and $\widetilde{v}_j \in \mathbb{V}_j$.

Theorem 1 (Convergence Rate of CDM) *If Assumption 1 is satisfied, then Algorithm 1 converges linearly and*

$$\frac{\mathcal{J}(w) - \mathcal{J}(u^*)}{\mathcal{J}(u) - \mathcal{J}(u^*)} \le 1 - \frac{1}{(\sqrt{1 + C_0} + \sqrt{C_0})^2}, \tag{6}$$

where u^* is the solution of (2) *and $C_0 = 2C_2 + C_1^2 C_2^2$.*

3 A Constraint Decomposition on Bisection Grids

In this section, we construct subspace decompositions of the linear finite element space $\mathbb{V}$, as well as a constraint decomposition of $\mathbb{K}$, on a bisection grid $\mathcal{T}$. Our new algorithm is based on a decomposition of bisection grids introduced in [3]; see also [13].

For each triangle $\tau \in \mathcal{T}$, we label one vertex of τ as the *newest vertex* and call it $V(\tau)$. The opposite edge of $V(\tau)$ is called the *refinement edge* and denoted by $E(\tau)$. This process is called *labeling* of $\mathcal{T}$. Given a labeled initial grid $\mathcal{T}_0$, newest vertex bisection follows two rules:

(i) a triangle (*father*) is bisected to obtain two new triangles (*children*) by connecting its newest vertex with the midpoint of its refinement edge;
(ii) the new vertex created at the midpoint of the refinement edge is labeled as the newest vertex of each child.

Therefore, refined grids $\mathcal{T}$ from a labeled initial grid $\mathcal{T}_0$ inherit labels according to the second rule and the bisection process can thus proceed. We define $\mathbb{C}(\mathcal{T}_0)$ as the set of conforming triangulations obtained from $\mathcal{T}_0$ by newest vertex bisection(s). It can be easily shown that all the descendants of a triangle in $\mathcal{T}_0$ fall into four similarity classes and hence any triangulation $\mathcal{T} \in \mathbb{C}(\mathcal{T}_0)$ is shape-regular.

Let $\mathcal{T}$ be a labeled conforming mesh. Two triangles sharing a common edge are called *neighbors* to each other. A triangle τ has at most three neighbors. The neighbor sharing the refinement edge of τ is called the *refinement neighbor* and denoted by

$F(\tau)$. Note that $F(\tau) = \varnothing$ if $E(\tau)$ is on the boundary of Ω. Although $E(\tau) \subset F(\tau)$, the refinement edge of $F(\tau)$ could be different than $E(\tau)$. An element τ is called *compatible* if $F(F(\tau)) = \tau$ or $F(\tau) = \varnothing$. We call a grid $\mathcal{T}$ *compatibly labeled* if every element in $\mathcal{T}$ is compatible and call such a labeling of $\mathcal{T}$ a *compatible labeling*.

For a compatible element τ, its refinement edge e is called a *compatible edge*, and $\omega_e = \tau \cup F(\tau)$ is called a *compatible patch*. By this definition, if e is a compatible edge, ω_e is either a pair of two triangles sharing the same refinement edge e or one triangle whose refinement edge e is on the boundary. In both cases, bisection of triangles in ω_e preserves mesh conformity; we call such a bisection a *compatible bisection*. Mathematically, we define the compatible bisection as a map $b_e : \omega_e \to \omega_p$, where ω_p consists of all triangles sharing the new point p introduced in the bisection. We then define the addition $\mathcal{T} + b_e := (\mathcal{T} \backslash \omega_e) \cup \omega_p$. For a sequence of compatible bisections $\mathcal{B} = (b_1, b_2, \cdots, b_m)$, we define

$$\mathcal{T} + \mathcal{B} := ((\mathcal{T} + b_1) + b_2) + \cdots + b_m,$$

whenever the addition is well defined.

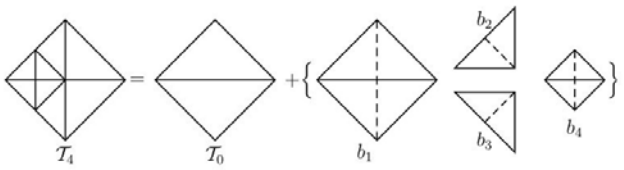


Fig. 1. A decomposition of a bisection grid.

Theorem 2 (Decomposition of Bisection Grids) *If $\mathcal{T}_0$ is conforming and compatibly labeled, then for any $\mathcal{T} \in \mathbb{C}(\mathcal{T}_0)$, there exists a compatible bisection sequence $\mathcal{B} = (b_1, b_2, \cdots, b_m)$, such that*

$$\mathcal{T} = \mathcal{T}_0 + \mathcal{B}. \tag{7}$$

Remark 1. We only give a pictorial demonstration in Fig. 1 to illustrate the decomposition. For the proof of Theorem 2, we refer to [3, 13]. A practical decomposition algorithm has been developed and implemented in [4]. □

Throughout this paper, we will assume that $\mathcal{T} \in \mathbb{C}(\mathcal{T}_0)$ has been decomposed as in (7). We denote the intermediate grids by

$$\mathcal{T}_i := ((\mathcal{T}_0 + b_1) + b_2) \cdots + b_i \qquad i = 1, \ldots, m,$$

and observe that $\mathcal{T}_i \in \mathbb{C}(\mathcal{T}_0)$. Let $\mathcal{P}(\mathcal{T}_i)$ denote the set of interior vertices of the triangulation $\mathcal{T}_i$. Denote by $\psi_{i,p} \in \mathbb{V}(\mathcal{T}_i)$ the nodal basis function associated with

a node $p \in \mathcal{P}(\mathcal{T}_i)$ and by $\omega_{i,p}$ the local patch (i.e. the support of $\psi_{i,p}$). The subspace corresponding to the compatible bisection b_i, which introduces the new vertex $p_i \in \mathcal{P}(\mathcal{T}_i)$, can be written as $\mathbb{V}_i := \text{span}\{\psi_{i,p}, p \in \mathcal{P}(\mathcal{T}_i) \cap \omega_{i,p_i}\}$. To enforce the homogenous Dirichlet boundary condition, we simply set $\mathbb{V}_i = \varnothing$ if p_i is a vertex on the boundary. Let $\mathbb{V}_0 = \mathbb{V}(\mathcal{T}_0)$ be the linear space corresponding to the initial mesh $\mathcal{T}_0$. Then we have a space decomposition $\mathbb{V} = \sum_{i=0}^{m} \mathbb{V}_i$.

Based on this space decomposition, there are infinitely many possibilities to decompose the feasible set $\mathbb{K}$. We do not consider the optimal way to choose such a constraint decomposition. We simply choose

$$\mathbb{K} = \sum_{i=0}^{m} \mathbb{K}_i \quad \text{with} \quad \mathbb{K}_i := \{v \in \mathbb{V}_i \,|\, v \geq 0\}, \tag{8}$$

and focus on how to decompose $u \in \mathbb{V}$ at each iteration in Algorithm 1. Let $\mathbb{W}_j = \sum_{i=0}^{j} \mathbb{V}_i, j = 1, \ldots, m$. For $i = m, m-1, \ldots, 1$, we first define $I_i^{i-1} : \mathbb{W}_i \to \mathbb{W}_{i-1}$ such that

$$I_i^{i-1} v(p) = \begin{cases} \min\{v(p), v(p_i)\}, & \text{if } p \in \mathcal{P}(\mathcal{T}_{i-1}) \cap \omega_{i,p_i} \\ v(p), & \text{if } p \in \mathcal{P}(\mathcal{T}_{i-1}) \backslash \omega_{i,p_i}. \end{cases}$$

We then define $Q_i : \mathbb{V} \to \mathbb{W}_{i-1}$ to be $Q_i := I_i^{i-1} I_{i+1}^{i} \cdots I_m^{m-1}$. Notice that Q_i's are nonlinear operators, i.e. $Q_i u - Q_i v \neq Q_i(u - v)$. Finally we define a decomposition $u = \sum_{i=0}^{m} u_i$, with

$$u_m := u - Q_m u, \quad u_i := Q_{i+1} u - Q_i u \ (i = m-1, \ldots, 1), \quad u_0 = Q_1 u. \tag{9}$$

Comparing these with the definitions of $\mathbb{V}_i$ and $\mathbb{K}_i$, we can easily see that $u_i \in \mathbb{K}_i$, for $i = 0, 1, \ldots, m$.

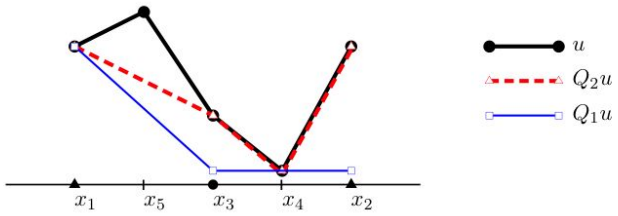


Fig. 2. A one-dimensional example for the decomposition of u. Suppose the initial grid $\mathcal{T}_0 = \{(x_1, x_3), (x_3, x_2)\}$. And the final grid $\mathcal{T}$ can be viewed as $\mathcal{T}_0 + b_1 + b_2$ where b_1 bisects the element (x_3, x_2) and introduces x_4 and b_2 bisects (x_1, x_2) and introduces x_5. As we discussed above $\mathcal{T}_1 = \mathcal{T}_0 + b_1$ and $\mathcal{T} = \mathcal{T}_2 = \mathcal{T}_1 + b_2$. From the definition of Q_i, we can easily obtain a decomposition of u.

Now we prove the convergence rate of the proposed algorithm.

Lemma 1 (Stability of Q_i). *Let $u, v \in \mathbb{V}$. For $i = 0, 1, \ldots, m$ and any element $\tau \in \mathcal{T}_i$, we have*

$$h_\tau^{-1}\|Q_{i+1}u - Q_{i+1}v\|_{L^2(\tau)} \le C\big(1 + |\log(h_\tau/h_{\min})|\big)^{\frac{1}{2}}\|u-v\|_{H^1(\omega_{i,\tau})},$$

where C is a generic constant independent of the meshsize.

Proof. From the definition of Q_i, for any $u, v \in \mathbb{V}$, we have that

$$\|Q_{i+1}u - Q_{i+1}v\|_{L^2(\tau)} \le C \sum_{p\in\mathcal{P}(\mathcal{T}_i)\cap\tau} \|u-v\|_{L^\infty(\omega_{i,p})}|\tau|^{\frac{1}{2}} \le Ch_\tau\|u-v\|_{L^\infty(\omega_{i,\tau})}.$$

The result then follows directly from the discrete Sobolev inequality between L^∞ and H^1 in two dimensions; see [2]. □

We introduce the *generation* of elements and compatible bisections. The generation of each element in the initial grid $\mathcal{T}_0$ is defined to be 0, and the generation of a child is 1 plus that of the father. In [13] we proved that all triangles in a compatible patch ω_e are of the same generation, which can be used to define the generation, gen$(\cdot)$, for a compatible bisection b_e and the corresponding new vertex. For two different compatible bisections, b_{e_1} and b_{e_2}, of the same generation, their patches are disjointed, i.e., $\omega_{e_1} \cap \omega_{e_2} = \varnothing$.

Lemma 2 (Stable Decomposition). *For any $u, v \in \mathbb{K}$, the decompositions $u = \sum_{i=0}^m u_i$, $v = \sum_{i=0}^m v_i$ given by (9) satisfy*

$$\left(\sum_{i=0}^m \|\!|u_i - v_i|\!\|^2\right)^{\frac{1}{2}} \le C\big(1 + |\log h_{\min}|\big)\,|\!|\!|u-v|\!|\!|\,;$$

Proof. First note that the support of $\mathbb{V}_i$ is restricted to the extended patch $\widetilde{\omega}_{i,p_i} := \cup_{x\in\omega_{i,p_i}}\omega_{i,x}$. Using an inverse inequality and stability of Q_i, we have

$$\|\!|u_i - v_i|\!\|^2_{\widetilde{\omega}_{i,p_i}} \le C\|h_\tau^{-1}(u_i - v_i)\|^2_{L^2(\widetilde{\omega}_{i,p_i})} \le C(1 + |\log h_{\min}|)\,|\!|\!|u-v|\!|\!|^2_{\widetilde{\omega}_{i,p_i}}.$$

For bisections with the same generation k, the extended patches, $\widetilde{\omega}_{i,p_i}$, have finite overlapping and $\cup_{p,\text{gen}(p)=k}\widetilde{\omega}_{i,p_i} \le C|\Omega|$. Let $L = \max_{\tau\in\mathcal{T}}\text{gen}(\tau)$. Then

$$\sum_{i=1}^m \|\!|u_i - v_i|\!\|^2_{\widetilde{\omega}_{i,p_i}} = \sum_{k=1}^L \sum_{p_i,\text{gen}(p_i)=k} \|\!|u_i - v_i|\!\|^2_{\widetilde{\omega}_{i,p_i}} \le CL(1+|\log h_{\min}|)\,|\!|\!|u-v|\!|\!|^2_\Omega.$$

The result then follows from the observation that $L \le C|\log h_{\min}|$. □

The proof of the following Strengthened Cauchy–Schwarz (SCS) inequality can be found in [13]. The idea of the proof is to apply standard SCS for each compatible decomposition and then rearrange the sum by generations.

Lemma 3 (Strengthened Cauchy Schwarz Inequality). *For any $u_i, v_i \in \mathbb{V}_i, i = 0, \ldots, m$, we have*

$$\Big|\sum_{i=0}^m\sum_{j=0}^m(\nabla u_i, \nabla v_j)\Big| \le C\Big(\sum_{i=0}^m |u_i|_1^2\Big)^{1/2}\Big(\sum_{i=1}^m |v_i|_1^2\Big)^{1/2}. \tag{10}$$

Applying the abstract theory (Theorem 1) and Lemma 2 and Lemma 3, we get the following rate of convergence.

Theorem 3 (Convergence Rate) *Let u^k be the k-th iteration of Algorithm 1 with the decomposition (9). We then have the following convergence rate*

$$\mathcal{J}(u^k) - \mathcal{J}(u^*) \leq C\Big(1 - \frac{1}{1 + |\log h_{\min}|^2}\Big)^k. \tag{11}$$

4 Numerical Experiments

In this section, we use a numerical example by [10] to test the proposed algorithm: Let $\Omega = (-2, 2)^2$, $f = 0$ and the obstacle $\chi(x) = \sqrt{1 - |x|^2}$ if $|x| \leq 1$ and -1, otherwise. In this case, the exact solution is known to be

$$u_*(x) = \begin{cases} \sqrt{1 - |x|^2} & \text{if } |x| \leq r_* \\ -r_*^2 \ln(|x|/2)\sqrt{1 - r_*^2} & \text{otherwise,} \end{cases}$$

where $r_* \approx 0.6979651482$. We give the Dirichlet boundary condition according to the exact solution above.

Table 1. The reduction factors for the CDM algorithm on adaptively refined meshes. The reduction factor is the ratio of energy error between two consecutive iterations.

Adaptive mesh	Degrees of freedom	$h_{\min}$	Reduction factor
1	719	1.563e-2	0.508
2	1,199	1.105e-2	0.599
3	2,107	7.813e-3	0.660
4	3,662	5.524e-3	0.651
5	6,560	3.901e-3	0.691
6	1,1841	2.762e-3	0.701

The contraction factors are computed and reported in Table 1 for a sequence of adaptive meshes, where the adaptive mesh refinement is driven by a posteriori error estimators starting from a uniform initial mesh; such adaptive algorithms and estimators can be found in [9] for example. The linear convergence rate is confirmed by our numerical experiments and the reduction rate is evaluated when the convergence becomes linear; there is a superlinear region in the beginning.

References

1. L. Badea, X.-C. Tai, and J. Wang. Convergence rate analysis of a multiplicative schwarz method for variational inequalities. *SIAM J. Numer. Anal.*, 41(3):1052–1073, 2003.
2. J.H. Bramble, J.E. Pasciak, and A.H. Schatz. The construction of preconditioners for elliptic problems by substructuring, I. *J. Comput. Math.*, 47:103–134, 1986.

3. L. Chen, R.H. Nochetto, and J. Xu. Local multilevel methods on graded bisection grids for H^1 system. Submitted to *J. Comput. Math.*, 2009.
4. L. Chen and C.-S. Zhang. A coarsening algorithm and multilevel methods on adaptive grids by newest vertex bisection. Submitted to *J. Comput. Math.*, 2009.
5. C.W. Cryer. Successive overrelaxation methods for solving linear complementarity problems arising from free boundary problems. *Proceedings of Intensive Seminary on Free Boundary Problems, Pavie, Ed. Magenes*, 1979.
6. C. Graser and R. Kornhuber. Multigrid methods for obstacle problems. *J. Comput. Math.*, 27(1):1–44, 2009.
7. R. Kornhuber. *Adaptive Monotone Multigrid Methods for Nonlinear Variational Problems*. Advances in Numerical Mathematics B. G. Teubner, Stuttgart, 1997.
8. R.H. Nochetto, K.G. Siebert, and A. Veeser. Theory of adaptive finite element methods: an introduction. In R.A. DeVore and A. Kunoth, editors, *Multiscale, Nonlinear and Adaptive Approximation*. Springer, Heidelberg, 2009.
9. K.G. Siebert and A. Veeser. A unilaterally constrained quadratic minimization with adaptive finite elements. *SIAM J. Optim.*, 18(1):260–289, 2007.
10. X.-C. Tai. Rate of convergence for some constraint decomposition methods for nonlinear variational inequalities. *Numer. Math.*, 93(4):755–786, 2003.
11. X.-C. Tai, B. Heimsund, and J. Xu. Rate of convergence for parallel subspace correction methods for nonlinear variational inequalities. In *Domain Decomposition Methods in Science and Engineering (Lyon, 2000)*, Theory Eng. Appl. Comput. Methods, pp. 127–138. Internat. Center Numer. Methods Eng. (CIMNE), Barcelona, 2002.
12. X.-C. Tai and J. Xu. Global convergence of subspace correction methods for convex optimization problems. *Math. Comput.*, 71(237):105–124, 2002.
13. J. Xu, L. Chen, and R.H. Nochetto. Optimal multilevel methods for H(grad), H(curl), and H(div) systems on adaptive and unstructured grids. In R.A. DeVore and A. Kunoth, editors, *Multiscale, Nonlinear and Adaptive Approximation*. Springer, Heidelberg, 2009.
14. J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.*, 34: 581–613, 1992.

How Close to the Fully Viscous Solution Can One Get with Inviscid Approximations in Subregions ?

Martin J. Gander[1], Laurence Halpern[2], and Veronique Martin[3]

[1] Section de mathématiques, Université de Genève, CH-1211 Genève 4, Switzerland, Martin.Gander@unige.ch
[2] LAGA, Université Paris XIII, 93430 Villetaneuse, France, halpern@math.univ-paris13.fr
[3] LAMFA UMR-CNRS 6140, Université de Picardie Jules Verne, 80039 Amiens, France, veronique.martin@u-picardie.fr

1 Introduction

The coupling of different types of partial differential equations is an active field of research, since the need for such coupling arises in various applications. A first main area is the simulation of complex objects, composed of different materials, which are naturally modeled by different equations; fluid-structure interaction is a typical example. A second main area is when homogeneous objects are simulated, but the partial differential equation modeling the object is too expensive to solve over the entire object. A simpler, less expensive model would suffice in most of the object to reach the desired accuracy. Fluid flow around an airplane could serve as an example, where viscous effects are important close to the airplane, but can be neglected further away. A third emerging area is the coupling of equations across dimensions, for example the blood flow in the artery can be modeled by a one dimensional model, but in the heart, it needs to be three dimensional.

We are interested in this paper in the second situation, where the motivation for using different equations comes from the fact that we would like to use simpler, less expensive equations in areas of the domain where the full model is not needed. We use as our guiding example the advection reaction diffusion equation. We are in principle interested in the fully viscous solution, but we would like to solve only an advection reaction equation for computational savings in part of the domain. Coupling conditions for this type of problem have been developed in the seminal paper [6], but with the first situation described above in mind, i.e. there is indeed a viscous and an inviscid physical domain, and the coupling conditions are obtained by a limiting process as the viscosity goes to zero; see also [7], and [1] for an innovative correction layer.

In his PhD thesis [2], Dubach developed coupling conditions based on absorbing boundary conditions, and such conditions have been used in order to define heterogeneous domain decomposition methods in [4]. A fundamental question however in the

second situation described above is how far the solution obtained from the coupled problem is from the solution of the original, more expensive one on the entire domain. A first comparison of different transmission conditions focusing on this aspect appeared in [5]. In [3], coupling conditions were developed for stationary advection reaction diffusion equations in one spatial dimensions, which lead to solutions of the coupled problem that can be exponentially close to the fully viscous solution, and rigorous error estimates are provided. The coupling conditions are based on the factorization of the differential operator, and the exact factorization can be used in this one dimensional steady case. We study in this paper time dependent advection reaction diffusion problems, where the exact factorization of the differential operator cannot be used any more, due to the non-local nature of the factors. Therefore new ideas are needed in order to obtain better coupling conditions than the classical ones developed for situation one, where the domains are really physically different.

2 Model Problem

We consider the time dependent advection reaction diffusion equation

$$\begin{aligned} \mathcal{L}_{ad}u := \frac{\partial u}{\partial t} - \nu\frac{\partial^2 u}{\partial x^2} + a\frac{\partial u}{\partial x} + cu = f \quad &\text{in } (-L_1, L_2)\times(0,T),\\ \mathcal{B}_1 u(-L_1,\cdot) = g_1 \ &\text{on } (0,T),\\ \mathcal{B}_2 u(L_2,\cdot) = g_2 \ &\text{on } (0,T),\\ u(x,0) = 0 \quad &\text{on } (-L_1, L_2), \end{aligned} \tag{1}$$

where a is the velocity field, $\nu > 0$ is the viscosity, $c > 0$ is the reaction and $\mathcal{B}_j$, $j = 1, 2$ are suitable boundary operators: if $a > 0$ (resp. $a < 0$) a Dirichlet condition is imposed at $x = -1$ (resp. $x = 1$) and an absorbing boundary condition of order 1 is imposed at $x = 1$ (resp. $x = -1$). We present for convenience our results using a homogeneous initial condition; in the case of an inhomogeneous initial condition u_0, the change of variables $\tilde{u}(x,t) = u(x,t) - e^{-t}u_0(x)$ leads to a problem of the form (1).

We suppose now that the viscosity term is only important in part of the domain, say in $(-L_1, 0)$, and we are thus willing to solve the full advection reaction diffusion equation there,

$$\begin{aligned} \mathcal{L}_{ad}u_{ad} = f \quad &\text{in } (-L_1, 0)\times(0,T),\\ \mathcal{B}_1 u_{ad}(-L_1,\cdot) = g_1 \ &\text{on } (0,T),\\ \mathcal{B}_{ad} u_{ad}(0,\cdot) = g_a \ &\text{on } (0,T),\\ u_{ad}(x,0) = 0 \quad &\text{on } (-L_1, L_2). \end{aligned} \tag{2}$$

We want to determine a boundary operator $\mathcal{B}_{ad}$ and a function g_a, which can only use information from solutions of advection reaction equations on the remaining domain $(0, L_2)$, such that u_{ad} is as close as possible to the fully viscous solution u on $(-L_1, 0)$. Because the viscosity is small, a first idea is to solve on the remaining domain $(0, L_2)$ the advection equation

$$\mathcal{L}_a u_a := \frac{\partial u_a}{\partial t} + a\frac{\partial u_a}{\partial x} + cu_a \quad \text{on } (0, L_2)\times(0,T). \tag{3}$$

This choice was made in [6] for the 2D stationary advection diffusion equation, and a variational coupling condition was introduced, which in our time dependent case is

$$\begin{array}{c}(-\nu u'_{ad} + a u_{ad})(0,\cdot) = a u_a(0,\cdot) \quad \text{if } a > 0 \text{ or } a < 0, \\ u_{ad}(0,\cdot) = u_a(0,\cdot) \quad \text{if } a > 0.\end{array} \tag{4}$$

We have shown in [3] that for the stationary case of (2) there exist coupling conditions which lead to coupled solutions that are much closer to the fully viscous solution on the entire domain than with the coupling conditions (4). The purpose of the present paper is to investigate if similar coupling conditions exist for (2).

3 Factorization of the Differential Operator

Let $\hat{u}(s) = \int_0^{+\infty} u(t)e^{-st}dt$, $\mathcal{R}(s) > \alpha$ be the Laplace transform of the continuous function u with $|u(t)| \leq e^{\alpha t}$, $t > 0$. Performing the Laplace transform of equation (1), we obtain

$$-\nu \frac{\partial^2 \hat{u}}{\partial x^2} + a\frac{\partial \hat{u}}{\partial x} + (c+s)\hat{u} = \hat{f}.$$

The characteristic roots of this equation are

$$\lambda^+ = \frac{1}{2\nu}(a + \sqrt{a^2 + 4\nu(c+s)}) \quad \text{and} \quad \lambda^- = \frac{1}{2\nu}(a - \sqrt{a^2 + 4\nu(c+s)}), \tag{5}$$

and we obtain a factorization of the Laplace transformed operator,

$$\widehat{\mathcal{L}}_{ad} = (a\partial_x - a\lambda^+)(-\frac{\nu}{a}\partial_x + \frac{\nu}{a}\lambda^-).$$

The two factors represent evolution operators in the x direction, one into the positive, and the other into the negative x direction, due to the square root with principal branch having positive real part. The evolution operators are however non-local, so we propose to expand $\lambda^{\pm}$ for small viscosity ν,

$$\lambda^+ = \frac{a+|a|}{2\nu} + \frac{c+s}{|a|} + \mathcal{O}(\nu) \quad \text{and} \quad \lambda^- = \frac{a-|a|}{2\nu} - \frac{c+s}{|a|} + \mathcal{O}(\nu). \tag{6}$$

If we truncate these expansions to obtain approximations $\lambda^{\pm}_{app}$, the factorization is not exact anymore, a remainder appears on the right hand side, see for example [8], and we obtain

$$\begin{aligned}&(a\partial_x - a\lambda^+_{app})(-\frac{\nu}{a}\partial_x + \frac{\nu}{a}\lambda^-_{app})\hat{u}(x,s) \\ &= -\nu\frac{\partial^2 \hat{u}}{\partial x^2}(x,s) + a\frac{\partial \hat{u}}{\partial x}(x,s) + (c+s)\hat{u}(x,s) - (\nu\lambda^+_{app}\lambda^-_{app} + (c+s))\hat{u}(x,s) \\ &= \hat{f} - (\nu\lambda^+_{app}\lambda^-_{app} + (c+s))\hat{u}(x,s).\end{aligned} \tag{7}$$

4 Optimal Coupling Conditions and Approximations

We start by deriving an optimal coupling condition: integrating (7) on $(0, L_2)$ once yields

$$\begin{aligned}(-\nu\frac{\partial \hat{u}}{\partial x} + \nu\lambda^-_{app}\hat{u})(0,s) = (-\nu\frac{\partial \hat{u}}{\partial x} + \nu\lambda^-_{app}\hat{u})(L_2,s)e^{-\lambda^+_{app}L_2} \\ - \int_0^{L_2} (\hat{f}(x,s) - (\nu\lambda^+_{app}\lambda^-_{app} + (c+s))\hat{u}(x,s))e^{-\lambda^+_{app}x}dx.\end{aligned} \tag{8}$$

The integral term suggests introducing the modified advection operator and associated equation

$$\widehat{\widetilde{\mathcal{L}}}_a\hat{w} := \left(a\partial_x - a\lambda^+_{app}\right)\hat{w} = \hat{\tilde{f}}, \tag{9}$$

since integrating this equation on $(0, L_2)$ gives $\hat{w}(0) = -\frac{1}{a}\int_0^{L_2}\hat{\tilde{f}}(x)e^{-\lambda^+_{app}x}\,dx + \hat{w}(L_2)e^{-\lambda^+_{app}L_2}$. Using this idea to replace the integral term, we find that the solution of (1) satisfies at $x = 0$ the coupling relation

$$(-\nu\frac{\partial \hat{u}}{\partial x} + \nu\lambda^-_{app}\hat{u})(0,s) = (-\nu\frac{\partial \hat{u}}{\partial x} + \nu\lambda^-_{app}\hat{u} - a\hat{\tilde{u}}_a)(L_2,s)e^{-\lambda^+_{app}L_2} + a\hat{\tilde{u}}_a(0,s), \tag{10}$$

where $\hat{\tilde{u}}_a$ is the solution of $\widehat{\widetilde{L}}_a\hat{\tilde{u}}_a = \hat{f} - (\nu\lambda^+_{app}\lambda^-_{app} + (c+s))\hat{u}$ on $(0, L_2)$. The coupling relation (10) gives an optimal coupling condition, since solving the advection diffusion equation on $(-L_1, 0)$ with the coupling condition

$$(-\nu\frac{\partial \hat{u}_{ad}}{\partial x} + \nu\lambda^-_{app}\hat{u}_{ad})(0,s) = (-\nu\frac{\partial \hat{u}}{\partial x} + \nu\lambda^-_{app}\hat{u} - a\hat{\tilde{u}}_a)(L_2,s)e^{-\lambda^+_{app}L_2} + a\hat{\tilde{u}}_a(0,s), \tag{11}$$

implies that u_{ad} is the restriction of u on $(-L_1, 0)$. But the right hand side of the coupling condition (11) depends on the fully viscous solution $\hat{u}$ on $(0, L_2)$, which we obviously do not know. We can however solve the advection equation $L_a u_a = f$ on $(0, L_2) \times (0, T)$, so that we obtain an approximation of u. This leads for $a > 0$ to the iterative procedure

$g^0 = 0$, $u_a^0 = 0$
for $k = 0, 1, 2, \ldots$

$$\begin{cases} \widetilde{\mathcal{L}}_a\tilde{u}_a^{k+1} = f - g^k & \text{on } (0, L_2) \times (0, T) \\ \tilde{u}_a^{k+1}(L_2, \cdot) = \frac{\nu}{a}(-\partial_x + \Lambda^-_{app})u_a^k(L_2, \cdot) & \text{on } (0, T) \end{cases}$$

$$\begin{cases} \mathcal{L}_{ad}u_{ad}^{k+1} = f & \text{on } (-L_1, 0) \times (0, T) \\ \mathcal{B}_1 u_{ad}^{k+1}(-L_1, \cdot) = g_1 & \text{on } (0, T) \\ \mathcal{B}_{ad}u_{ad}^{k+1}(0, \cdot) = a\tilde{u}_a^{k+1}(0, \cdot) & \text{on } (0, T) \end{cases}$$

$$\begin{cases} \mathcal{L}_a u_a^{k+1} = f & \text{on } (0, L_2) \times (0, T) \\ u_a^{k+1}(0, \cdot) = u_{ad}^{k+1}(0, \cdot) & \text{on } (0, T) \end{cases}$$

$$g^{k+1} = \mathcal{G}(u_a^{k+1})$$

end;

where $\mathcal{G} := \nu\Lambda^+_{app}\Lambda^-_{app} + c + \partial_t$ and $\Lambda^\pm_{app}$ are the differential operators corresponding to the symbols $\lambda^\pm_{app}$. The differential operators in this algorithm depend on the order of approximation of $\lambda^\pm$, and are given in Table 1 on the left. The initial conditions are

Table 1. Local approximation of the operators in the coupling algorithm.

	$a > 0$		$a < 0$	
	Order 0	Order 1	Order 0	Order 1
$\widetilde{\mathcal{L}}_a$	$a\frac{\partial}{\partial x} - \frac{a^2}{\nu}$	$a\frac{\partial}{\partial x} - \frac{\partial}{\partial t} - (\frac{a^2}{\nu} + c)$	$a\frac{\partial}{\partial x}$	$a\frac{\partial}{\partial x} + \frac{\partial}{\partial t} + c$
$\mathcal{B}_{ad}$	$-\nu\frac{\partial}{\partial x}$	$-\nu\frac{\partial}{\partial x} - \frac{\nu}{a}\frac{\partial}{\partial t} - \frac{c\nu}{a}$	$-\nu\frac{\partial}{\partial x} + a$	$-\nu\frac{\partial}{\partial x} + \frac{\nu}{a}\frac{\partial}{\partial t} + (a + \frac{c\nu}{a})$
$\mathcal{G}$	$\frac{\partial}{\partial t} + c$	$-\frac{\nu}{a^2}(\frac{\partial^2}{\partial t^2} + 2c\frac{\partial}{\partial t} + c^2)$	$\frac{\partial}{\partial t} + c$	$-\frac{\nu}{a^2}(\frac{\partial^2}{\partial t^2} + 2c\frac{\partial}{\partial t} + c^2)$

all homogeneous, except in the case of an approximation of order 1 in the modified advection problem, where the initial condition is $\tilde{u}_a^{k+1}(\cdot, 0) = -\frac{\nu}{a^2}\partial_t u_a^k(\cdot, 0)$.

If $a < 0$, the algorithm is in principle not iterative, since the advection problem (the third in the algorithm) has now as boundary condition

$$u_a^{k+1}(L_2, \cdot) = g_2 \text{ on } (0, T),$$

and can only improve the situation once. One could thus start directly with this step, but in order to compare the situation with and without this step, we leave the algorithm sequence as stated for the case of $a > 0$. The differential operators in this algorithm depend on the order of approximation of $\lambda^\pm$, and are given in Table 1 on the right. To investigate how small the error becomes in ν, and how this depends on the iteration, we present in the next section a numerical asymptotic study when the viscosity goes to zero.

5 Numerical Asymptotic Study

We chose for the data $f = 0$ and $u_0(x) = e^{-100x^2}$, $a = \pm 10$ and $c = 1$, and will consider several values for ν. The domain is $(-1, 1) \times (0, 0.1)$. Note that the support of the initial condition contains the interface between the two subdomains. We discretize the equations by centered finite differences, and the Crank–Nicolson scheme in time, with $\Delta x = \frac{1}{12800}$ and $\Delta t = \frac{1}{128000}$.

We show in Fig. 1 both for the zeroth and first order approximation the L^2 error in space and time between the coupled solution and the fully viscous mono-domain solution versus the viscosity when $a = 10$. The error in the advection region is always $\|u_a - u\|_{L^2(0,T;L^2(0,1))} = \mathcal{O}(\nu)$, and the variational method (4) and the factorization method of order 0 with one iteration give similar results, but as soon as one adds iterations, which seem to converge, or uses the first order approximation, significantly better results are obtained.

In Fig. 2, we show the L^2 error in space and time between the coupled solution

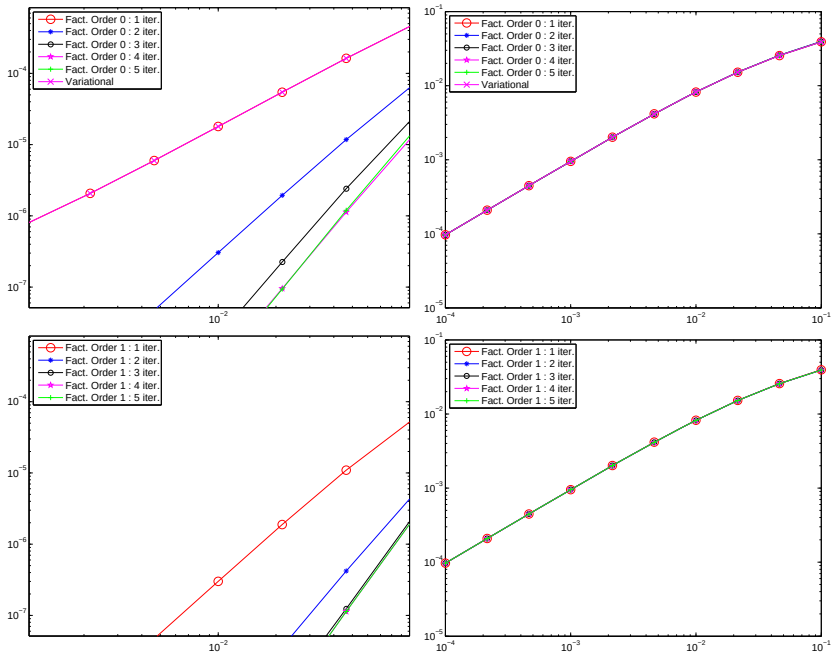


Fig. 1. Positive advection. Factorization method of order 0 and variational method (4) in the *top row*, and first order method in the *bottom row*. Error versus viscosity on $(-L_1, 0)$ on the *left*, and on $(0, L_2)$ on the *right*.

and the fully viscous mono-domain solution versus the viscosity when $a = -10$. We see that the error $\|u_a - u\|_{L^2(0,T;L^2(0,1))}$ is always $\mathcal{O}(\nu)$. It seems that using the factorization method of order 0 without iteration is not a useful method: in that case the neglected term $\nu\lambda^+_{app}\lambda^-_{app} + c + s = c + s$ is not small in ν, we need to iterate at least once, which is equivalent to changing the order in the algorithm, see the comment in Sect. 4. With the factorization method of order 1 and one iteration we obtain an error of $\mathcal{O}(\nu^2)$ which is a substantial improvement over the variational method, since in that case, because the advection is negative, the information comes from the right where the error is $\mathcal{O}(\nu)$. Our modified advection operator clearly carries more relevant information in this case.

In Table 2 we show in summary the numerically estimated dependence on the viscosity ν, both for the case of positive and negative advection.

Table 2. Numerically measured error estimates for $\|u_{ad} - u\|_{L^2(0,T;(0,1))}$.

$a > 0$					$a < 0$			
Order 0			Order 1		Order 0		Order 1	
1 iter.	2 iter.	5 iter.	1 iter.	5 iter.	1 iter.	2 iter.	1 iter.	2 iter.
$\mathcal{O}(\nu^{1.4})$	$\mathcal{O}(\nu^{2.4})$	$\mathcal{O}(\nu^{3.3})$	$\mathcal{O}(\nu^{2.4})$	$\mathcal{O}(\nu^{3.3})$	$--$	$\mathcal{O}(\nu)$	$\mathcal{O}(\nu)$	$\mathcal{O}(\nu^2)$

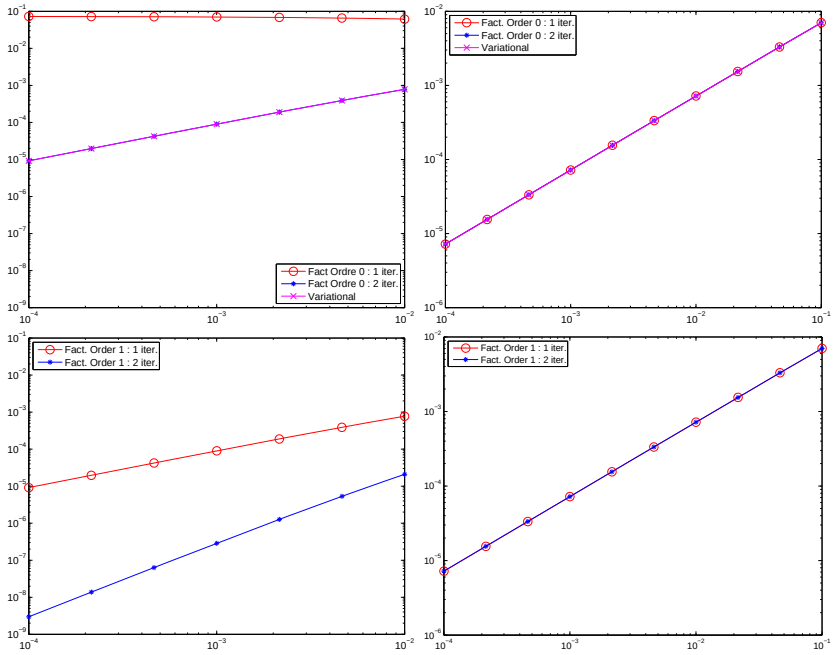


Fig. 2. Negative advection. Factorization method of order 0 and variational method in the *top row*, and first order method in the *bottom row*. Error versus the viscosity on $(-L_1, 0)$ on the *left*, and on $(0, L_2)$ on the *right*.

6 Conclusions

We have derived coupling conditions for the time dependent advection reaction diffusion equation, which lead to coupled solutions that are closer to the fully viscous solution than when using classical variational coupling conditions. Our numerical experiments allowed us to estimate the asymptotic dependence on the viscosity of the new approach, and we are currently working on rigorous error estimates for the new coupling mechanism.

References

1. C.A. Coclici, W.L. Wendland, J. Heiermann, and M. Auweter-Kurtz. A heterogeneous domain decomposition for initial-boundary value problems with conservation laws and electromagnetic fields. In T. Chan, T. Kako, H. Kawarada, and O. Pironneau, editors, *Twelfth International Conference on Domain Decomposition Methods, Chiba, Japan*, pp. 281–288. Domain Decomposition Press, Bergen, 2001.
2. E. Dubach. *Contribution à la Résolution des Équations fluides en domaine non borné*. PhD thesis, Université Paris 13, 1993.

3. M.J. Gander, L. Halpern, C. Japhet, and V. Martin. Viscous problems with inviscid approximations in subregions: A new approach based on operator factorization. *ESAIM Proc.*, 27: 272–288, 2009.
4. M.J. Gander, L. Halpern, and C. Japhet. Optimized Schwarz algorithms for coupling convection and convection-diffusion problems. In *Proceedings of the Thirteenth International Conference of Domain Decomposition*, pp. 253–260, 2001.
5. M.J. Gander, L. Halpern, C. Japhet, and V. Martin. Advection diffusion problems with pure advection approximation in subregions. In O.B. Widlund and D.E. Keyes, editors, *Domain Decomposition Methods in Science and Engineering XVI*, volume XVI of *Lecture Notes in Computational Science and Engineering 55*, pp. 239–246. Springer Berlin, Heidelberg, New York, 2007.
6. F. Gastaldi and A. Quarteroni. On the coupling of hyperbolic and parabolic systems: Analytical and numerical approach. *Appl. Numer. Math.*, 6:3–31, 1989.
7. F. Gastaldi, A. Quarteroni, and G.S. Landriani. On the coupling of two dimensional hyperbolic and elliptic equations : Analytical and numerical approach. In T. Chan et al., editor, *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pp. 22–63, SIAM, Philadelphia, PA, 1990.
8. J.P. Lohéac, F. Nataf, and M. Schatzman. Parabolic approximations of the convection-diffusion equation. *Math. Comput.*, 60:515–530, 1993.

Schwarz Waveform Relaxation Algorithms with Nonlinear Transmission Conditions for Reaction-Diffusion Equations

Filipa Caetano[1], Martin J. Gander[2], Laurence Halpern[3] and Jérémie Szeftel[4]

[1] Univ. Paris-Sud, Département de Mathématiques; CNRS, F-91405 Orsay, France, filipa.caetano@math.u-psud.fr
[2] Section de Mathématiques, Université de Genève, 1211 Genève, Switzerland, martin.gander@math.unige.ch
[3] Université Paris 13, CNRS, UMR 7539 LAGA, F-93430 Villetaneuse, France, halpern@math.univ-paris13.fr
[4] DMA, Ecole Normale Supérieure, Paris, France, szeftel@dma.ens.fr

1 Introduction

Our objective is to develop efficient parallel algorithms for reactive transport equations, which appear in problems related to the numerical simulation of geological CO_2 storage. We present in this paper a new class of Schwarz waveform relaxation (SWR) algorithms with nonlinear transmission conditions for the model problem of semilinear reaction diffusion equations. These methods are based on the partition of the spatial domain into smaller sub-domains, and then on the approximation of the restriction of the solution to each sub-domain. Transmission conditions at the interfaces must be defined in order to couple the problems between sub-domains. In the case of linear advection-reaction-diffusion equations, different types of transmission conditions were considered in [5] and in [6], in dimensions 1 and 2, and partially or numerically optimized. The optimization problem was then solved for Robin transmission conditions in [3], and for higher order conditions in [1]. We are interested here in nonlinear problems and develop for the first time associated non-linear algorithms. After introducing our problem in Sect. 2, we define in Sect. 3 a Schwarz waveform relaxation algorithm together with the different types of transmission conditions that we consider: Robin, second order and nonlinear transmission conditions. The nonlinear conditions are based on best approximation problems for the linearized equation and provide an efficient algorithm. Section 4 is devoted to the numerical implementation of the iterative algorithm and finally, in Sect. 5, we present numerical results that illustrate the performance of our new algorithms.

2 Problem Description

We consider the semilinear reaction diffusion equation in two dimensions

$$u_t - \nu \Delta u + f(u) = 0, \text{ in } \mathbb{R}^2 \times (0,T), \tag{1}$$

with initial condition

$$u(\cdot,\cdot,0) = u_0(\cdot,\cdot), \tag{2}$$

where $T > 0$ and the diffusion coefficient ν is a strictly positive constant. Let us suppose that $u_0 \in H^2(\mathbb{R}^2)$ and $f \in \mathcal{C}^2(\mathbb{R})$ are given and that f satisfies $f(0) = 0$. A weak solution of problem (1) and (2) is defined to be a function $u \in L^2(0,T;H^1(\mathbb{R}^2)) \cap \mathcal{C}([0,T];L^2(\mathbb{R}^2))$, such that $f(u) \in L^2(0,T;L^2(\mathbb{R}^2))$, satisfying for all $v \in H^1(\mathbb{R}^2)$

$$\frac{d}{dt}(u,v) + \nu(\nabla u, \nabla v) + (f(u),v) = 0, \text{ in } \mathcal{D}'(0,T),$$

and $u_{|t=0} = u_0$, where $(\cdot,\cdot)$ denotes the inner product in $L^2(\mathbb{R}^2)$.

Let us recall the following result concerning the well-posedness of the Cauchy problem (1) and (2) (for the proof, see for instance [2]):

Theorem 1. *If $f \in \mathcal{C}^2(\mathbb{R})$ and $u_0 \in H^2(\mathbb{R}^2)$, then there exists $T > 0$ such that problem (1) and (2) possesses a unique weak solution $u \in L^2(0,T;H^1(\mathbb{R}^2)) \cap \mathcal{C}([0,T];L^2(\mathbb{R}^2))$. We have in addition that $u \in L^\infty(0,T;H^2(\mathbb{R}^2))$.*

3 The Schwarz Waveform Relaxation Algorithm

We decompose the domain $\mathbb{R}^2$ into two sub-domains $\Omega_1 = (-\infty,0) \times \mathbb{R}$ and $\Omega_2 = (0,+\infty) \times \mathbb{R}$. We denote by $\Gamma := \{0\} \times \mathbb{R}$ the common boundary of Ω_1 and Ω_2 and by $n_1 = (1,0)$ and $n_2 = (-1,0)$ respectively the unit outward normals to Ω_1 and Ω_2 at Γ. We introduce the following non-overlapping Schwarz waveform relaxation algorithm to approximate the solution of problem (1) and (2). If after step k of the algorithm the pair (u_1^k, u_2^k) is known, we propose to define (u_1^{k+1}, u_2^{k+1}) by solving both problems

$$\begin{aligned} \partial_t u_i^{k+1} - \nu \Delta u_i^{k+1} + f(u_i^{k+1}) &= 0 && \text{in } \Omega_i \times (0,T), \\ u_i^{k+1}(\cdot,\cdot,t=0) &= u_{0|\Omega_i} && \text{in } \Omega_i, \\ B_i(u_i^{k+1}) &= B_i(u_j^k) && \text{on } \Gamma \times (0,T), \end{aligned} \tag{3}$$

for $i = 1,\ j = 2$ and $i = 2,\ j = 1$, where B_1 and B_2 are differential operators to be defined below. To initialize the algorithm, an initial guess (g_{b1}^0, g_{b2}^0) must be given: at step 0 of the algorithm we solve then both problems (3), $i = 1,\ 2$, with transmission conditions replaced respectively by conditions

$$B_1(u_1^0) = g_{b1}^0 \quad \text{and} \quad B_2(u_2^0) = g_{b2}^0. \tag{4}$$

It is well known that the solution of (1) and (2) as well as its normal derivative must be continuous across Γ. The issue is then to define algorithms which converge rapidly to the solution of this problem in the global domain. For the linear reaction-diffusion equation, the transparent boundary condition at the boundary Γ is obtained through a Fourier transform in time and in transverse direction y (see [1, 3]). A good approximation of the Fourier symbol can be obtained using Robin or second order, so called Ventcel, transmission conditions.

3.1 Non-overlapping Algorithms of Order Zero and Two

The non-overlapping Schwarz waveform relaxation algorithm of order zero is obtained by performing a zeroth order polynomial approximation of the Fourier symbol of the transparent boundary condition over Γ, which leads to Robin transmission conditions defined by

$$B_i(u) := \partial_{n_i} u + pu, \quad p > 0. \tag{5}$$

The non-overlapping Schwarz waveform relaxation algorithm of order 2 is obtained by performing a first order polynomial approximation of the Fourier symbol of the transparent boundary condition over Γ, which leads to the second order (or Ventcel) transmission conditions

$$B_i(u) := \partial_{n_i} u + pu + q\big(\partial_t u - \nu \partial_y^2 u\big), \quad p > 0,\ q > 0. \tag{6}$$

3.2 Well-Posedness and Convergence

For $s > \frac{1}{2}$, we introduce the function spaces $H_s^s(\Omega_i) = \{u \in H^s(\Omega_i) \,|\, u_\Gamma \in H^s(\Gamma)\}$. By using *a priori* estimates in appropriate spaces and the Gronwall lemma, we can extend the results of [3] and [1] for the linear advection-reaction-diffusion equation to the nonlinear case. We obtain the following theorem concerning the well-posedness of the initial and boundary value problems in the sub-domains, and the convergence of the algorithm.

Theorem 2. *Let $g_{b_1}^0$ and $g_{b_2}^0$ in $H^1(0,T;L^2(\Gamma)) \cap L^\infty(0,T;H^{\frac{1}{2}}(\Gamma))$, $u_0 \in H^2(\mathbb{R}^2)$, $f \in \mathcal{C}^2(\mathbb{R})$, $p > 0$ and $q \geq 0$ be given. Then*

(i) There exists $T > \overline{T} > 0$ such that algorithm (3), initialized with (4), and with the transmission operators defined by (6) (or by (5) if $q = 0$), defines a unique sequence of iterates (u_1^k, u_2^k) such that

$$u_i^k \in \begin{cases} L^2(0,\overline{T};H_2^2(\Omega_i)) \cap L^\infty(0,\overline{T};H^2(\Omega_i)) \cap H^1(0,\overline{T};L^2(\Omega_i)), \text{ if } q > 0, \\ L^2(0,\overline{T};H^2(\Omega_i)) \cap L^\infty(0,\overline{T};H^2(\Omega_i)) \cap H^1(0,\overline{T};L^2(\Omega_i)), \text{ if } q = 0. \end{cases}$$

Furthermore $u_{i\,|\Gamma}^k \in H^1(0,\overline{T};L^2(\Gamma))$, and $\partial_t u_i^k \in L^\infty(0,\overline{T};L^2(\Omega_i))$.

(ii) The sequence (u_1^k, u_2^k) converges, as $k \to \infty$, to $(u_{|\Omega_1}, u_{|\Omega_2})$.

4 Discretization

We discretize the sub-domain problems by finite elements in space and a finite difference in time, implicit for the linear part and explicit for the nonlinear term. We describe here the numerical method. We are interested in the boundary value problem

$$\begin{aligned} u_t - \nu\Delta u + f(u) &= 0, \quad \text{in } \Omega_i \times (0,T), \\ u_{|t=0} &= u_0, \text{ in } \Omega_i, \\ \partial_{n_i} u + pu + q(\partial_t u - \nu\partial_y^2 u) &= g, \quad \text{on } \Gamma \times (0,T), \end{aligned} \tag{7}$$

for a given function g defined on $\Gamma \times (0,T)$. We consider V_h, a finite dimensional subspace of $H^1(\Omega_i)$ of finite $\mathbb{P}_1$ elements, and a basis $\Phi_1, \ldots, \Phi_M$ of V_h, $N_1, \ldots, N_M$ being the mesh points. We search an approximate solution $u_h(t) = u_1(t)\Phi_1 + \cdots + u_M(t)\Phi_M$, which satisfies

$$\begin{aligned} (u_h', \phi_i) + \nu(\nabla u_h, \nabla\phi_i) + \nu p(u_h, \phi_i)_\Gamma + \nu q(u_h', \phi_i)_\Gamma + \\ \nu q(\frac{\partial u_h}{\partial y}, \frac{\partial \phi_i}{\partial y})_\Gamma + (f(u_h), \phi_i) = \nu(g, \phi_i)_\Gamma, \ \forall i = 1, \ldots, M, \end{aligned}$$

where $(\cdot,\cdot)$ is the inner product in $L^2(\Omega_i)$, and $(\cdot,\cdot)_\Gamma$ is the inner product in $L^2(\Gamma)$. We denote by $t^n = n\Delta t$ the time grid points, and let $u^n := u_1^n\Phi_1 + \cdots + u_M^n\Phi_M$ be the approximate solution at time t^n. If the approximate numerical solution $U^n = (u_1^n, \cdots, u_M^n)$ at time t^n is given, the solution U^{n+1} at time t^{n+1} is computed by solving the algebraic system

$$\begin{aligned} \Big(\frac{M}{\Delta t} + \nu K + \nu M_\Gamma \text{Diag}_p + \nu\frac{M_\Gamma}{\Delta t}\text{Diag}_q + \nu K_\Gamma \text{Diag}_q\Big) U^{n+1} = \\ \frac{(M + \nu M_\Gamma \text{Diag}_q)}{\Delta t} U^n - MF(U^n) + \nu M_\Gamma G^{n+1}, \end{aligned}$$

where the mass and stiffness matrices are $M_{i,j} = (\phi_j, \phi_i)$, $K_{i,j} = (\nabla\phi_j, \nabla\phi_i)$, and on the boundary $M_{\Gamma i,j} = (\phi_j, \phi_i)_\Gamma$, $K_{\Gamma i,j} = (\partial_y\phi_j, \partial_y\phi_i)$. Diag_p and Diag_q are the diagonal matrices $\text{diag}(p, \ldots, p)$ and $\text{diag}(q, \ldots, q)$, and we set $F(U^n) = (f(u_1^n), \ldots, f(u_M^n))$ and $G^{n+1} = \big(g(N_1, t^{n+1}), \cdots, g(N_M, t^{n+1})\big)$.

4.1 Nonlinear Transmission Conditions

The linear Robin and second order transmission conditions defined by the operators (5) and (6) imply a choice of the constants p and q. In [1, 3], the authors established asymptotic formulas (in Δt) for the values of p and q that optimize the convergence factor of the algorithm, in the case of the linear advection-reaction-diffusion equation. The results are based on Fourier transforms in time and in the transversal direction y, of the error equations, which are just the homogeneous counterpart of the equations for u_i^k in this linear case. For $\partial_t u - \nu\Delta u + bu = 0$, where b is a positive constant, explicit formulas for the optimal parameters are given,

$$p_{opt}^{R}(\Delta t, b, \nu), \tag{8}$$

in the case of Robin transmission conditions (see [3]), and

$$(p_{opt}^{V}, q_{opt}^{V})(\Delta t, b, \nu), \tag{9}$$

in the case of second order transmission conditions (see [1]).

Such an explicit analysis seems difficult for a nonlinear equation, since on the one hand the equation satisfied by the errors is not the same, and on the other hand we do not know the Fourier transform of the nonlinear term $f(u)$. However, the equation satisfied by the errors $e_i^k := u_i^k - u$ is

$$\partial_t e_i^k - \nu \Delta e_i^k + f(u_i^k) - f(u) = 0,$$

and a linearization at the solution u gives

$$\partial_t e_i^k - \nu \Delta e_i^k + f'(u) e_i^k \simeq 0.$$

This motivates our choice of nonlinear transmission conditions, where we replace b by $f'(u)$, in the formulas (8) for Robin, and (9) for second order transmission conditions. Considering nonlinear transmission conditions leads to the discretization of the boundary value problem (7), where in the linear operators (5) and (6), the constants p and (p, q) are replaced by non linear functions $p(u) = p_{opt}^{R}(\Delta t, f'(u), \nu)$ and $(p, q)(u) = (p_{opt}^{V}, q_{opt}^{V})(\Delta t, f'(u), \nu)$. In this case the diagonal matrices Diag_p and Diag_q are replaced by the time-dependent matrices

$$\mathrm{Diag}_p^n = \mathrm{diag}(p(u_1^n), \ldots, p(u_M^n)), \quad \mathrm{Diag}_q^n = \mathrm{diag}(q(u_1^n), \ldots, q(u_M^n)).$$

4.2 Implementation of the Iterative Algorithm

One step of the iterative Schwarz waveform relaxation algorithm consists in solving both initial boundary value problems in each sub-domain and in defining the new boundary conditions for the next step. We must then discretize the operator $(u_k^1, u_k^2) \longrightarrow (B_1(u_k^2), B_2(u_k^1))$. To do so, we remark that, if at step k of the algorithm, the transmission conditions are defined by

$$\partial_{n_i} u_i^k + p(u_i^k) u_i^k + q(u_i^k)(\partial_t u_i^k - \nu \partial_y^2 u_i^k) = g_i^k, \tag{10}$$

$i = 1,\ 2$, (with the possibility to take into account constant functions $p(u)$ and $q(u)$ or $q(u) = 0$), at step $k+1$, the transmission conditions are defined by (10), with

$$g_i^{k+1} = -g_j^k + 2p(u_j^k) u_j^k + 2q(u_j^k)(\partial_t u_j^k - \nu \partial_y^2 u_j^k),$$

with $i = 1,\ j = 2$ or $i = 2,\ j = 1$. Rewriting the transmission condition in this way has the advantage that no normal derivative has to be computed (cf. [4] for further details on this kind of technique). We discretize then the boundary condition g_i^k using the discretizations of the corresponding terms defined in the previous paragraphs.

5 Numerical Results

In this section, the spatial domain is the square $\Omega = (-1, 1) \times (0, 2)$, which is decomposed into two sub-domains $\Omega_1 = (-1, 0) \times (0, 2)$ and $\Omega_2 = (0, 1) \times (0, 2)$. The nonlinear function that we test here is the function $f(u) = 10(\exp(u) - 1)$. We compare in the next figures the results obtained with the linear and nonlinear Robin and second order transmission conditions described in the previous sections. The figures represent the error between the domain decomposition solution obtained after a fixed number of iterations, and the so-called mono-domain solution, which corresponds to the numerical solution computed in the global domain Ω, by using the same numerical method. The boundary conditions at the boundary $\partial\Omega$ are of Dirichlet type. We consider three spatial meshes, corresponding to the values of $h = 0.125,\ h = 0.0625$ and $h = 0.03125$ and two values for the diffusion coefficient $\nu,\ \nu = 0.1$ and $\nu = 1$. The time step Δt is such that $\Delta t = h$. The time interval is $[0, 1]$.

In Fig. 1, we compare, in the case $\nu = 0.1$, the results obtained with the nonlinear

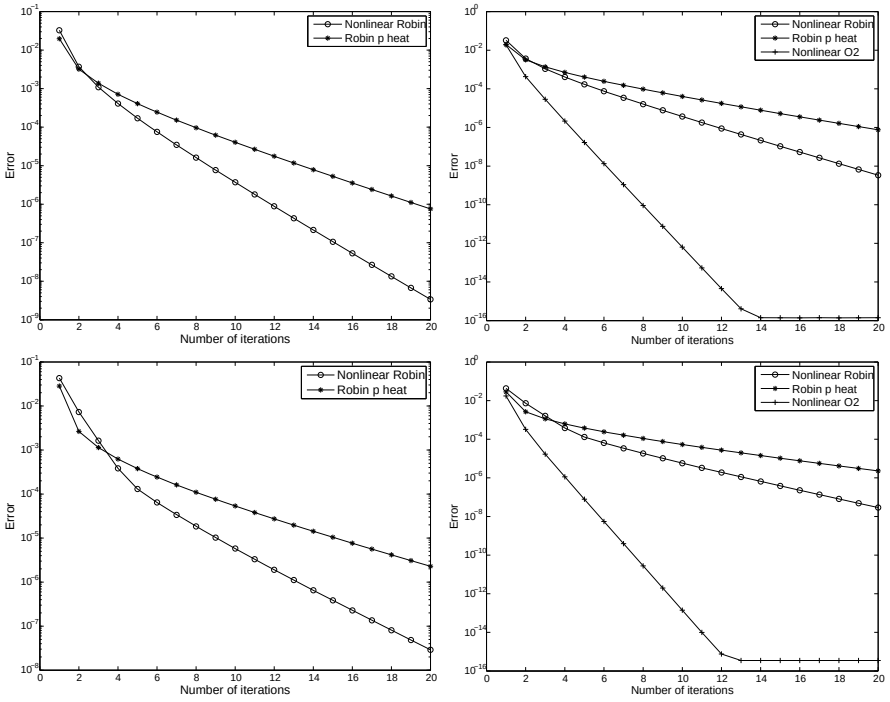


Fig. 1. $f(u) = 10(\exp(u) - 1),\ \nu = 0.1$: Difference in the $L^\infty(0, T; L^2(\Omega))$ norm between domain decomposition and mono-domain solutions, for $h = 0.0625$ in the *top row*, and $h = 0.03125$ in the *bottom row*. On the *left*, Robin transmission conditions, and on the *right* Robin and second order transmission conditions.

Robin conditions, the nonlinear second order conditions and the linear Robin conditions where the parameter p corresponds to the optimal parameter for the heat equation. These results validate the use of the nonlinear parameters. The use of nonlinear second order conditions gives better results than both linear and nonlinear Robin conditions: the convergence speed of the algorithm with second order transmission conditions is higher than the convergence speed of the algorithm with Robin transmission conditions. This result was also expected, since the second order conditions correspond to a higher order approximation of the transparent boundary condition on Γ. We obtained the same qualitative results with other nonlinear functions such as $f(u) = u^3,\ f(u) = u^5$ and other functions with a polynomial behavior.

5.1 A Simple Model in Geological CO_2 Storage Modeling

We present here a very simple model of a reactive system which can appear in the framework of geological CO_2 storage modeling. We consider a reactive chemical system with two types of materials, evolving according to the equation

$$u_t - \nu \Delta u + f(x, y, u) = 0.$$

The nonlinear function f depends on the space variables, describing a heterogeneous distribution of the materials in the spatial domain. Both materials are evolving through equilibrium values u_1^{eq} and u_2^{eq}. The reaction is described here by the function

$$f(x, y, u) = k_1 S_1(x, y)(u - u_1^{eq})^3 + k_2 S_2(x, y)(u - u_2^{eq})^3.$$

The positive constants k_1 and k_2 represent the reaction speeds of material 1 and 2, and the surface functions S_i describe the spatial distribution of the material $i,\ i = 1,\ 2$.

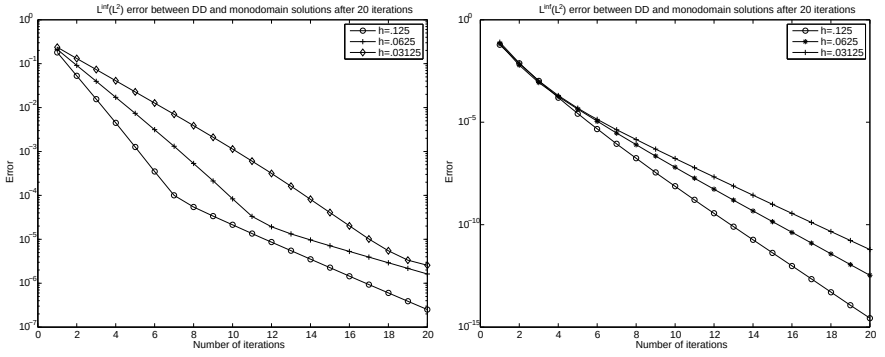


Fig. 2. $f(x, y, u),\ \nu = 1$: Error after 20 iterations of the algorithm for $h = 0.125,\ h = 0.0625$ and $h = 0.03125$. On the *left*, Robin transmission conditions, on the *right* second order transmission conditions.

In the test below, we considered $k_1 = 5,\ u_1^{eq} = 1,\ k_2 = 3,\ u_2^{eq} = 0, S_1(x,y) = \sin(\frac{3\pi}{2}x + \frac{\pi}{2})\sin(\frac{3\pi}{2}y + \frac{\pi}{2})\chi_W$, where W is a zone corresponding to a part of a circle in the spatial domain, and $S_2(x,y) = \max\big(\sin(\frac{5\pi}{2}x + \frac{\pi}{3})\sin(\frac{5\pi}{2}y + \frac{\pi}{3}), 0\big)$. The initial and Dirichlet data are both equal to 0.5. We used here nonlinear Robin and second order transmission conditions, obtained by replacing b with $\partial_u f$ in formula (8). In Fig. 2 we compare the results obtained with the nonlinear conditions for different values of the mesh-spacing h.

Acknowledgement. We thank Anthony Michel for proposing the reactive transport model of the last section, and explaining it to us. This research is supported by the research project SHPCO2 funded by ANR-07-CIS7-007-03.

References

1. D. Bennequin, M.J. Gander, and L. Halpern. A homographic best approximation problem with application to optimized Schwarz waveform relaxation. *Math. Comput.*, 78(265): 185–223, 2009.
2. T. Cazenave and A. Haraux. *An Introduction to Semilinear Evolution Equations*, volume 13 of *Oxford Lecture Series in Mathematics and Its Applications*. The Clarendon Press, Oxford University Press, New York, NY, 1998.
3. M.J. Gander and L. Halpern. Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems. *SIAM J. Numer. Anal.*, 45(2):666–697 (electronic), 2007.
4. M.J. Gander, F. Magoulès, and F. Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24(1):38–60 (electronic), 2002.
5. C. Japhet, F. Nataf, and F.-X. Roux. Extension of a coarse grid preconditioner to nonsymmetric problems. In *Domain decomposition methods, 10 (Boulder, CO, 1997)*, volume 218 of *Contemporary Mathematical*, pp. 279–286. AMS, Providence, RI, 1998.
6. V. Martin. An optimized Schwarz waveform relaxation method for the unsteady convection diffusion equation in two dimensions. *Appl. Numer. Math.*, 52(4):401–428, 2005.

Recent Advances in Schwarz Waveform Moving Mesh Methods – A New Moving Subdomain Method

Ronald D. Haynes

Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's, NL, Canada A1C 5S7, rhaynes@mun.ca

1 Introduction

It is well accepted that the efficient solution of complex partial differential equations (PDEs) often requires methods which are adaptive in both space and time. In this paper we are interested in a class of spatially adaptive moving mesh (r-refinement) methods introduced in [9, 10, 12]. Our purpose is to introduce and explore a natural coupling of domain decomposition, Schwarz waveform relaxation (SWR) [4], and spatially adaptive moving mesh PDE (MMPDE) methods for time dependent PDEs. SWR allows the focus of computational energy to evolve to the changing behaviour of the solution locally in regions or subdomains of the space-time domain. In particular, this will enable different time steps and indeed integration methods in each subdomain. The spatial mesh, provided by the MMPDE, will react to the local solution dynamics, providing distinct advantages for problems with evolving regions of interesting features.

In this paper we detail and compare approaches which couple SWR with moving meshes. Section 2 provides a brief review of the r-refinement method. We contrast the related approaches introduced in [6, 7] with a new moving subdomain method in Sect. 3. We conclude in Sect. 4 with a brief presentation of numerical results to demonstrate the moving subdomain method.

2 Moving Meshes

A recent and thorough review of moving mesh methods may be found [2] and further details are provided in the extensive bibliography therein.

Moving mesh methods solve for the solution and underlying mesh simultaneously. Consider the solution of a PDE of the form

$$u_t = \mathcal{L}(u) \qquad 0 < x < 1, \quad t > 0,$$

subject to appropriate initial and boundary conditions, where $\mathcal{L}$ denotes a spatial differential operator in the physical coordinate x. Our goal is to find, for fixed t, a

one-to-one coordinate transformation

$$x = x(\xi, t) : [0, 1] \to [0, 1], \quad \text{with } x(0, t) = 0,\ x(1, t) = 1$$

such that $u(x(\xi, t), t)$ is sufficiently smooth that a simple mesh, often uniform $\xi_i = \frac{i}{N}, i = 0, \ldots, N$ can be used to resolve solution features in the computational variable $\xi \in [0, 1]$. The mesh in the physical coordinate x is then specified from the mesh transformation by $x_i(t) = x(\xi_i, t),\ i = 0, 1, \ldots, N$.

A standard way to perform mesh adaptation in space is to use the equidistribution principle (EP). Given some measure $M(t, x, u)$ of the error in the solution, the (EP) requires that the mesh points satisfy

$$\int_{x_{i-1}}^{x_i} M(t, \tilde{x}, u)\, d\tilde{x} \equiv \frac{1}{N} \int_0^1 M(t, \tilde{x}, u) d\tilde{x},$$

or equivalently

$$\int_0^{x(\xi_i, t)} M(t, \tilde{x}, u)\, d\tilde{x} = \frac{i}{N}\theta(t) = \xi_i \theta(t), \tag{EP}$$

where $\theta(t) \equiv \int_0^1 M(t, \tilde{x}, u)\, d\tilde{x}$ is the total error in the solution.

Enforcing this condition concentrates mesh points where M or the error is large. It follows directly from (EP) that

$$\frac{\partial}{\partial \xi} \left\{ M(t, x(\xi, t), u) \frac{\partial}{\partial \xi} x(\xi, t) \right\} = 0. \tag{1}$$

Discretizing (1) and the physical PDE spatially results in an index-2 DAE system which is stiff and ill-conditioned – a problem numerically [1]. Consequently, the (EP) is often relaxed to require equidistribution at a later time $t + \tau$. Using Taylor series and dropping higher order terms a number of parabolic MMPDEs are developed. One particularly useful MMPDE is

$$\dot{x} = \frac{1}{\tau} \frac{\partial}{\partial \xi} \left(M(t, x(\xi, t), u) \frac{\partial x}{\partial \xi} \right). \tag{MMPDE5}$$

The relaxation parameter τ is chosen in practice so that the mesh evolves at a rate commensurate with that of the solution $u(x, t)$. A simple, popular choice is the arclength like monitor function $M(x, u, t) = (1 + \alpha |u_x|^2)^{1/2}$. This choice is based on the premise that we expect the error in the numerical solution to be largest in regions where the solution has large gradients. The choice of monitor function is often problem class dependent; generally M is related to specific powers of the solution or its derivatives. For the generalization to two and three spatial dimensions, the reader is referred to [8].

Using the mesh transformation $x = x(\xi, t)$ to rewrite the physical PDE in quasi-Lagrangian form we have $\dot{u} - u_x \dot{x} = \mathcal{L}u$, where $\dot{u} = u_t + u_x \dot{x}$. The MMPDE and physical PDE are solved simultaneously for the mesh $x(\xi, t)$ and corresponding solution $u(x(\xi, t), t)$. Traditionally, this system is solved using the moving method of

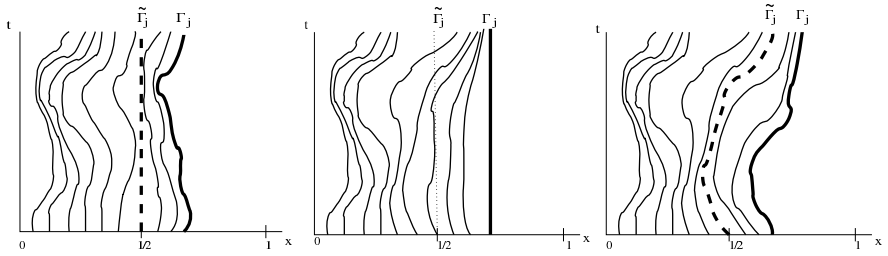
lines (MMOL) approach – the problem is discretized in space and the resulting system of ODEs is solved using a stiff IVP solver like DASSL [11]. Initial and boundary conditions for the physical PDE come from the problem description. On a fixed interval we specify $\dot{x}_0 = \dot{x}_N = 0$ as boundary conditions for the mesh. If the initial solution is smooth then an initial uniform mesh for $x(\xi, 0)$ is normally sufficient, else an initial mesh is computed which equidistributes $u(x, 0)$.

This traditional MMOL approach is not able to exploit local time scales in specific components of the solution – rather a single step size is used for all components. In practice, time step selection, via local error control, is often dictated by a very few components which are localized spatially. This suggests that a spatial partitioning via a domain decomposition (DD) approach may exploit these local time scales. A DD strategy would also enable different solution strategies in regions of a space and time; in particular a mixture of fixed and moving grids may be used as dictated by the solution. Of course various DD methods are amenable to parallel implementation – an approach not commonly utilized by the moving mesh community.

3 Domain Decomposition Strategies

Moving mesh methods naturally provide two spatial variables: the physical co-ordinate x and the computational co-ordinate ξ. DD methods partition the spatial variable into overlapping or non-overlapping subdomains. SWR iteratively solves the PDE forward in time on each subdomain. Boundary information is exchanged at the end of a time window. Designing an algorithm which couples DD and moving meshes requires a choice of the spatial variable to partition – resulting in dramatically different DD methods, see Fig. 1.

The physical space-time domain Ω is divided into non-overlapping subdomains $\tilde{\Omega}_j$ with boundaries $\partial\tilde{\Omega}_j$. $\tilde{\Gamma}_j$ is the portion of $\partial\tilde{\Omega}_j$ interior to Ω. An overlapping decomposition Ω_j is created by enlarging each $\tilde{\Omega}_j$ in such a way so that the boundaries of Ω_j interior to Ω, Γ_j, are at least some distance $\delta > 0$ from $\tilde{\Gamma}_j$.



(a) In x variable as in [7] (b) In x variable as in [6] (c) In ξ variable – moving subdomains

Fig. 1. A typical subdomain for the three SWR Moving Mesh Methods.

In this section we describe and contrast two approaches [6, 7] which utilize SWR in the physical coordinate x with a new strategy which applies SWR to the MMPDE in the computational coordinate ξ. As we will see this new approach gives rise to moving subdomains.

3.1 SWR in Physical Co-ordinates – Existing Methods

The first attempts [6, 7] to couple SWR and moving mesh methods use partitions of fixed width in the physical space.

In [7], depicted in Fig. 1a, the width of $\tilde{\Omega}_j$ is fixed. The overlap region is specified by a number of mesh points. The mesh points in the interior of $\tilde{\Omega}_j$ and in the overlap region are free to move according to the MMPDE. In this way we recover much of the strength of the moving mesh approach. The position of $\tilde{\Gamma}_j$ is fixed to ensure a reasonable partitioning of the physical space and allow the user to ensure a sufficient resolution of the subdomain by specifying an acceptable number of mesh points. Moving mesh methods are designed to prevent mesh crossings, hence the fixed location of $\tilde{\Gamma}_j$ does restrict the free flow of mesh points in and out of the overlap region. A modification of moving mesh software is required to fix the location of $\tilde{\Gamma}_j$ within Ω_j.

The algorithm solves the coupled system of physical PDE and MMPDE iteratively on overlapping subdomains. After each subdomain solve (in the Gauss–Seidel approach) or after all the subdomain solves (in the Jacobi variant) boundary information is exchanged. Dirichlet transmission conditions are specified on each subdomain. Unlike typical SWR methods, both the solution of the physical PDE on the boundary and the location of the boundary itself is exchanged. Since the overlap is a simply a number of mesh points, the location of the boundary of the neighbouring subdomain for the next iteration is extracted directly from a specific mesh trajectory obtained during the subdomain solve. The solution along that moving boundary provides the boundary data for the physical PDE. Interpolation in time is required as subdomains are free to choose time steps dictated by their own local solution dynamics.

In [6] it was realized that it is unnecessary to fix the location of $\tilde{\Gamma}_j$. As illustrated in Fig. 1b the extended subdomain Ω_j is of fixed width in the physical space, the position of Γ_j is fixed. The overlap is of fixed width but now mesh points are able to move in and out of the overlap region as directed by solution. Indeed, it differs from the typical SWR approach (cf. [3, 4]) only in the choice of the solver on each subdomain. The moving mesh solver may be used without modification. Hence [6] is better aligned with the motivation and philosophy of the DD approach.

As in [7] the user is responsible to ensure a sufficient number of mesh points reside in each subdomain to resolve any features which may arise. Although this approach may not be scalable, it may be useful in situations where the solution has many interesting features developing in disparate locations in the physical space. Current moving mesh methods on one domain have difficulty with this situation. The ability to vary the number of mesh points on each subdomain makes it easier to

ensure a sufficient number of mesh points in all parts of physical space. This suggests that this technique should be coupled with time windows and a mechanism to estimate the number of points required on that time window, ie. use an hr-refinement strategy on each subdomain. Work is ongoing to explore this idea.

The fixed location of Γ_j in [6] provides the advantage of being able to reuse quality moving mesh software as the solver on each subdomain. However, interpolation in both space and time is required to obtain the correct boundary data for the next iteration. Within each subdomain the mesh points are all moving, hence there is no guarantee that a mesh point will be located at position Γ_j at any instant in time. As a result, the boundary data for the physical PDE is obtained by interpolating (in space) the solution on the neighbouring subdomain. Subsequent interpolation in time may be required to provide the correct boundary data at the sequence of time steps chosen by the IVP software.

Applying a SWR moving mesh method in physical coordinates is conceptually analogous to the previous descriptions of Schwarz waveform relaxation on fixed grids. However, as mentioned above there are many practical challenges posed by using the moving mesh solver on each subdomain. The fixed boundaries of each subdomain require a careful choice of the number of mesh points and relatively costly interpolations to provide the boundary conditions for adjacent subdomains. The standard DD method (with fixed and uniform grids) divides the total number of physical mesh points evenly amongst the subdomains. There is a direct correspondence between the number of mesh points and the width of each subinterval. If we partition in physical space, we can not (in general) simply divide the number of mesh points required for the one domain solve evenly amongst the number of subdomains. Hence the algorithm may not scale appropriately. We begin to address these difficulties with the new method presented in the next section.

3.2 SWR in Computational Co-ordinates – A New Approach

In this paper we introduce a decomposition of the computational co-ordinate ξ into overlapping subdomains of fixed width, see Fig. 1(c). The boundaries Γ_j are fixed in ξ-space, which gives rise to time dependent boundaries in physical space – we have a *moving subdomain method*.

In the discrete version of the algorithm a subdomain is simply defined by a set number of mesh points not a region of physical space. We divide the number of mesh points required for the one domain solve evenly amongst the subdomains. As a result the method is (at least) spatially scalable. This allows mesh points (and subdomains) complete freedom to move throughout the physical space as controlled by the dynamics of the underlying solution. The subdomains provide a coarse grain adaptivity – they are chosen to automatically equidistribute the error measure in the solution and must (at least approximately) equidistribute the computational effort to compute it.

The overlap region is of fixed width in the computational space but is specified only by a fixed number of mesh points in the physical space. The required boundary values of the subdomain solution, at any time t, is obtained by interpolating the solution from the neighbouring subdomains from the previous iteration. Since the

location of the boundary is obtained from the neighbouring subdomains directly, only interpolation in time is required.

4 Numerical Results and Comments

Numerical results for SWR applied in the physical coordinates may be found in [6, 7]. Here we illustrate the new moving subdomain method for a typical test problem for moving mesh methods taken from [10]. Consider the function

$$u(x,t) = \frac{1}{2}\left[1 - \tanh(c(t)(x - t - 0.4))\right]$$
$$c(t) = 1 + \frac{999}{2}\left[1 + \tanh(100(t - 0.2))\right], \quad 0 \le x \le 1, 0 \le t \le 0.55.$$

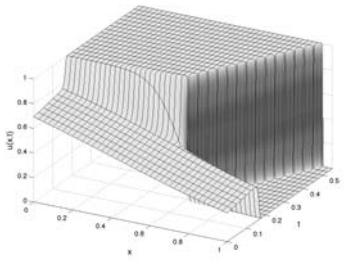


Fig. 2. Exact solution of test problem.

The exact solution, illustrated in Fig. 2, has regions of rapid transition in space and time. The surface is shaded according to the (spatial) gradient. A typical hyperbolic tangent profile develops just before $t = 0.2$ and then moves from left to right (in x).

The mesh transformation which satisfies (MMPDE5) for $u(x,t)$ is given in Fig. 3. The heavily shaded region has small $dx/d\xi$ values – these flat regions in the mesh transformation indicate a high concentration of mesh points. By design this region of high resolution corresponds to the location of sharp transition in u in Fig. 2.

In Fig. 4 we depict the mesh movement by drawing the mesh trajectories obtained during the one domain solution. Each line corresponds to the position of a grid point as a function of time. The mesh lines concentrate just before $t = 0.2$, the moment of front formation and follow the front to the right.

Figure 5 demonstrates the moving subdomains which result by solving (MMPDE5) using a SWR method in the computational coordinate ξ. Three subdomains are illustrated for two subsequent Schwarz iterations. We see that the boundaries of the subdomain are time-dependent and in fact change from iteration to iteration. The subdomains consist of an equal number of mesh points and automatically adapt to the dynamics of the solution.

Theoretical results for the alternating Schwarz iteration applied to the steady form of (1) are now available [5] and give a local convergence result. In fact, numerical evidence suggests a more robust performance. Extensions of theoretical results in

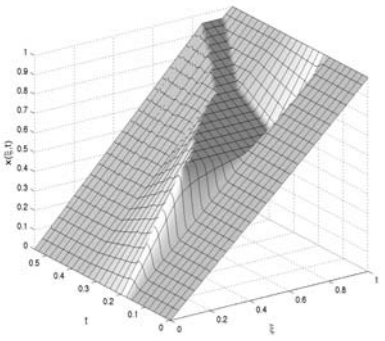


Fig. 3. One domain mesh transformation satisfying the relaxed EP.

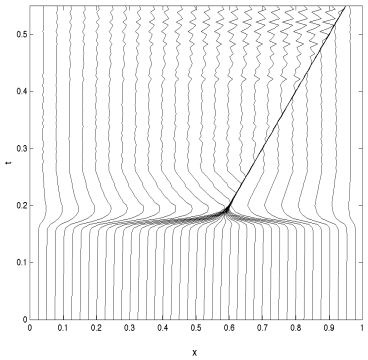
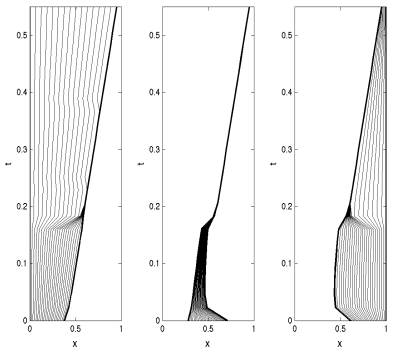
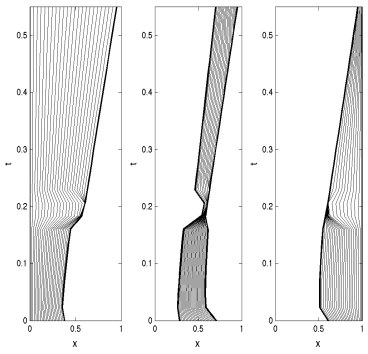


Fig. 4. One domain mesh trajectories.



(a) Schwarz iteration 1.



(b) Schwarz iteration 2.

Fig. 5. Mesh trajectories for three moving subdomains on subsequent Schwarz iterations.

the time dependent case are underway. MPI code for the two spatial dimension version of the algorithm presented in [6] is complete and rigourous numerical studies have commenced. Clearly, improved performance of these DD approaches require the development of optimal transmission conditions tuned for this class of problems. Theoretical investigations and numerical experimentation are in progress.

Acknowledgments The author would like to acknowledge the support of NSERC (Canada) under discovery grant 311796.

References

1. U.M. Ascher. DAEs that should not be solved. In *Dynamics of algorithms (Minneapolis, MN, 1997)*, volume 118 of *IMA Volumes in Mathematics and Its Applications*, pp. 55–67. Springer, New York, NY 2000.
2. C.J. Budd, W. Huang, and R.D. Russell. Adaptivity with moving grids. *Acta Numer.*, 18: 111–241, 2009. URL http://dx.doi.org/10.1017/S0962492906400015.
3. M.J. Gander and L. Halpern. Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems. *SIAM J. Numer. Anal.*, 45(2):666–697 (electronic), 2007. URL http://dx.doi.org/10.1137/050642137.
4. M.J. Gander and A.M. Stuart. Space–time continuous analysis of waveform relaxation for the heat equation. *SIAM J. Sci. Comput.*, 19(6):2014–2031, 1998.
5. R.D. Haynes. A domain decomposition approach for the equidistribution princple. In Preparation, May 2010.
6. R.D. Haynes, W. Huang, and R.D. Russell. A moving mesh method for time-dependent problems based on Schwarz waveform relaxation. In *Domain Decomposition Methods in Science and Engineering XVII*, volume 60 of *Lecture Notes in Computational Science and Engineering*, pp. 229–236. Springer, Berlin, 2008.
7. R.D. Haynes and R.D. Russell. A Schwarz waveform moving mesh method. *SIAM J. Sci. Comput.*, 29(2):656–673 (electronic), 2007.
8. W. Huang. Practical aspects of formulation and solution of moving mesh partial differential equations. *J. Comput. Phys.*, 171(2):753–775, 2001.
9. W. Huang, Y. Ren, and R.D. Russell. Moving mesh methods based on moving mesh partial differential equations. *J. Comput. Phys.*, 113(2):279–290, 1994.
10. W. Huang, Y. Ren, and R.D. Russell. Moving mesh partial differential equations (MMPDES) based on the equidistribution principle. *SIAM J. Numer. Anal.*, 31(3):709–730, 1994.
11. L.R. Petzold. A description of DASSL: a differential/algebraic system solver. In *Scientific computing (Montreal, Que., 1982)*, IMACS Trans. Sci. Comput., I, pp. 65–68. IMACS, New Brunswick, NJ, 1983.
12. Y. Ren and R.D. Russell. Moving mesh techniques based upon equidistribution, and their stability. *SIAM J. Sci. Statist. Comput.*, 13(6):1265–1286, 1992.

Optimized Schwarz Waveform Relaxation Methods: A Large Scale Numerical Study

Martin J. Gander[1], Loïc Gouarin[2], and Laurence Halpern[2]

[1] Section de mathématiques, Université de Genève, CH-1211 Genève 4, Switzerland, Martin.Gander@unige.ch

[2] Laboratoire Analyse, Géométrie et Applications Université Paris XIII, 93430 Villetaneuse, France, gouarin@math.univ-paris13.fr; halpern@math.univ-paris13.fr

1 Introduction

Schwarz waveform relaxation methods are naturally parallel methods to solve evolution problems. They are based on a decomposition of the physical domain into overlapping subdomains, and a decomposition of the time domain into time windows. On each time window, one then solves the original time dependent problem, and a subdomain iteration like in the classical Schwarz method, but now in space-time, is used in order to obtain a converged solution on the present time window. Only after convergence on the time window is the next time window treated by the algorithm. This type of algorithm was first proposed in [2], and analyzed in [8] and independently in [10]. Optimized Schwarz waveform relaxation methods were introduced in [7] to obtain more effective space-time iterative methods, compared to the classical variants, and the associated optimization problem was studied in [6] for the case of Robin conditions, and in [1] for higher order transmission conditions, see also [12]. Our extensive numerical experiments (a summary is given in Table 1), reveal that, while the theoretical parameters are asymptotically optimal, the performance can be substantially improved using a more accurate estimate for the constant. We show in this paper that this difference can be put on a theoretical foundation by taking into account geometric parameters from the decomposition. We illustrate the improved performance with the new parameters by numerical experiments, and also study numerically the dependence of the parameter on the number of subdomains.

2 Optimized Schwarz Waveform Relaxation

We study the optimized Schwarz waveform relaxation algorithm for the advection reaction diffusion equation in $\Omega \subset \mathbb{R}^2$,

$$\mathcal{L}u := u_t + \mathbf{a} \cdot \nabla u - \nu \Delta u + bu = f, \quad \text{in } \Omega \times (0,T), \tag{1}$$

where $\nu > 0$, $b \geq 0$ and $\mathbf{a} = (a, c)^T$. In order to describe the Schwarz waveform relaxation algorithm, we decompose the domain into two, possibly overlapping subdomains Ω_1 and Ω_2, with interfaces $\Gamma_1 = \partial\Omega_1 \cap \Omega_2$ and $\Gamma_2 = \partial\Omega_1 \cap \Omega_2$. The algorithm for this two subdomain decomposition calculates then for $n = 1, 2, \ldots$ the iterates (u_1^n, u_2^n) defined by

$$\begin{array}{rlrl} \mathcal{L}u_1^n = f & \text{in } \Omega_1 \times (0,T), & \mathcal{L}u_2^n = f & \text{in } \Omega_2 \times (0,T), \\ u_1^n(\cdot,\cdot,0) = u_0 & \text{in } \Omega_1, & u_2^n(\cdot,\cdot,0) = u_0 & \text{in } \Omega_2, \\ \mathcal{B}_1 u_1^n = \mathcal{B}_1 u_2^{n-1} & \text{on } \Gamma_1 \times (0,T), & \mathcal{B}_2 u_2^n = \mathcal{B}_2 u_1^{n-1} & \text{on } \Gamma_2 \times (0,T), \end{array} \tag{2}$$

where $\mathcal{B}_1$ and $\mathcal{B}_2$ are linear operators in space and time, possibly pseudo-differential, and an initial guess $\mathcal{B}_2 u_1^0(0,\cdot,\cdot)$ and $\mathcal{B}_1 u_2^0(L,\cdot,\cdot)$, $t \in (0,T)$, needs to be provided.

3 Theoretical Results

There are many different choices for the operators $\mathcal{B}_j$. The identity leads to the classical Schwarz waveform relaxation method, and zeroth or higher order conditions lead to optimized variants, see for example [1, 6]. We study here in detail the case where the transmission operators are

$$\mathcal{B}_1 := \partial_x - \frac{a-p}{2\nu} \qquad \mathcal{B}_2 := \partial_x - \frac{a+p}{2\nu}. \tag{3}$$

Using Fourier analysis, and a decomposition of the domain $\Omega = \mathbb{R}^2$ into two half spaces $\Omega_1 = (-\infty, L) \times \mathbb{R}$ and $\Omega_2 = (0, \infty) \times \mathbb{R}$, see for example [6], one can obtain the convergence factor of algorithm (2),

$$\rho(\omega, k, p) = \frac{z-p}{z+p}\, e^{-\frac{Lz}{2\nu}}, \tag{4}$$

where $z := \sqrt{x_0^2 + 4\nu^2 k^2 + 4i\nu(\omega + ck)}$ (standard branch of the square root with positive real part), $x_0^2 := a^2 + 4\nu b$, and k and ω are the Fourier variables in space and time. Computing on a grid, we assume that $k_{\max} = \frac{\pi}{h}$ where h is the local mesh size in x and y, and $\omega_{\max} = \frac{\pi}{\Delta t}$, and that we also have estimates for the lowest frequencies $k_{\min}$ and $\omega_{\min}$ from the geometry, see for example [4], or for a more precise analysis see [5].

Defining $D := \{(\omega, k), \omega_{\min} \leq |\omega| \leq \omega_{\max},\ k_{\min} \leq |k| \leq k_{\max}\}$, the parameter p^* which gives the best convergence rate is solution of the best approximation problem

$$\inf_{p \in \mathbb{C}} \sup_{(\omega,k) \in D} |\rho(\omega, k, p)| = \sup_{(\omega,k) \in D} |\rho(\omega, k, p^*)| =: \delta^*. \tag{5}$$

In what follows, we will use

$$\bar{k} := \frac{|c|(\sqrt{(|c|^2 + x_0^2)^2 + 16\nu^2\omega_{\min}^2} - |c|^2 - x_0^2)}{8\nu^2\omega_{\min}}.$$

By a direct calculation, we see that $0 \le \bar{k}|c| \le \omega_{\min}$, and we define the function

$$\varphi(k,\xi) := 2\sqrt{2}\sqrt{\sqrt{(x_0^2+4\nu^2k^2)^2+16\nu^2\xi^2}+x_0^2+4\nu^2k^2}, \tag{6}$$

and the constant

$$A = \begin{cases} \varphi(\bar{k}, -\omega_{\min}+|c|\bar{k}) & \text{if } k_{\min} \le \bar{k}, \\ \varphi(k_{\min}, -\omega_{\min}+|c|k_{\min}) & \text{if } \bar{k} \le k_{\min} \le \frac{1}{|c|}\omega_{\min}, \\ \varphi(k_{\min}, 0) & \text{if } k_{\min} \ge \frac{1}{|c|}\omega_{\min}. \end{cases} \tag{7}$$

We assume that the mesh sizes in time and space are related either by $\Delta t = C_h h$, or $\Delta t = C_h h^2$. The following theorem gives the asymptotic value of the best parameter p^* in the case of no overlap, $L = 0$, for the general case where the geometric parameters $k_{\min}$ and $\omega_{\min}$ are non zero. This is an important generalization of the result from [11], where $k_{\min} = \omega_{\min} = 0$. The proof of this result is beyond the scope of this short paper, and will appear elsewhere.

Theorem 1. *For h small, the best approximation problem (5) has a unique solution p^*, which is given asymptotically by*

$$p^* \sim \sqrt{\frac{A}{Bh}}, \quad \delta^* = 1 - \sqrt{ABh} + O(h),$$

where A is defined in (7), and

$$B = \begin{cases} \frac{2}{\nu\pi} & \text{if } \Delta t = C_h h, \\ C\frac{\sqrt{2d}}{\nu\pi} & \text{if } \Delta t = C_h h^2,\ d := \nu\pi C_h,\ C = \begin{cases} 1 & \text{if } d < d_0, \\ \sqrt{\frac{d+\sqrt{1+d^2}}{1+d^2}} & \text{if } d \ge d_0, \end{cases} \end{cases}$$

where $d_0 \approx 1.543679$ is the unique real root of the polynomial $d^3 - 2d^2 + 2d - 2$.

We take a closer look at two special cases:

(i) If $k_{\min} = \omega_{\min} = 0$, all three cases for A in (7) coincide, since $\bar{k} = 0$, and the constant A simplifies to

$$A = 4x_0, \tag{8}$$

and we find the special case analyzed in [11].

(ii) For the heat equation, $a = 0$, $b = 0$, $c = 0$, $\nu = 1$, and if $k_{\min}$ and $\omega_{\min}$ do not both vanish simultaneously, we also obtain $\bar{k} = 0$, and

$$A = 4\sqrt{2\left(\sqrt{k_{\min}^4+\omega_{\min}^2}+k_{\min}^2\right)},$$

the special case analyzed in [14].

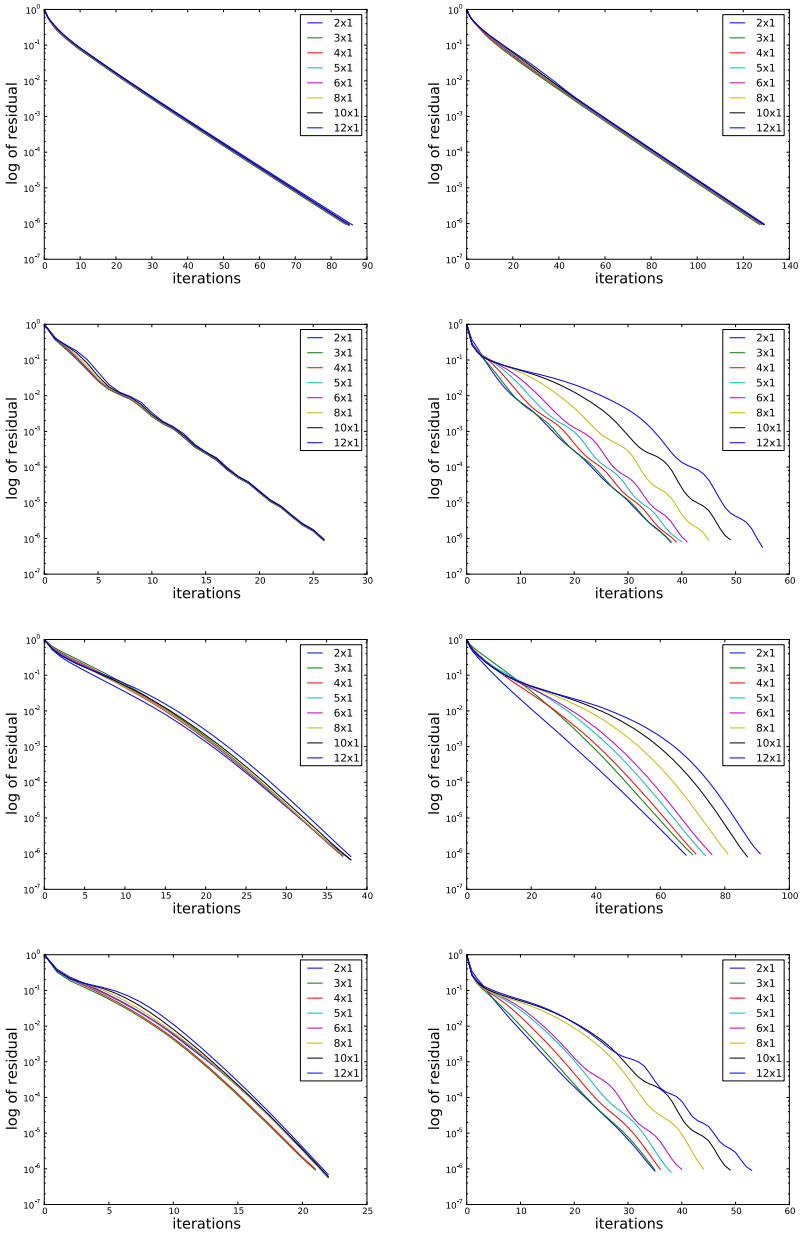


Fig. 1. Convergence behavior with the theoretically optimized parameter p^* in the top four pictures, and the numerically optimized one below: for $T = \frac{1}{20}$ on the *left* and $T = 1$ on the *right* using Schwarz as an iterative method in the first and third row, and as preconditioner for GMRES in the second and fourth row.

4 Numerical Experiments

We discretize (1) with $\mathbf{a} := (1,1)$, $\nu = 0.1$ and $c = 0$ on $\Omega = (0,1.2) \times (0,1.2)$ using P1 finite elements on a regular triangular mesh with $h = \frac{1}{100}$, and backward Euler with $\Delta t = \frac{1}{400}$. We simulate directly the homogeneous error equations, start the iteration with a random initial guess, and stop when the residual becomes smaller than 10^{-6}. We do not use a coarse grid.

Table 1. Number of iterations for $T = 1$ using the old ($p^* = 3.77$) and new ($p^* = 8.42$) theoretically optimized parameters compared to the best choice (given in parentheses after the iteration number) obtained by numerical minimization of the iteration number.

	Iterative			GMRES		
Decomposition	Old	New	Numerical	Old	New	Numerical
2×1	271	131	70 (16.55)	49	38	35 (13.42)
2×2	272	132	81 (14.05)	49	40	38 (13.42)
4×1	270	130	73 (15.30)	49	39	36 (12.80)
8×1	271	131	83 (13.58)	51	45	44 (10.92)
4×4	272	131	91 (12.48)	50	44	44 (8.42)
8×8	274	132	109 (10.30)	58	59	56 (5.77)

Table 1 shows iteration numbers for the new optimized parameter p^* from Theorem 1, compared to the old one from [11] given in (8), and the parameters which work numerically best (found using a multi directional simplex method, see [13]). We provide iteration numbers both for the algorithm used as an iterative solver, and as a preconditioner for GMRES.

The left column gives the type of decomposition: 8×1 means for example a one dimensional, banded decomposition into 8 subdomains. Clearly the new estimate of p^* leads to a significantly better method, and the iteration number is now much closer to the best possible in the algorithm. We also see that the numerically best parameter depends on the number of subdomains, a fact which our analysis based on two subdomains cannot capture. The iteration number then grows with the number of subdomains for larger time windows without coarse grid, as in the case of the classical Schwarz waveform relaxation algorithm, see [9].

Next, we present a large scale study in Table 2, where Ω is decomposed into bands, 2×1, 3×1, $4 \times 1 \ldots$, for various length of the time interval.

We observe how the convergence is independent of the number of subdomains for the theoretically optimized parameter, a result that has been proved for classical Schwarz waveform relaxation methods over short time intervals in [9], see also [3], but no analytical results exist so far for optimized Schwarz methods. We also see again that the new optimized parameter performs as well as the numerically optimized one when GMRES is used. We show in Fig. 1 the convergence curves of the Schwarz waveform relaxation algorithm used as an iterative solver and as a preconditioner for GMRES, both for the theoretically and numerically optimized parameter. We see that the algorithm's convergence behavior does not depend on the number of

Table 2. Number of iterations for different final times T and different numbers of subdomains, both using the theoretical and numerically optimal p^*, which was $p^* \in \{8.4, 8.4, 9.3, 10.7, 12.6\}$ for $T \in \{1, \frac{1}{2}, \frac{1}{5}, \frac{1}{10}, \frac{1}{20}\}$, and the value of the numerically optimal p^*.

	Iterative solver								Preconditioner for GMRES							
	Iteration number using the theoretical p^*															
T	**2×1**	**3×1**	**4×1**	**5×1**	**6×1**	**8×1**	**10×1**	**12×1**	**2×1**	**3×1**	**4×1**	**5×1**	**6×1**	**8×1**	**10×1**	**12×1**
1	131	129	130	130	130	131	131	131	38	38	39	40	41	45	49	55
1/2	131	129	129	130	130	130	130	131	36	36	36	37	37	38	40	43
1/5	119	117	118	118	119	119	119	119	33	33	33	33	33	33	33	34
1/10	103	102	103	103	103	103	103	104	29	29	29	30	30	30	30	30
1/20	88	86	87	87	87	87	87	87	26	26	26	26	26	26	26	26
	Iteration number using the numerical p^*															
T	**2×1**	**3×1**	**4×1**	**5×1**	**6×1**	**8×1**	**10×1**	**12×1**	**2×1**	**3×1**	**4×1**	**5×1**	**6×1**	**8×1**	**10×1**	**12×1**
1	70	72	73	76	78	83	89	93	35	35	36	38	40	44	49	53
1/2	63	63	63	64	66	68	72	75	33	32	33	33	34	36	39	42
1/5	52	52	52	52	53	54	55	56	28	28	28	28	28	29	30	31
1/10	45	45	45	45	45	46	46	47	25	25	25	25	25	25	25	26
1/20	40	39	39	39	39	39	40	40	22	21	21	22	22	22	22	22
	Value of the numerical p^*															
T	**2×1**	**3×1**	**4×1**	**5×1**	**6×1**	**8×1**	**10×1**	**12×1**	**2×1**	**3×1**	**4×1**	**5×1**	**6×1**	**8×1**	**10×1**	**12×1**
1	16.4	15.4	15.4	14.9	14.4	13.7	12.9	12.0	13.4	13.4	12.8	12.2	9.7	10.9	8.4	5.9
1/2	18.4	17.9	17.9	17.7	17.4	16.7	15.9	15.4	18.4	17.8	18.4	17.2	17.2	14.1	13.4	10.9
1/5	22.3	22.3	22.3	22.3	22.3	22.3	21.3	20.3	21.8	21.8	21.8	21.8	21.8	19.3	19.3	21.8
1/10	25.7	25.7	25.7	25.7	25.7	25.7	25.7	24.7	25.7	25.7	25.7	25.7	25.7	25.7	25.7	20.7
1/20	29.6	29.6	29.6	29.6	29.6	29.6	29.6	32.6	32.6	30.1	30.1	32.6	32.6	32.6	32.6	32.6

subdomains over the short time interval. We also observe that the numerically optimized parameter leads to a superlinear convergence regime, while the theoretical one gives a linear convergence regime.

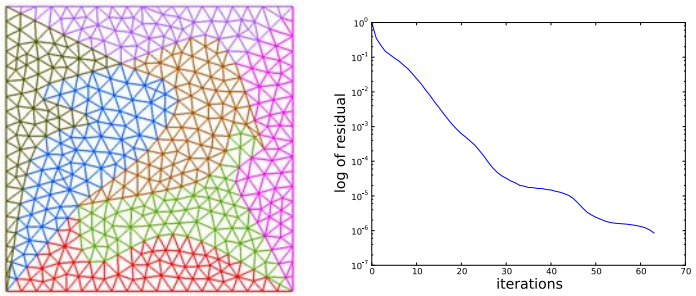


Fig. 2. Challenging geometrical decomposition on the *left*, and convergence curve on the *right*.

We next turn to the case of decompositions with cross points, where the square domain Ω is decomposed into smaller square or rectangular subdomains, 2×2, 3×2,

Table 3. Number of iterations for the theoretical optimized parameter, and in parentheses for the numerically optimized one for different final times T and different decompositions with cross points.

T	Iterative solver						Preconditioner for GMRES					
	$\times$	**2**	**3**	**4**	**5**	**6**	$\times$	**2**	**3**	**4**	**5**	**6**
$\frac{1}{20}$	**2**	88(45)					**2**	28(25)				
	3	87(45)	88(45)				**3**	28(25)	28(25)			
	4	88(45)	87(45)	87(46)			**4**	28(25)	29(26)	29(25)		
	5	88(45)	88(46)	88(46)	88(46)		**5**	28(25)	29(26)	29(26)	29(25)	
	6	88(45)	87(46)	88(46)	88(46)	88(46)	**6**	28(25)	29(26)	29(26)	29(25)	29(25)
$\frac{1}{5}$	**2**	120(60)					**2**	35(32)				
	3	119(60)	119(61)				**3**	35(32)	35(32)			
	4	119(61)	119(61)	120(61)			**4**	35(32)	35(32)	35(32)		
	5	119(61)	119(61)	120(62)	120(62)		**5**	35(32)	35(32)	35(32)	35(32)	
	6	119(62)	119(62)	119(62)	120(63)	120(64)	**6**	35(33)	35(33)	35(32)	35(33)	35(33)
1	**2**	132(81)					**2**	40(38)				
	3	131(84)	131(86)				**3**	41(40)	42(41)			
	4	131(86)	131(89)	131(91)			**4**	42(41)	43(43)	44(44)		
	5	131(88)	131(91)	131(93)	131(96)		**5**	43(43)	44(44)	46(45)	48(47)	
	6	131(91)	131(93)	131(96)	131(98)	131(100)	**6**	45(45)	46(45)	48(47)	50(49)	52(51)

$3 \times 3, \ldots$. The results are given in Table 3. The algorithm performs similarly to the banded case, but an interesting new observation is that the number of iterations over the long time interval is constant over anti-diagonals in the table (notice the numerically optimized case in particular), which shows that the diameter of the graph of the decomposition is relevant for the dependence on the number of subdomains over long times.

We finally show a numerical experiment for $T = 1$ on the geometrically challenging decomposition shown in Fig. 2 on the left, which our generic simulator, implemented in Python using MPI, can easily handle. The computational mesh is twice refined from the mesh shown, and the convergence history with GMRES is shown in Fig. 2 on the right.

5 Conclusions

We presented new theoretical estimates for the parameters in the optimized Schwarz waveform relaxation algorithm. Our large scale numerical study shows that the new parameters perform significantly better than the old ones, and they reveal properties of the algorithm which are not yet understood theoretically, like the good scaling properties when the number of subdomains is increased, or the dependence of the parameter on the number of subdomains.

Acknowledgments This work was partially supported by the Swiss SNF grant 200020-121561/1, and by the French ANR projects COMMA and SHP-CO2.

References

1. D. Bennequin, M.J. Gander, and L. Halpern. A homographic best approximation problem with application to optimized Schwarz waveform relaxation. *Math. Comput.*, 78(265): 185–232, 2009.
2. M. Bjørhus. *On Domain Decomposition, Subdomain Iteration and Waveform Relaxation*. PhD thesis, University of Trondheim, Norway, 1995.
3. D.S. Daoud and M.J. Gander. Overlapping Schwarz waveform relaxation for advection reaction diffusion problems. *Bol. Soc. Esp. Mat. Apl.*, 46:75–90, 2009.
4. M.J. Gander. Optimized Schwarz methods. *SIAM J. Numer. Anal.*, 44(2):699–731, 2006.
5. M.J. Gander. On the influence of geometry on schwarz methods. These Proceedings. Springer Berlin, Heidelberg, New York, 2010.
6. M.J. Gander and L. Halpern. Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems. *SIAM J. Numer. Anal.*, 45(2):666–697, 2007.
7. M.J. Gander, L. Halpern, and F. Nataf. Optimal convergence for overlapping and non-overlapping Schwarz waveform relaxation. In C.-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, editors, *Eleventh International Conference of Domain Decomposition Methods*. ddm.org, 1999.
8. M.J. Gander and A.M. Stuart. Space time continuous analysis of waveform relaxation for the heat equation. *SIAM J. Sci. Comput.*, 19:2014–2031, 1998.
9. M.J. Gander and H. Zhao. Overlapping Schwarz waveform relaxation for the heat equation in n-dimensions. *BIT*, 42(4):779–795, 2002.
10. E. Giladi and H.B. Keller. Space time domain decomposition for parabolic problems. *Numer. Math.*, 93(2):279–313, 2002.
11. L. Halpern. Optimized Schwarz waveform relaxation: Roots, blossoms and fruits. In *Eighteenth International Conference of Domain Decomposition Methods*. Springer, 2009.
12. V. Martin. An optimized Schwarz waveform relaxation method for unsteady convection diffusion equation. *Appl. Numer. Math.*, 52(4):401–428, 2005.
13. V. Torczon. *Multi-Directional Search: A Direct Search Algorithm for Parallel Machines*. PhD thesis, Department of Computational and Applied Mathematics, Rice University, Houston, 1990.
14. B.M. Tran. *Schwarz Waveform Relaxation Methods*. PhD thesis, LAGA, Paris 13, 2010. In preparation.

Optimized Schwarz Methods for Maxwell's Equations with Non-zero Electric Conductivity

Victorita Dolean[1], Mohamed El Bouajaji[2], Martin J. Gander[3], Stéphane Lanteri[2]

[1] Laboratoire J.A. Dieudonné, University de Nice Sophia-Antipolis, CNRS UMR 6621, F-06108 Nice Cedex, France, dolean@unice.fr
[2] NACHOS project-team, INRIA Sophia Antipolis - Méditerranée research center, F-06902 Sophia Antipolis Cedex, France, Stephane.Lanteri@inria.fr
[3] Mathematics Section, University of Geneva, CH-1211, Geneva, Switzerland, martin.gander@unige.ch

1 Introduction

The study of optimized Schwarz methods for Maxwell's equations started with the Helmholtz equation, see [2, 3, 4, 11]. For the rot-rot formulation of Maxwell's equations, optimized Schwarz methods were developed in [1], and for the more general form in [9, 10]. An entire hierarchy of families of optimized Schwarz methods was analyzed in [8], see also [5] for discontinuous Galerkin discretizations and large scale experiments. We present in this paper a first analysis of optimized Schwarz methods for Maxwell's equations with non-zero electric conductivity. This is an important case for real applications, and requires a new, and fundamentally different optimization of the transmission conditions. We illustrate our analysis with numerical experiments.

2 Schwarz Methods for Maxwell's Equations

The time dependent Maxwell equations are

$$-\varepsilon\frac{\partial \boldsymbol{\mathcal{E}}}{\partial t} + \text{curl}\, \boldsymbol{\mathcal{H}} - \sigma \boldsymbol{\mathcal{E}} = \mathbf{J}, \qquad \mu\frac{\partial \boldsymbol{\mathcal{H}}}{\partial t} + \text{curl}\, \boldsymbol{\mathcal{E}} = 0, \tag{1}$$

where $\boldsymbol{\mathcal{E}} = (\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3)^T$ and $\boldsymbol{\mathcal{H}} = (\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3)^T$ denote the electric and magnetic fields, respectively, ε is the *electric permittivity*, μ is the *magnetic permeability*, σ is the *electric conductivity* and $\boldsymbol{\mathcal{J}}$ is the applied current density. We assume the applied current density to be divergence free, $\text{div}\boldsymbol{\mathcal{J}} = 0$.

One can show, see for example [8] for the context of domain decomposition methods, that the time dependent Maxwell equations (1) are a system of hyperbolic partial differential equations. This hyperbolic system has for any interface two incoming and two outgoing characteristics. Imposing incoming characteristics is equivalent to imposing the impedance condition

$$\mathcal{B}_{\mathbf{n}}(\boldsymbol{\mathcal{E}},\boldsymbol{\mathcal{H}}) := \mathbf{n}\times\frac{\boldsymbol{\mathcal{E}}}{Z}+\mathbf{n}\times(\boldsymbol{\mathcal{H}}\times\mathbf{n}) = \mathbf{s}. \tag{2}$$

We consider in this paper the time-harmonic Maxwell equations,

$$-i\omega\varepsilon\mathbf{E}+\text{curl }\mathbf{H}-\sigma\mathbf{E}=\mathbf{J}, \qquad i\omega\mu\mathbf{H}+\text{curl }\mathbf{E}=\mathbf{0}. \tag{3}$$

A family of Schwarz methods for (3) with a possibly non-overlapping decomposition of the domain Ω into Ω_1 and Ω_2, with interfaces $\Gamma_{12} := \partial\Omega_1\cap\Omega_2$ and $\Gamma_{21} := \partial\Omega_2\cap\Omega_1$, is given by

$$\begin{aligned}
-i\omega\varepsilon\mathbf{E}^{1,n}+\text{curl }\mathbf{H}^{1,n}-\sigma\mathbf{E}^{1,n} &= \mathbf{J} && \text{in } \Omega_1,\\
i\omega\mu\mathbf{H}^{1,n}+\text{curl }\mathbf{E}^{1,n} &= \mathbf{0} && \text{in } \Omega_1,\\
(\mathcal{B}_{\mathbf{n}_1}+\mathcal{S}_1\mathcal{B}_{\mathbf{n}_2})(\mathbf{E}^{1,n},\mathbf{H}^{1,n}) &= (\mathcal{B}_{\mathbf{n}_1}+\mathcal{S}_1\mathcal{B}_{\mathbf{n}_2})(\mathbf{E}^{2,n-1},\mathbf{H}^{2,n-1}) && \text{on } \Gamma_{12},\\
-i\omega\varepsilon\mathbf{E}^{2,n}+\text{curl }\mathbf{H}^{2,n}-\sigma\mathbf{E}^{2,n} &= \mathbf{J} && \text{in } \Omega_2,\\
i\omega\mu\mathbf{H}^{2,n}+\text{curl }\mathbf{E}^{2,n} &= \mathbf{0} && \text{in } \Omega_2,\\
(\mathcal{B}_{\mathbf{n}_2}+\mathcal{S}_2\mathcal{B}_{\mathbf{n}_1})(\mathbf{E}^{2,n},\mathbf{H}^{2,n}) &= (\mathcal{B}_{\mathbf{n}_2}+\mathcal{S}_2\mathcal{B}_{\mathbf{n}_1})(\mathbf{E}^{1,n-1},\mathbf{H}^{1,n-1}) && \text{on } \Gamma_{21},
\end{aligned} \tag{4}$$

where $\mathcal{S}_j$, $j=1,2$ are tangential, possibly pseudo-differential operators. Different choices of $\mathcal{S}_j$, $j=1,2$ lead to different parallel solvers for Maxwell's equations, see [8]. The classical Schwarz method is exchanging characteristic information at the interfaces between subdomains, which means $\mathcal{S}_j=0$, $j=1,2$. For the case of constant coefficients and the domain $\Omega=\mathbb{R}^3$, with the Silver-Müller radiation condition

$$\lim_{r\to\infty} r\,(\mathbf{H}\times\mathbf{n}-\mathbf{E}) = 0, \tag{5}$$

and the two subdomains

$$\Omega_1=(0,\infty)\times\mathbb{R}^2, \quad \Omega_2=(-\infty,L)\times\mathbb{R}^2, \qquad L\geq 0, \tag{6}$$

the following convergence result was obtained in [8] using Fourier analysis:

Theorem 1. *For any* $(\mathbf{E}^{1,0};\mathbf{H}^{1,0})\in(L^2(\Omega_1))^6$, $(\mathbf{E}^{2,0};\mathbf{H}^{2,0})\in(L^2(\Omega_2))^6$, *the classical algorithm with* $\sigma>0$ *converges in* $(L^2(\Omega_1))^6\times(L^2(\Omega_2))^6$. *The convergence factor for each Fourier mode* $\mathbf{k}:=(k_y,k_z)$ *with* $|\mathbf{k}|^2:=k_y^2+k_z^2$ *is*

$$\rho_{cla}(\mathbf{k},\tilde{\omega},\sigma,Z,L) = \left|\frac{\sqrt{|\mathbf{k}|^2-\tilde{\omega}^2+i\tilde{\omega}\sigma Z}-i\tilde{\omega}}{\sqrt{|\mathbf{k}|^2-\tilde{\omega}^2+i\tilde{\omega}\sigma Z}+i\tilde{\omega}}e^{-\sqrt{|\mathbf{k}|^2-\tilde{\omega}^2+i\tilde{\omega}\sigma Z}L}\right|,$$

where $\tilde{\omega}:=\omega\sqrt{\varepsilon\mu}$, *and* $Z:=\sqrt{\frac{\mu}{\varepsilon}}$.

This result shows that if $\sigma>0$, the method converges, also without overlap, $L=0$, which is unusual for classical Schwarz methods, but normal for optimized ones, for an explanation, see [6]. If however the electric conductivity $\sigma=0$, then for $|\mathbf{k}|^2=\tilde{\omega}^2$ the convergence factor equals 1, and the method is stagnating for this frequency, and thus by continuity slow for nearby frequencies. In addition, if there is no overlap, $L=0$, we have $\rho_{cla}(\mathbf{k})<1$ only for the propagative modes, $|\mathbf{k}|^2<\tilde{\omega}^2$, and $\rho_{cla}(\mathbf{k})=1$ for evanescent modes, i.e. when $|\mathbf{k}|^2\geq\tilde{\omega}^2$; the method is now stagnating for all evanescent modes. Hence for $\sigma=0$, better transmission conditions were developed in [8]. The analysis in [8] does however not apply if the electric conductivity $\sigma>0$.

3 Analysis for Non-zero Electric Conductivity

We present now an analysis of algorithm (4), (6) for the case where the electric conductivity is non-zero, $\sigma > 0$, in the special case of the two dimensional transverse magnetic Maxwell equations. For these equations, the unknowns are independent of z, and we have $\mathbf{E} = (0, 0, E_z)$ and $\mathbf{H} = (H_x, H_y, 0)$. The results are again based on Fourier transforms, here in the y direction with Fourier variable k.

Theorem 2. *For $\sigma > 0$, if $\mathcal{S}_j$, $j = 1, 2$ have the constant Fourier symbol*

$$\sigma_j = \mathcal{F}(\mathcal{S}_j) = -\frac{s - i\tilde{\omega}}{s + i\tilde{\omega}}, \qquad s \in \mathbb{C}, \tag{7}$$

then the optimized Schwarz method (4), (6) has the convergence factor

$$\rho_\sigma(\tilde{\omega}, Z, \sigma, L, k, s) = \left| \left(\frac{\sqrt{k^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z} - s}{\sqrt{k^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z} + s} \right) e^{-\sqrt{k^2 - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z} L} \right|. \tag{8}$$

Proof. Taking a Fourier transform in the y variable of (4) with $\mathbf{J} = 0$, the so-called error equations, we get

$$\partial_x \begin{pmatrix} \hat{E}_z^{j,n} \\ \hat{H}_y^{j,n} \end{pmatrix} = \begin{pmatrix} 0 & i\omega\mu \\ \frac{k^2 - \omega^2 \varepsilon\mu + i\omega\mu\sigma}{i\omega\mu} & 0 \end{pmatrix} \begin{pmatrix} \hat{E}_z^{j,n} \\ \hat{H}_y^{j,n} \end{pmatrix} =: M \begin{pmatrix} \hat{E}_z^{j,n} \\ \hat{H}_y^{j,n} \end{pmatrix}, \; j = 1, 2. \tag{9}$$

The eigenvalues of the matrix M, and their corresponding eigenvectors are

$$\lambda_j = \pm\lambda = \pm\sqrt{k - \tilde{\omega}^2 + i\tilde{\omega}\sigma Z}, \quad \mathbf{v}_j = \begin{pmatrix} \mp\frac{i\omega\mu}{\lambda} \\ 1 \end{pmatrix}, \quad j = 1, 2, \tag{10}$$

and therefore the solutions of (9) are given by

$$\left(\hat{E}_z^{1,n}, \hat{H}_y^{1,n}\right) = \alpha_1^n \mathbf{v}_1 e^{\lambda x} + \alpha_2^n \mathbf{v}_2 e^{-\lambda x}, \; \left(\hat{E}_z^{2,n}, \hat{H}_y^{2,n}\right) = \beta_1^n \mathbf{v}_1 e^{\lambda x} + \beta_2^n \mathbf{v}_2 e^{-\lambda x}.$$

Using the Silver-Müller radiation condition (5), we have $\alpha_2^n = \beta_1^n = 0$, and inserting the solutions into the interface conditions in (4), we get

$$\alpha_1^n = A\beta_2^{n-1} e^{-\lambda L}, \quad \beta_2^n = A\alpha_1^{n-1} e^{-\lambda L}, \text{ with } A := \frac{\lambda - s}{\lambda + s},$$

and the definition $\rho_\sigma(\tilde{\omega}, Z, \sigma, L, k, s) := \left| \frac{\alpha_1^n}{\alpha_1^{n-2}} \right|^{\frac{1}{2}}$ leads to the result (8).

In a numerical implementation, the range of frequencies is bounded, $k \in K := [k_{min}, k_{max}]$, where the minimum frequency $k_{\min} > 0$ is a constant depending on the geometry, and the maximum numerical frequency that can be represented on a mesh is $k_{max} = \frac{C}{h}$ where C is a constant. From Theorem 2, we can immediately get a convergence result for the classical Schwarz method that uses characteristic transmission conditions.

Corollary 1. *For $\sigma > 0$, in the case of the classical Schwarz method, $\sigma_j = 0$, $j = 1, 2$, the asymptotic convergence factor for small mesh size h is*

$$\bar{\rho}_\sigma := \max_{k\in K}(\rho_\sigma) = \begin{cases} 1 - \frac{4}{3}\left(9\omega^4\sigma^2\mu^3\varepsilon C_L^6\right)^{\frac{1}{8}} h^{\frac{3}{4}} + O(h^{\frac{5}{4}}), & L = C_L h, \\ 1 - \frac{\omega^2\sigma\sqrt{\mu^3\varepsilon}}{C^3}h^3 + O(h^5), & L = 0. \end{cases} \tag{11}$$

Proof. The proof is obtained by inserting $s = i\tilde{\omega}$ into (8), and then expanding the maximum of ρ_σ over $k \in K$ for h small.

In order to obtain a more efficient algorithm, we choose σ_j, $j = 1, 2$ such that ρ_σ is minimal over the range of frequencies $k \in K$. We look for s of the form $s = p(1+i)$, such that p is solution of the min-max problem

$$\min_{p\geq 0}\left(\max_{k\in K}\rho_\sigma(\tilde{\omega}, Z, \sigma, L, k, p(1+i))\right). \tag{12}$$

Theorem 3. *For $\sigma > 0$, and the non-overlapping case, $L = 0$, the solution of the minmax problem (12) is for h sufficiently small given by*

$$p^* = \frac{(\omega\sigma\mu)^{\frac{1}{4}}\sqrt{C}}{2^{\frac{1}{4}}\sqrt{h}} \quad \textit{and} \quad \rho_\sigma^* = 1 - \frac{2^{\frac{3}{4}}(\omega\sigma\mu)^{\frac{1}{4}}\sqrt{h}}{\sqrt{C}} + O(h). \tag{13}$$

Proof. We assume that $p \geq p_c := \sqrt{\frac{3\sigma\omega\mu}{2}}$, a hypothesis that can be removed with an additional analysis, which is too long however for this short paper. Using the change of variables $\xi(k) := \mathbb{R}\left(\sqrt{k^2 - \omega^2\epsilon\mu + i\sigma\omega\mu}\right)$ and $y := \sigma\omega\mu$, the convergence factor simplifies to

$$\rho_\sigma(\tilde{\omega}, Z, \sigma, 0, k, p(1+i)) = \sqrt{\frac{4\xi^2(\xi-p)^2 + (y-2\xi p)^2}{4\xi^2(\xi+p)^2 + (y+2\xi p)^2}} =: R(\xi, y, p).$$

Since the mapping $k \mapsto \xi(k)$ is increasing in k for $k \geq 0$, we have

$$\min_{p\geq 0}\left(\max_{k_{min}\leq k\leq k_{max}}\rho_\sigma(\tilde{\omega}, Z, \sigma, 0, k, p(1+i))\right) = \min_{p\geq 0}\left(\max_{\xi_0\leq\xi\leq\xi_{max}} R(\xi, y, p)\right),$$

where $\xi_0 = \xi(k_{min})$ and $\xi_{max} = \xi(k_{max})$. We start by studying the variation of R for fixed p; the polynomial

$$P(\xi) = (\xi^2 - \frac{1}{2}y)(8\xi^4 + 16\xi^2(y - p^2) + 2y^2)$$

is the numerator of the partial derivative of R with respect to ξ. P has at most three positive roots, and $\xi_2 = \sqrt{\frac{y}{2}}$ is always a root. We now show that for $p \geq p_c$, ξ_2 is a maximum and the other two roots of P cannot be maxima. The second partial derivative of R with respect to ξ evaluated at ξ_2 is

$$\frac{\partial^2 R}{\partial \xi^2}(\xi_2, y, p) = -4\sqrt{\frac{2}{y}}\frac{(2p^2 - 3y)p}{|\sqrt{2}p - \sqrt{y}|(\sqrt{2}p + \sqrt{y})^3}.$$

Since for $p \geq p_c$ by assumption, $\frac{\partial^2 R}{\partial \xi^2}(\xi_2, y, p) \leq 0$, ξ_2 is a local maximum. Since $R(\xi_0, y, p) \leq 1$ and $\lim_{\xi \to +\infty} R(\xi, y, p) = 1$, the other two roots of P can not be maxima if $p \geq p_c$. Therefore the maximum of R is either at ξ_0, ξ_2 or ξ_{max}. But we also find that $R(\xi_0, y, p) \leq R(\xi_2, y, p)$ for $p \geq p_c$, which excludes ξ_0 as a candidate for the maximum. Moreover, for ξ_{max} sufficiently large we have $R(\xi_2, y, p_c) \leq R(\xi_{max}, y, p_c)$, and for p large, we have $R(\xi_2, y, p) \geq R(\xi_{max}, y, p)$. By continuity, there exists at least one p^* such that $R(\xi_2, y, p^*) = R(\xi_{max}, y, p^*)$. Moreover, the computation of

$$\frac{\partial R}{\partial p}(\xi, y, p) = -\frac{1}{R(\xi, y, p)}\frac{(p^2 - \frac{1}{8}\frac{4\xi^4 + y^2}{\xi^2})}{(4\xi^4 + 8\xi^3 p + 8\xi^2 p^2 + y^2 + 4y\xi p)^2}$$

shows that the function $p \mapsto R(\xi_2, y, p)$ is monotonically increasing, and $p \mapsto R(\xi_{max}, y, p)$ is monotonically decreasing for ξ_{max} sufficiently large. Hence p^* is unique and therefore the unique solution of the min-max problem. As we have

$$R(\xi_2, y, p^*) = R(\xi_{max}, y, p^*) \Longrightarrow p^* = \frac{\sqrt[4]{2y}\sqrt{\xi_{\max}\left(y + 2{\xi_{\max}}^2 + \xi_{\max}\sqrt{2}\sqrt{y}\right)}}{2\xi_{\max}},$$

by expanding p^* for h small, $(\xi_{max} = \xi(C/h))$, we get the desired result.

A numerical example of the convergence factor is shown in Fig. 1. We can see that for $\sigma > 0$, the classical Schwarz algorithm does not have convergence problems any more close to the resonance frequency. Increasing σ further improves the performance, since the maximum of the convergence factor decreases. We also see that the optimization, which is based on equioscillation, leads to a uniformly small contraction factor when $\sigma > 0$, whereas in the case $\sigma = 0$ a small region close to the resonance frequency needs to be excluded in order to minimized the convergence factor for the remaining frequencies.

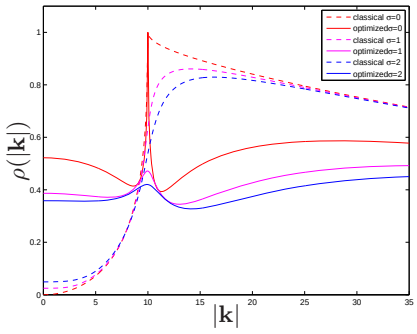
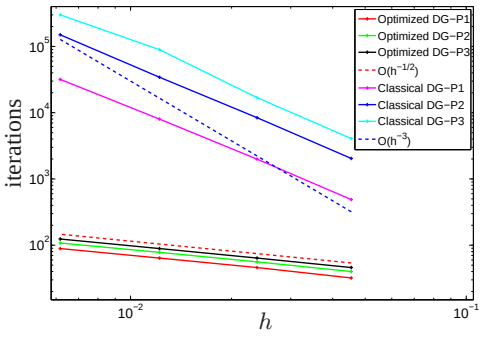


Fig. 1. Convergence factors as function of the frequency parameter $|\mathbf{k}|$.

4 Numerical Results

We present now some numerical tests in order to illustrate the performance of the algorithms. The domain Ω is partitioned into several subdomains Ω_j. In each subdomain, we use a discontinuous Galerkin method (DG), see [5].

We first test the propagation of a plane wave in a homogeneous medium. The domain is $\Omega = (0,1)^2$, and the parameters are constant in Ω, with $\varepsilon = \mu = 1$, $\sigma = 5$ and $\omega = 2\pi$. We impose on the boundary an incident field $\mathbf{W}^{inc} = (H_x^{inc}, H_y^{inc}, E_z^{inc}) = (\frac{k_y}{\mu\omega}, \frac{-k_x}{\mu\omega}, 1)e^{-i\mathbf{k}\cdot\mathbf{x}}$ with $\mathbf{k} = (k_x, k_y) = (\omega\sqrt{\varepsilon - i\frac{\sigma}{\omega}}, 0)$, $\mathbf{x} = (x, y)$. The domain Ω is decomposed into two subdomains $\Omega_1 = (0, 1/2) \times (0, 1)$ and $\Omega_2 = (1/2, 1) \times (0, 1)$. For this test case, the DG method is used with a uniform polynomial approximation of order one, two and three, denoted by DG-$P1$, DG-$P2$ and DG-$P3$. The performance of the algorithm is shown in Fig. 2.



h	$\frac{1}{10}$	$\frac{1}{20}$	$\frac{1}{40}$	$\frac{1}{80}$
Optimized Schwarz algorithm :				
DG-$P1$	32	46	64	89
DG-$P2$	40	56	78	108
DG-$P3$	46	64	89	124
Classical Schwarz algorithm :				
DG-$P1$	488	1990	8000	31821
DG-$P2$	2038	8419	34262	15E04
DG-$P3$	4066	16875	89486	30E04

Fig. 2. Number of iterations against the mesh size h, to attain a relative residual reduction of 10^{-8} obtained with the classical and optimized Schwarz algorithm.

These results are in good agreement with the theoretical result in Theorem 3: the curves fit nicely the dependence on h predicted, i.e they behave like $h^{-0.5}$. We also see the tremendous improvement of the optimized Schwarz method over the classical Schwarz method, which nevertheless performs a bit better than predicted in Corollary 1, the dependence on h measured is $O(h^{-2})$, instead of $O(h^{-3})$; for an explanation, see [7].

The second test problem is a simplified model of the propagation of an electromagnetic wave, emitted by a localized source, in the head tissues. The geometric configuration is given in Fig. 3.

The electromagnetic parameters of the material in the head tissues are: $\mu = 1$ in the whole domain, $\varepsilon = 43.85$, $\sigma = 1.23 \cdot 120\pi$ for the skin, $\varepsilon = 15.56$, $\sigma = 0.43 \cdot 120\pi$ for the skull, $\varepsilon = 67.20$, $\sigma = 2.92 \cdot 120\pi$ for the cerebrospinal fluid and $\varepsilon = 43.55$, $\sigma = 1.15 \cdot 120\pi$ for the brain. The antenna is modeled by two perfectly conducting rods (with base section of 0.25^2cm^2) and between these rods a current density J_z is applied. The computational domain is decomposed into several

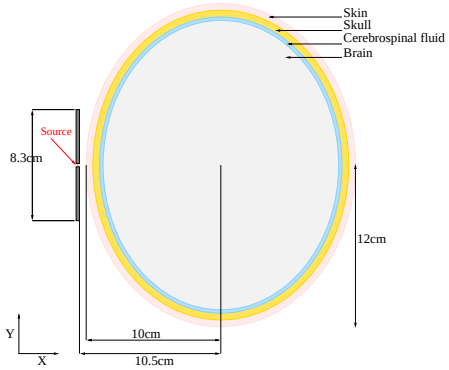


Fig. 3. Model of the different layers of a skull.

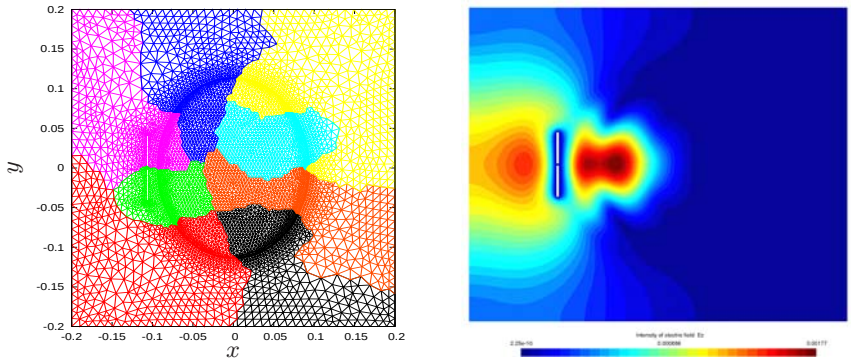


Fig. 4. Decomposition into subdomains and solution.

subdomains (a decomposition into eight subdomains is shown for example in Fig. 4 on the left).

We compare in this test the performance of the classical Schwarz and the new optimized Schwarz algorithm for a decomposition into two, four, eight and sixteen subdomains. In Table 1,

we show the number of iterations needed for convergence, i.e to attain a relative residual of 10^{-8}, depending on the number of subdomains. These results show that the optimized Schwarz algorithm converges much faster than the classical Schwarz algorithm. Here we used a Krylov method (BiCGStab) for the solution of the linear system, preconditioned with the classical and optimized Schwarz preconditioner.

Table 1. Iteration number comparison for the cell phone antenna problem.

Number of subdomains	2	4	8	16
Classical Schwarz	94	197	179	174
Optimized Schwarz	69	92	82	85

5 Conclusion

We analyzed an optimized Schwarz method for the two dimensional Maxwell equations with non-zero electric conductivity. The new method performs much better than the classical one, and our theoretical results are well confirmed by the numerical experiments presented, also for a non-trivial test case.

References

1. A. Alonso-Rodriguez and L. Gerardo-Giorda. New nonoverlapping domain decomposition methods for the harmonic Maxwell system. *SIAM J. Sci. Comput.*, 28(1):102–122, 2006.
2. P. Chevalier and F. Nataf. An OO2 (Optimized Order 2) method for the Helmholtz and Maxwell equations. In *Tenth International Conference on Domain Decomposition Methods in Science and in Engineering*, pp. 400–407, AMS, Providence, RI, 1997.
3. B. Després. Décomposition de domaine et problème de Helmholtz. *C.R. Acad. Sci. Paris*, 1(6):313–316, 1990.
4. B. Després, P. Joly, and J.E. Roberts. A domain decomposition method for the harmonic Maxwell equations. In *Iterative methods in linear algebra*, pp. 475–484. North-Holland, Amsterdam, 1992.
5. H. Yanping, R. Kornhuber, O. Widlund, and J. Xu (eds.). Domain Decomposition in Science and Engineering XIX, Springer Verlag, Berlin Heidelberg, p. 15–25, 2011.
6. V. Dolean and M.J. Gander. Why classical Schwarz methods applied to hyperbolic systems can converge even without overlap. In *Domain Decomposition Methods in Science and Engineering XVIII*, pp. 467–476. Springer, 2007.
7. H. Yanping, R. Kornhuber, O. Widlund, and J. Xu (eds.). Domain Decomposition in Science and Engineering XIX, Springer Verlag, Berlin Heidelberg, p. 117–124, 2011.
8. V. Dolean, L. Gerardo-Giorda, and M.J. Gander. Optimized Schwarz methods for Maxwell equations. *SIAM J. Sci. Comput.*, 31(3):2193–2213, 2009.
9. V. Dolean, S. Lanteri, and R. Perrussel. A domain decomposition method for solving the three-dimensional time-harmonic Maxwell equations discretized by discontinuous Galerkin methods. *J. Comput. Phys.*, 227(3):2044–2072, 2008.
10. V. Dolean, S. Lanteri, and R. Perrussel. Optimized Schwarz algorithms for solving time-harmonic Maxwell's equations discretized by a discontinuous Galerkin method. *IEEE. Trans. Magn.*, 44(6):954–957, 2008.
11. M.J. Gander, F. Magoulès, and F. Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24(1):38–60, 2002.

Robust Boundary Element Domain Decomposition Solvers in Acoustics

Olaf Steinbach and Markus Windisch

Institute of Computational Mathematics, TU Graz, A 8010 Graz, Austria,
o.steinbach@tugraz.at; markus.windisch@tugraz.at

Summary. A stable boundary element tearing and interconnecting domain decomposition method is considered for the parallel solution of the Helmholtz equation. In particular, we discuss the preconditioned iterative solution of the resulting linear system and present some numerical results.

1 Introduction

Tearing and interconnecting domain decomposition methods [2, 3] are well established for an efficient and parallel solution of various elliptic partial differential equations by using finite and boundary element methods. But in the case of the Helmholtz equation, additional difficulties may appear. Although the global boundary value problem admits a unique solution, local subdomain solvers as used in the tearing and interconnecting approach may fail due to spurious modes. In a recent paper [7] we have introduced a boundary element tearing and interconnecting domain decomposition approach which is robust for all local wave numbers. The aim of the present paper is the discussion of some efficient preconditioners which are needed in the iterative solution of the resulting linear system. In particular we will use preconditioners of the opposite order [6] for the solution of the local boundary value problems, while the construction of the global preconditioner is based on the use of planar waves following the FETI–H method as introduced in [1]. Numerical results confirm the efficiency and the robustness of the proposed solution strategies.

2 Formulation of the Domain Decomposition Approach

As a model problem we consider the Neumann boundary value problem of the Helmholtz equation

$$\Delta u(x) + [\kappa(x)]^2 u(x) = 0 \quad \text{for } x \in \Omega, \qquad \frac{\partial}{\partial n_x} u(x) = g(x) \quad \text{for } x \in \Gamma, \tag{1}$$

where $\Omega \subset \mathbb{R}^3$ is a bounded domain with Lipschitz boundary $\Gamma = \partial\Omega$. We assume that the boundary value problem (1) admits a unique solution. Since the wave number $\kappa(x)$ is assumed to be piecewise constant, i.e. $\kappa(x) = \kappa_i$ for $x \in \Omega_i$, $i = 1, \ldots, p$, instead of (1) we consider the local boundary value problems

$$-\Delta u_i(x) + \kappa_i^2 u_i(x) = 0 \quad \text{for } x \in \Omega_i, \qquad \frac{\partial}{\partial n_i} u_i(x) = g(x) \quad \text{for } x \in \Gamma_i \cap \Gamma, \quad (2)$$

together with the transmission or interface boundary conditions, see Fig. 1,

$$u_i(x) = u_j(x) \quad \text{for } x \in \Gamma_{ij}, \quad (3)$$

$$\frac{\partial}{\partial n_i} u_i(x) + \frac{\partial}{\partial n_j} u_j(x) = 0 \quad \text{for } x \in \Gamma_{ij}. \quad (4)$$

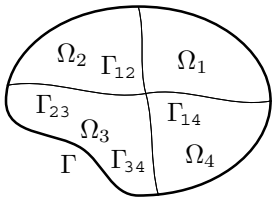


Fig. 1. Decomposition.

To avoid non-unique solutions of either local Dirichlet or Neumann boundary value problems, instead of the Neumann transmission boundary condition in (4) we consider a Robin type interface condition given as

$$\frac{\partial}{\partial n_i} u_i(x) + \frac{\partial}{\partial n_j} u_j(x) + i\eta_{ij} R_{ij}[u_i(x) - u_j(x)] = 0 \quad \text{for } x \in \Gamma_{ij}, i < j, \quad (5)$$

together with the Dirichlet transmission conditions (3). Note that the operators $R_{ij} : H^{1/2}(\Gamma_{ij}) \to \widetilde{H}^{-1/2}(\Gamma_{ij})$ are assumed to be self-adjoint and $H^{1/2}(\Gamma_{ij})$-elliptic, and $\eta_{ij} \in \mathbb{R}\backslash\{0\}$.

The local subdomain boundary $\Gamma_i = \partial\Omega_i$ of a subdomain Ω_i is considered as the union

$$\Gamma_i = (\Gamma_i \cap \Gamma) \cup \bigcup_{\Gamma_{ij}} \Gamma_{ij},$$

where $\Gamma_i \cap \Gamma$ corresponds to the original boundary where Neumann boundary conditions are given, while Γ_{ij} denotes the coupling boundary with an adjacent subdomain. We define

$$(R_i u_{|\Gamma_i})(x) := (R_{ij} u_{|\Gamma_{ij}})(x) \quad \text{for } x \in \Gamma_{ij} \quad (6)$$

and

$$\eta_i(x) := \begin{cases} \eta_{ij} & \text{for } x \in \Gamma_{ij},\ i < j, \\ -\eta_{ij} & \text{for } x \in \Gamma_{ij},\ i > j, \\ 0 & \text{for } x \in \Gamma_i \cap \Gamma\,. \end{cases} \quad (7)$$

We assume, that $\eta_i(x)$ for $x \in \Gamma_i$ does not change its sign. This can be guaranteed either by considering a checker board domain decomposition [1], or by enforcing Robin type boundary conditions only on a part of the local boundary Γ_i, i.e. setting $\eta_{ij} = 0$ on some coupling boundaries Γ_{ij}.

The solutions of the local boundary value problems (2) are given by using the representation formulae [4, 5]

$$u_i(x) = \int_{\Gamma_i} U^*_{\kappa_i}(x,y) t_i(y) ds_y - \int_{\Gamma_i} \frac{\partial}{\partial n_y} U^*_{\kappa_i}(x,y) u_i(y) ds_y \quad \text{for } x \in \Omega_i, \quad (8)$$

where

$$U^*_{\kappa_i}(x,y) = \frac{1}{4\pi} \frac{e^{i\kappa_i |x-y|}}{|x-y|}, \quad t_i(y) := \frac{\partial}{\partial n_y} u_i(y), \; y \in \Gamma$$

are the fundamental solution of the Helmholtz equation and the associated normal derivative of the solution u_i, respectively. By taking the Dirichlet and Neumann traces of the representation formulae (8) we obtain systems of local boundary integral equations which can be written by means of the Calderon projector on Γ_i as

$$\begin{pmatrix} u_i \\ t_i \end{pmatrix} = \begin{pmatrix} \frac{1}{2} I - K_{\kappa_i} & V_{\kappa_i} \\ D_{\kappa_i} & \frac{1}{2} I + K'_{\kappa_i} \end{pmatrix} \begin{pmatrix} u_i \\ t_i \end{pmatrix}. \quad (9)$$

In (9),

$$(V_{\kappa_i} t_i)(x) = \int_{\Gamma_i} U^*_{\kappa_i}(x,y) t_i(y) ds_y, \; (D_{\kappa_i} u)(x) = -\frac{\partial}{\partial n_x} \int_{\Gamma_i} \frac{\partial}{\partial n_y} U^*_{\kappa_i}(x,y) u_i(y) ds_y$$

are the single layer integral operator and the hypersingular integral operator, and

$$(K_{\kappa_i} u_i)(x) = \int_{\Gamma_i} \frac{\partial}{\partial n_y} U^*_{\kappa_i}(x,y) u_i(y) ds_y, \; (K'_{\kappa_i} t_i)(x) = \int_{\Gamma_i} \frac{\partial}{\partial n_x} U^*_{\kappa_i}(x,y) t_i(y) ds_y$$

are the double layer and the adjoint double layer integral operator, respectively. The mapping properties of all boundary integral operators as introduced above are well known [4, 5].

For a Galerkin boundary element discretisation of the boundary integral equations (9) we use piecewise constant basis functions for the local Neumann data t_i and piecewise linear basis functions for the local Dirichlet data u_i. By inserting the transmission boundary conditions (3) and (5), and when applying a tearing and interconnecting approach, we finally obtain the linear system, see [7],

$$\begin{pmatrix} V_{\kappa_1,h} & -\widetilde{K}_{\kappa_1,h} & & & & \\ \widetilde{K}'_{\kappa_1,h} & D_{\kappa_1,h} + i\eta R_{1,h} & & & & -B_1^\top \\ & & \ddots & & & \vdots \\ & & & V_{\kappa_p,h} & -\widetilde{K}_{\kappa_p,h} & \\ & & & \widetilde{K}'_{\kappa_p,h} & D_{\kappa_p,h} + i\eta R_{p,h} & -B_p^\top \\ & B_1 & \dots & & B_p & \end{pmatrix} \begin{pmatrix} \underline{t}_1 \\ \underline{u}_1 \\ \vdots \\ \underline{t}_p \\ \underline{u}_p \\ \underline{\lambda} \end{pmatrix} = \begin{pmatrix} \underline{0} \\ \underline{g}_1 \\ \vdots \\ \underline{0} \\ \underline{g}_p \\ \underline{0} \end{pmatrix}. \quad (10)$$

Note that $\widetilde{K}_{\kappa_i,h} := (\frac{1}{2} M_{i,h} + K_{\kappa_i,h})$, and B_i are boolean matrices which ensure the continuity of the Dirichlet data. Since the related global bilinear form is coercive

satisfying a Gårding inequality, unique solvability of the linear system (10) follows if the mesh size h is sufficiently small [7]. After eliminating the primal degrees of freedom we end up with the Schur complement system

$$F\underline{\lambda} = \sum_{i=1}^{p} \begin{pmatrix} 0 & B_i \end{pmatrix} \begin{pmatrix} V_{\kappa_i,h} & -\widetilde{K}_{\kappa_i,h} \\ \widetilde{K}'_{\kappa_i,h} & D_{\kappa_i,h} + i\eta_i R_{i,h} \end{pmatrix}^{-1} \begin{pmatrix} \underline{0} \\ B_i^\top \underline{\lambda} \end{pmatrix} \tag{11}$$

$$= -\sum_{i=1}^{p} \begin{pmatrix} 0 & B_i \end{pmatrix} \begin{pmatrix} V_{\kappa_i,h} & -\widetilde{K}_{\kappa_i,h} \\ \widetilde{K}'_{\kappa_i,h} & D_{\kappa_i,h} + i\eta_i R_{i,h} \end{pmatrix}^{-1} \begin{pmatrix} \underline{0} \\ \underline{g}_i \end{pmatrix} = \underline{d}\,.$$

3 Construction of Preconditioners

We need to have efficient preconditioners for an iterative solution of the linear system (11) using GMRES in parallel. This involves the construction of a global preconditioner C_F for the assembled stiffness matrix F, and the derivation of local preconditioners C_{A_i} for the local matrices

$$A_i = \begin{pmatrix} -V_{\kappa_i,h} & \widetilde{K}_{\kappa_i,h} \\ \widetilde{K}'_{\kappa_i,h} & D_{\kappa_i,h} + i\eta_i R_{i,h} \end{pmatrix}, \; i = 1, \ldots, p. \tag{12}$$

3.1 Local Preconditioners

We first describe local preconditioners C_{A_i} for the local matrices (12). For this we use a block diagonal preconditioner which is based on the idea of operators of opposite order [6],

$$C_{A_i}^{-1} := \begin{pmatrix} -M_{0,i,h}^{-1} \overline{D}_{i,h} M_{0,i,h}^{-1} & \\ & M_{1,i,h}^{-1} \overline{V}_{i,h} M_{1,i,h}^{-1} \end{pmatrix}, \tag{13}$$

where $M_{0,i,h}$ and $M_{1,i,h}$ are the mass matrices using constant and linear basis functions, respectively. The matrix $\overline{V}_{i,h}$ is the Galerkin discretisation of the single layer integral operator by using piecewise linear and continuous basis functions,

$$\overline{V}_{i,h}[\ell, k] = \frac{1}{4\pi} \int_{\Gamma_i} \phi_{i,\ell}(x) \int_{\Gamma_i} \frac{1}{|x-y|} \phi_{i,k}(y) ds_y ds_s.$$

Accordingly, $\overline{D}_{i,h}$ is the Galerkin discretisation of the stabilised hypersingular boundary integral operator. When using integration by parts, the matrix entries are given as

$$\overline{D}_{i,h}[\ell, k] = \frac{1}{4\pi} \int_{\Gamma_i} \int_{\Gamma_i} \frac{\mathbf{curl}_\Gamma \psi_{i,k}(y) \cdot \mathbf{curl}_\Gamma \psi_{i,\ell}(x)}{|x-y|} ds_y ds_s + \langle 1, \psi_{i,k} \rangle_{\Gamma_i} \langle 1, \psi_{i,\ell} \rangle_{\Gamma_i}.$$

Since the local single layer potential $V_{\kappa_i,h}$ is discretised by using piecewise constant basis functions, also the preconditioning matrix $\overline{D}_{i,h}$ has to be discretised by using

the same piecewise constant basis functions. The application of $\mathbf{curl}_\Gamma$ on a constant function can be interpreted as a distribution on the edges which leads to a formulation based on line integrals

$$\overline{D}_{i,h}[\ell,k] = \frac{1}{4\pi}\int_{\partial\tau_{i,k}}\int_{\partial\tau_{i,\ell}} \frac{r_{i,k}\cdot r_{i,\ell}}{|x-y|} ds_y ds_x + \langle 1,\psi_{i,k}\rangle_{\Gamma_i}\langle 1,\psi_{i,\ell}\rangle_{\Gamma_i},$$

where $r_{i,k}$ and $r_{i,\ell}$ are the direction vectors of the edges of the triangles $\tau_{i,k}$ and $\tau_{i,\ell}$. Note that the described Galerkin discretisation of the hypersingular boundary integral operator by using piecewise constant basis functions is non-conforming, and requires special techniques when evaluating singular line integrals involved.

3.2 Global Preconditioners

For the construction of a global preconditioner we follow an idea of [1]. Let $\underline{r}$ be the residual of the global problem (11), i.e.

$$\underline{r} := \underline{d} - F\underline{\lambda}.$$

The solution algorithm is modified in such a way that the residual $\underline{r}$ is orthogonal to a given m-dimensional subspace which is represented by the columns of an orthogonal matrix Q, i.e.

$$Q^\top \underline{r} = Q^\top(\underline{d} - F\underline{\lambda}) = 0. \tag{14}$$

This restriction implies a solution constraint, since the residual represents the jump of the Dirichlet data on the interface,

$$\underline{r} = \underline{d} - F\underline{\lambda} = \sum_{i=1}^{p} B_i \underline{u}_i.$$

To enforce the orthogonality relation (14), we first introduce a new iterate

$$\widetilde{\underline{\lambda}} := \underline{\lambda} + Q\underline{\gamma} \tag{15}$$

and obtain

$$Q^\top F Q\underline{\gamma} = Q^\top(\underline{d} - F\underline{\lambda}).$$

By solving this equation we get from (15) the alternative representation

$$\widetilde{\underline{\lambda}} = P\underline{\lambda} + \underline{\lambda}^0$$

with the projector

$$P := I - Q(Q^\top F Q)^{-1} Q^\top F$$

and

$$\underline{\lambda}^0 = Q(Q^\top FQ)^{-1}Q^\top \underline{d}\,.$$

From $F\underline{\lambda} = \underline{d}$ we then obtain the linear system

$$FP\underline{\lambda} + F\underline{\lambda}^0 = \underline{d}\,,$$

and after multiplication with the transposed projector $P^\top$ we have to solve the linear system

$$P^\top FP\underline{\lambda} = P^\top \underline{d}\,.$$

Note that $P^\top F\underline{\lambda}^0 = 0$. Moreover, due to

$$P^\top FP = (I - FQ(Q^\top FQ)^{-1}Q^\top)F(I - Q(Q^\top FQ)^{-1}Q^\top F) = FP$$

we can save one application of P, and therefore one application of F in each iteration step.

It remains to discuss the choice of the subspace which is spanned by the orthonormal matrix Q. As in [1] we consider planar waves, which are evaluated locally. In particular, for each subdomain Ω_i we consider a set of m_i directions θ_j, and evaluate the planar wave with the wave number κ_i locally at nodes x_{ℓ_i} to obtain

$$Q^i[\ell_i, j] = e^{i\kappa_i(\theta_j, x_{\ell_i})}\,.$$

The global matrix is finally constructed by

$$Q = [Q^1 \dots Q^i \dots Q^p].$$

By using this local approach one speeds up the construction of $Q^\top FQ$, since only a few local subproblems have to be solved for every direction θ_j. However, we still have to ensure that $Q^\top FQ$ is invertible. For this one has to eliminate certain columns of Q which can be realized when considering a LU factorization of $Q^\top FQ$.

4 Numerical Examples

4.1 Local Preconditioners

We first test the local preconditioner as defined in (13) to solve a linear system with the stiffness matrix (12) where $R_{i,h} = D_{1,h}$ is the Galerkin discretisation of the Yukawa hypersingular integral operator, $\eta_i = 1$, and $\Omega_i = (0,1)^3$ is the unit cube. As solver we use a standard GMRES algorithm with a relative accuracy of $\varepsilon = 10^{-8}$. The right hand side of the linear system to be solved is given by an evaluation of the sinus function, i.e. no geometric information is used. In Table 1, the iteration numbers are given for several wave numbers κ when solving the preconditioned local system (12). Note that N is the number of triangles, and M is the number of nodes.

The number of iterations indicate that the proposed preconditioner is optimal with respect to the boundary element mesh size h, as predicted in theory [6]. However, the spectral condition number of the preconditioned system depends on the properties of the double layer potential, and therefore involves the dependency on the domain Ω, and on the wave number κ. In particular, for an increasing wave number the number of iterations also increases mildly.

Table 1. Number of iterations for preconditioned local system.

N	M	$\kappa = 1.0$	$\kappa = 2.0$	$\kappa = 4.0$	$\kappa = 6.0$	$\kappa = 8.0$
12	8	13	13	15	16	17
48	26	20	22	30	37	46
192	98	24	25	39	52	69
768	386	26	27	42	58	80
3,072	1,538	28	29	43	57	81
12,288	6,146	29	29	42	56	79

4.2 Global Preconditioners

As numerical example we consider the Neumann boundary value problem (1) for the unit cube $\Omega = (0,1)^3$ which is decomposed into $p = n^3$ subdomains Ω_i. The subdomain boundaries $\Gamma_i = \partial\Omega_i$ are discretised uniformly by using 24 plane triangular elements and 14 nodes on the coarsest level ($L = 0$), and are refined uniformly on the next levels. As iterative solver we use a projected GMRES algorithm with a relative accuracy of $\varepsilon = 10^{-8}$. The boundary datum g is chosen such that the exact solution is the fundamental solution $U^*_\kappa(x, \overline{x})$ of the Helmholtz equation with the source $\overline{x} = (-0.2, 0, 0)^\top$.

In Tables 2, 3 and 4 we present the number of global GMRES iterations for different wave numbers κ, and different numbers p of subdomains. By m_i we denote the number of local planar waves as used in the construction of the global preconditioner. We see, that the number of iterations decreases as the number of planar waves increases, while the number of iterations increases with an increasing wave number, as expected.

Acknowledgement. This work was supported by the Austrian Science Fund (FWF) within the project *Data sparse boundary and finite element domain decomposition methods in electromagnetics* under the grant P19255.

Table 2. Number of iterations in the case $\kappa = 2$.

	p=8					p=27					p=64					p=125				
$L \setminus m_i$	0	2	4	6	8	0	2	4	6	8	0	2	4	6	8	0	2	4	6	8
0	23	15	11	7	8	54	25	17	12	13	100	32	19	13	14	165	37	21	14	14
1	29	21	19	17	16	58	32	26	24	22	105	39	28	25	24	156	43	31	26	26
2	31	24	23	21	21	59	34	28	27	25	104	41	32	23	25	137	44	35	30	30
3	35	29	27	26	24	62	39	35	33	30	105	45	37	34	35	137	47	40	31	31

Table 3. Number of iterations in the case $\kappa = 4$.

	p=8					p=27					p=64					p=125				
$L \setminus m_i$	0	2	4	6	8	0	2	4	6	8	0	2	4	6	8	0	2	4	6	8
0	27	19	13	6	12	69	36	23	17	22	130	47	28	18	19	215	59	31	18	17
1	32	23	21	18	16	70	38	33	28	24	128	52	39	33	29	205	60	40	35	33
2	35	27	24	22	21	68	39	33	29	27	121	50	41	36	29	191	56	44	38	34
3	40	31	27	25	24	71	44	39	36	31	119	52	45	40	36	181	57	49	43	38

Table 4. Number of iterations in the case $\kappa = 8$.

	p=8					p=27					p=64					p=125				
$L \setminus m_i$	0	2	4	6	8	0	2	4	6	8	0	2	4	6	8	0	2	4	6	8
0	43	31	16	2	1	88	56	39	25	2	157	93	69	49	24	254	138	88	57	35
1	49	39	28	21	17	80	53	42	36	31	162	89	70	59	50	267	128	95	74	64
2	54	41	31	25	22	81	55	44	37	32	145	77	61	53	47	252	124	81	64	57

References

1. C. Farhat, A. Macedo, and M. Lesoinne. A two-level domain decomposition method for the iterative solution of high frequency exterior Helmholtz problems. *Numer. Math.*, 85(2): 283–308, 2000.
2. C. Farhat and F.-X. Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *Int. J. Numer. Methods Eng.*, 32:1205–1227, 1991.
3. U. Langer and O. Steinbach. Boundary element tearing and interconnecting methods. *Computing*, 71:205–228, 2003.
4. S. Sauter and C. Schwab. *Randelementmethoden. Analyse, Numerik und Implementierung schneller Algorithmen*. B. G. Teubner, Stuttgart, Leipzig, Wiesbaden, 2004.
5. O. Steinbach. *Numerical Approximation Methods for Elliptic Boundary Value Problems. Finite and Boundary Elements*. Springer, New York, NY, 2008.
6. O. Steinbach and W.L. Wendland. The construction of some efficient preconditioners in the boundary element method. *Adv. Comput. Math.*, 9(1–2):191–216, 1998.
7. O. Steinbach and M. Windisch. Stable boundary element domain decomposition methods for the Helmholtz equation. Berichte aus dem Institut für Numerische Mathematik 5/2009, TU Graz, 2009.

A Newton Based Fluid–Structure Interaction Solver with Algebraic Multigrid Methods on Hybrid Meshes

Huidong Yang[1] and Walter Zulehner[2]

[1] Institute of Computational Mathematics, Johannes Kepler University Linz, Altenberger Strasse 69, 4040 Linz, Austria, huidong@numa.uni-linz.ac.at
[2] Institute of Computational Mathematics, Johannes Kepler University Linz, Altenberger Strasse 69, 4040 Linz, Austria, zulehner@numa.uni-linz.ac.at

Summary. Fluid–structure interaction problems arise in many application fields such as flows around elastic structures or blood flow problems in arteries. One method for solving such a problem is based on a reduction to an equation on the interface, involving the so-called Steklov–Poincaré operators. This interface equation is solved by a Newton iteration for which directional derivatives with respect to the interface perturbation have to be evaluated appropriately. One step of the Newton iteration requires the solution of several decoupled linear sub-problems in the structure and the fluid domains. These sub-problems are spatially discretized by a finite element method on hybrid meshes containing different types of elements. For the time discretization implicit first-order methods are used for both sub-problems. The discretized equations are solved by algebraic multigrid methods.

1 Problem Setting of the Fluid–Structure Interaction

1.1 Geometrical Description

Let Ω_0 denote the initial domain at time $t = 0$ consisting of the structure and the fluid domains Ω_0^{s} and Ω_0^{f}, respectively. The domain $\Omega(t)$ at time t is composed of the deformable structure domain $\Omega^{\mathrm{s}}(t)$ and the fluid domain $\Omega^{\mathrm{f}}(t)$. The corresponding interface $\Gamma(t)$ is evolving from the initial interface Γ_0.

The evolution of $\Omega(t)$ is obtained by two families of mappings:

$$\mathcal{L}_t : \Omega_0^{\mathrm{s}} \times R^+ \to \Omega^{\mathrm{s}}(t) \quad \text{and} \quad \mathcal{A}_t : \Omega_0^{\mathrm{f}} \times R^+ \to \Omega^{\mathrm{f}}(t).$$

The maps $\mathcal{L}_t = \mathcal{L}(\cdot, t)$ and $\mathcal{A}_t = \mathcal{A}(\cdot, t)$ track the structure and the fluid domains in time (see Fig. 1 for an illustration). They satisfy the continuity condition of the velocity on the interface $\Gamma(t)$, i.e.

$$\mathcal{L}_t = \mathcal{A}_t \quad \text{on} \quad \Gamma(t). \tag{1}$$

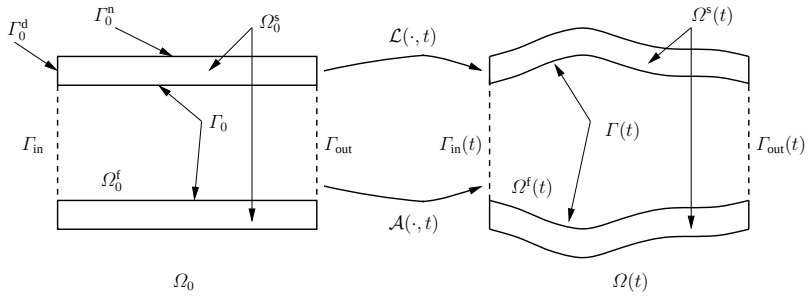


Fig. 1. Two families of mappings.

The structure problem is described in a Lagrangian framework. Therefore, the position of a point $x_0 \in \Omega_0^{\mathrm{s}}$ at time t is given by

$$x(x_0,t) \equiv \mathcal{L}(x_0,t) = x_0 + d^{\mathrm{s}}(x_0,t),$$

where $d^{\mathrm{s}}(x_0,t)$ denotes the displacement $d^{\mathrm{s}}(x_0,t)$ of the structure domain.

Correspondingly, the position of any point $x_0 \in \Omega_0^{\mathrm{f}}$ at time t is given by

$$x(x_0,t) \equiv \mathcal{A}(x_0,t) = x_0 + d^{\mathrm{f}}(x_0,t),$$

where $d^{\mathrm{f}}(x_0,t)$ denotes the displacement of the fluid domain. The fluid problem is stated in an Arbitrary-Lagrangian-Eulerian (ALE) framework. Using the continuity condition (1), $d^{\mathrm{f}}(x_0,t)$ is determined by an arbitrary extension of its value on the interface $d^{\mathrm{f}} = \mathrm{Ext}(d^{\mathrm{s}}|_{\Gamma_0})$, e.g. the harmonic extension:

$$-\Delta d^{\mathrm{f}} = 0 \text{ in } \Omega_0^{\mathrm{f}},\ d^{\mathrm{f}} = d^{\mathrm{s}} \text{ on } \Gamma_0, \text{ and } d^{\mathrm{f}} = 0 \text{ on } \Gamma_{\mathrm{in}}(t) \cup \Gamma_{\mathrm{out}}(t). \tag{2}$$

Furthermore, we introduce the domain velocities by

$$w^{\mathrm{s}}(x_0,t) := \frac{\partial d^{\mathrm{s}}}{\partial t}(x_0,t) \quad \text{and} \quad \hat{w}^{\mathrm{f}}(x_0,t) := \frac{\partial d^{\mathrm{f}}}{\partial t}(x_0,t)$$

for the structure and the fluid domains, respectively.

1.2 The Physical Model

The Lagrange formulation of the pure displacement model of linearized elasticity is defined in the reference material configuration Ω_0^{s}. The state variable d^{s} satisfies the momentum balance law

$$\rho_{\mathrm{s}} \frac{\partial^2 d^{\mathrm{s}}}{\partial t^2} - \mathrm{div}(\sigma_{\mathrm{s}}(d^{\mathrm{s}})) = f_{\mathrm{s}} \qquad \text{in } \Omega_0^{\mathrm{s}}, \tag{3}$$

and the boundary conditions

$$\sigma_{\mathrm{s}}(d^{\mathrm{s}})n_{\mathrm{s}} = 0 \text{ on } \Gamma_0^{\mathrm{n}} \quad \text{and} \quad d^{\mathrm{s}} = 0 \text{ on } \Gamma_0^{\mathrm{d}}, \tag{4}$$

where ρ_s is the density, σ_s the first Piola-Kirchoff stress tensor, f_s is the external force density, and n_s is the outward normal of Ω_0^s. We use the linear Saint-Venant Kirchoff elastic model: $\sigma_s(d^s) = 2\mu^l \varepsilon(d^s) + \lambda^l \mathrm{div}(d^s) I$ with $\varepsilon(d^s) = (\nabla d^s + (\nabla d^s)^T)/2$, and the Lamé constants λ^l and μ^l.

The system of equations for the incompressible fluid problem in the ALE framework is obtained from the balance law of momentum

$$\rho_f \frac{\partial u}{\partial t}\bigg|_{x_0} + \rho_f \left((u - w^f) \cdot \nabla\right) u - 2\mu \mathrm{div}\varepsilon(u) + \nabla p = 0 \quad \text{in } \Omega^f(t), \tag{5}$$

mass conservation

$$\mathrm{div} u = 0 \quad \text{in } \Omega^f(t), \tag{6}$$

and properly chosen boundary conditions

$$\sigma_f(u,p) n_f = g_{\text{in}} \text{ on } \Gamma_{\text{in}}(t) \quad \text{and} \quad \sigma_f(u,p) n_f = 0 \text{ on } \Gamma_{\text{out}}(t), \tag{7}$$

where ρ_f is the fluid density, μ is the dynamic viscosity, $\sigma_f(u,p) = -pI + 2\mu\varepsilon(u)$ and $\varepsilon(u) = (\nabla u + (\nabla u)^T)/2$ are the Cauchy stress tensor σ_f and the strain rate tensor ε, respectively. Here the ALE time derivative of $u(x,t)$ is introduced:

$$\frac{\partial u}{\partial t}\bigg|_{x_0} := \frac{\partial}{\partial t}(u \circ \mathcal{A}_t) \circ (\mathcal{A}_t)^{-1} = \frac{\partial u}{\partial t} + (w^f \cdot \nabla) u$$

for $x \in \Omega^f(t)$, where $w^f(x,t) = \hat{w}^f \circ (\mathcal{A}_t)^{-1}(x)$.

When coupling the two sub-problems together, interface conditions are needed. In particular, no-slip conditions on the interface Γ_0 are explicitly imposed at time t on Γ_0 between the structure and the fluid domains:

$$u \circ \mathcal{A}_t|_{\Gamma_0} = \frac{\partial d^s}{\partial t}\bigg|_{\Gamma_0}. \tag{8}$$

The second interface condition is the equilibrium of normal stresses:

$$(\sigma_f(u,p) n_f) \circ \mathcal{A}_t + \sigma_s(d^s) n_s = 0. \tag{9}$$

To summarize, the complete model consists of problem (2), Eqs. (3), (5), (6), boundary conditions (4), (7), and interface conditions (8), (9) for the state variables d^s, u, p, d^f.

1.3 Reformulation of the Model

As in [1], we express the interface conditions in terms of the so-called Steklov–Poincaré operators for which we introduce the interface variable $\lambda(t)$ by $d^s = d^f = \lambda$ for time t at Γ_0. Then the no-slip interface condition is automatically satisfied.

Let $S_s(\lambda)$ denote the Neumann data $\sigma_s(d^s) n_s$ of the structure problem, where the displacement $d^s := d^s(x_0, t)$ satisfies the Eqs. (3) and (4) with prescribed Dirichlet data $d^s = \lambda$ on the interface Γ_0.

Let $S_{\mathrm{f}}(\lambda)$ denote the Neumann data $\sigma_{\mathrm{f}}(u,p)n_{\mathrm{f}} \circ \mathcal{A}_t$ of the fluid problem, where u and p are determined in the following way: We first compute the harmonic extension $d^{\mathrm{f}} := d^{\mathrm{f}}(x_0,t)$ by solving (2) with Dirichlet condition $d^{\mathrm{f}} = \lambda$ on Γ_0. Then the fluid domain is given by $\Omega^{\mathrm{f}}(t) = d^{\mathrm{f}} + \Omega_0^{\mathrm{f}}$ and we compute u and p by solving (5), (6), (7) with prescribed Dirichlet data $u \circ \mathcal{A}_t = \partial\lambda/\partial t$ on the interface Γ_0.

Then the coupled problem is reduced to the following equation

$$S(\lambda) := S_{\mathrm{s}}(\lambda) + S_{\mathrm{f}}(\lambda) = 0,$$

which is the so-called Steklov–Poincaré equation.

1.4 Time Semi-Discretized Weak Formulations

We need the following function spaces $V^{\mathrm{s}} = [H^1(\Omega_0^{\mathrm{s}})]^3$, $V_0^{\mathrm{s}} = \{v^{\mathrm{s}} \in V^{\mathrm{s}} | v^{\mathrm{s}} = 0 \text{ on } \Gamma_0^{\mathrm{d}} \bigcup \Gamma_0\}$, and $V_g^{\mathrm{s}}(t) = \{v^{\mathrm{s}} \in V^{\mathrm{s}} | v^{\mathrm{s}} = \lambda(t) \text{ on } \Gamma_0\}$ for the structure. For the fluid, we define $D^{\mathrm{f}} = [H^1(\Omega_0^{\mathrm{f}})]^3$, $D_0^{\mathrm{f}} = \{d \in D^{\mathrm{f}} | d = 0 \text{ on } \Gamma_0\}$, $D_g^{\mathrm{f}}(t) = \{d \in D^{\mathrm{f}} | d = \lambda(t) \text{ on } \Gamma_0\}$, $V^{\mathrm{f}}(t) = \{v^{\mathrm{f}} | v^{\mathrm{f}} \circ x_t^{\mathrm{f}} \in [H^1(\Omega_0^{\mathrm{f}})]^3\}$, $V_0^{\mathrm{f}}(t) = \{v^{\mathrm{f}} \in V^{\mathrm{f}}(t) | v^{\mathrm{f}} \circ x_t^{\mathrm{f}} = 0 \text{ on } \Gamma_0\}$, $V_g^{\mathrm{f}}(t) = \{v^{\mathrm{f}} \in V_0^{\mathrm{f}} | v^{\mathrm{f}} \circ x_t^{\mathrm{f}} = w^{\mathrm{f}} \circ x_t^{\mathrm{f}} \text{ on } \Gamma_0\}$, and $Q^{\mathrm{f}}(t) = \{q^{\mathrm{f}} | q^{\mathrm{f}} \circ x_t^{\mathrm{f}} \in L^2(\Omega_0^{\mathrm{f}})\}$, where $H^1(\Omega_0^{\mathrm{s}})$ and $H^1(\Omega_0^{\mathrm{f}})$ denote the standard Sobolev spaces.

Time Semi-discretized Structure Weak Formulation

We denote the time step size by δt and introduce the time level $t^n = n\delta t$.

For the time discretization of the structure problem, we follow the strategy in [1], where the Newmark method with $\gamma = 2\beta = 1$ was proposed:

$$\int_{\Omega_0^{\mathrm{s}}} \rho_{\mathrm{s}} \frac{\partial^2 d^{\mathrm{s}}}{\partial t^2} \cdot v^{\mathrm{s}} dx_0 \approx \frac{2}{\delta t^2} \int_{\Omega_0^{\mathrm{s}}} \rho_{\mathrm{s}} d^{\mathrm{s},n+1} v^{\mathrm{s}} dx_0 - \frac{2}{\delta t^2} \int_{\Omega_0^{\mathrm{s}}} \rho_{\mathrm{s}} (d^{\mathrm{s},n} + \delta t w^{\mathrm{s},n}) v^{\mathrm{s}} dx_0.$$

Here $w^{\mathrm{s},n}$ is the structure domain velocity at time t^n. Using the calculated displacement $d^{\mathrm{s},n+1}$ at time t^{n+1}, we update the structure domain velocity $w^{\mathrm{s},n+1} = 2(d^{\mathrm{s},n+1} - d^{\mathrm{s},n})/\delta t - w^{\mathrm{s},n}$. This leads to the following variational problem, which must be solved in each time step:

Find $d^{\mathrm{s},n+1} = d^{\mathrm{s}}(t^{n+1}) \in V_g^{\mathrm{s}}(t^{n+1})$ such that for all $v^{\mathrm{s}} \in V_0^{\mathrm{s}}$:

$$\begin{aligned} &\frac{2}{\delta t^2} \int_{\Omega_0^{\mathrm{s}}} \rho_{\mathrm{s}} d^{\mathrm{s},n+1} v^{\mathrm{s}} dx_0 + \int_{\Omega_0^{\mathrm{s}}} [\lambda^l \mathrm{div} d^{\mathrm{s},n+1} \mathrm{div} v^{\mathrm{s}} + 2\mu^l \epsilon(d^{\mathrm{s},n+1}) : \epsilon(v^{\mathrm{s}})] dx_0 \\ &= \frac{2}{\delta t^2} \int_{\Omega_0^{\mathrm{s}}} \rho_{\mathrm{s}} (d^{\mathrm{s},n} + \delta t w^{\mathrm{s},n}) v^{\mathrm{s}} dx_0. \end{aligned}$$

Time Semi-discretized Fluid Weak Formulation

Firstly, we compute the harmonic extension of the fluid domain:

Find $d^{\mathrm{f},n+1} \in D_g^{\mathrm{f}}(t^{n+1})$ such that for all $\phi \in D_0^{\mathrm{f}}$:

$$\int_{\Omega_0^{\mathrm{f}}} \nabla d^{\mathrm{f},n+1} : \nabla \phi dx_0 = 0.$$

Then the computational fluid domain is given by $\Omega^{\mathrm{f}}(t^{n+1}) = d^{\mathrm{f},n+1} + \Omega_0^{\mathrm{f}}$, and we set $w^{\mathrm{f},n+1} = ((d^{\mathrm{f},n+1} - d^{\mathrm{f},n})/\delta t) \circ (\mathcal{A}_{t^{n+1}})^{-1}$ for the fluid domain velocity.

For the fluid problem an implicit Euler scheme is used:

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega^{\mathrm{f}}(t)} \rho_{\mathrm{f}} u \cdot v^{\mathrm{f}} dx \bigg|_{t^{n+1}} \approx \frac{\int_{\Omega^{\mathrm{f}}(t^{n+1})} \rho_{\mathrm{f}} u^{n+1} \cdot v^{\mathrm{f},n+1} dx - \int_{\Omega^{\mathrm{f}}(t^n)} \rho_{\mathrm{f}} u^n \cdot v^{\mathrm{f},n} dx}{\delta t},$$

where $v^{\mathrm{f}} = \hat{v}^{\mathrm{f}} \circ (\mathcal{A}_t)^{-1}$, in particular, $v^{\mathrm{f},k} = \hat{v}^{\mathrm{f}} \circ (\mathcal{A}_{t^k})^{-1}$. The non-linear convective term is treated in a semi-implicit way (see [2]). Then we obtain the following time semi-implicit fluid weak formulation:

Find $(u^{n+1}, p^{n+1}) = (u(t^{n+1}), p(t^{n+1})) \in V_g^{\mathrm{f}}(t^{n+1}) \times Q^{\mathrm{f}}(t^{n+1})$ such that for all $(v^{\mathrm{f},n+1}, q^{\mathrm{f},n+1}) \in V_0^{\mathrm{f}}(t^{n+1}) \times Q^{\mathrm{f}}(t^{n+1})$:

$$\begin{aligned}
&\frac{1}{\delta t} \int_{\Omega^{\mathrm{f}}(t^{n+1})} \rho_{\mathrm{f}} u^{n+1} \cdot v^{\mathrm{f},n+1} dx - \int_{\Omega^{\mathrm{f}}(t^{n+1})} \rho_{\mathrm{f}} \left(\mathrm{div} w^{\mathrm{f},n+1}\right) u^{n+1} \cdot v^{\mathrm{f},n+1} dx \\
&\quad + \int_{\Omega^{\mathrm{f}}(t^{n+1})} \rho_{\mathrm{f}} \left(\left(\hat{u}^n - w^{\mathrm{f},n+1}\right) \cdot \nabla\right) u^{n+1} \cdot v^{\mathrm{f},n+1} dx \\
&\quad + 2\mu \int_{\Omega^{\mathrm{f}}(t^{n+1})} \varepsilon(u^{n+1}) : \varepsilon(v^{\mathrm{f},n+1}) dx - \int_{\Omega^{\mathrm{f}}(t^{n+1})} p^{n+1} \, \mathrm{div} v^{\mathrm{f},n+1} dx \\
&\quad = \frac{1}{\delta t} \int_{\Omega^{\mathrm{f}}(t^n)} \rho_{\mathrm{f}} u^n \cdot v^{\mathrm{f},n} dx + \int_{\Gamma_{\mathrm{in}}(t^{n+1})} g_{\mathrm{in}} \cdot v^{\mathrm{f},n+1} ds, \\
&- \int_{\Omega^{\mathrm{f}}(t^{n+1})} q^{n+1} \, \mathrm{div} u^{\mathrm{f},n+1} dx = 0,
\end{aligned}$$

where $\hat{u}^n = u^n \circ \mathcal{A}_{t^n} \circ (\mathcal{A}_{t^{n+1}})^{-1}$.

The Variational Form of the Interface Equation

In the weak form, the previously introduced Steklov–Poincaré operators become operators from the Sobolev space $H^{1/2}(\Gamma_0)$ (which is the space of traces of H^1-functions on Γ_0) to its dual $H^{-1/2}(\Gamma_0)$:

$$S_{\mathrm{s}} : H^{1/2}(\Gamma_0) \to H^{-1/2}(\Gamma_0), \quad S_{\mathrm{f}} : H^{1/2}(\Gamma_0) \to H^{-1/2}(\Gamma_0).$$

Then we end up with the following problem:

Find $\lambda \in H^{1/2}(\Gamma_0)$ such that for all $\mu \in H^{1/2}(\Gamma_0)$:

$$\langle S_{\mathrm{f}}(\lambda), \mu \rangle_{\Gamma_0} + \langle S_{\mathrm{s}}(\lambda), \mu \rangle_{\Gamma_0} = 0. \tag{10}$$

2 Newton's Method for the Interface Equation

The problem (10) has to be solved at each time level $t = t^{n+1} = t^n + \delta t$. For simplicity, we will drop the time variables in the following.

Newton's method applied to the interface equation is given by

$$\lambda^{k+1} = \lambda^k + \delta\lambda^k$$

with

$$\left(S_{\mathrm{s}}^{'}(\lambda^k) + S_{\mathrm{f}}^{'}(\lambda^k)\right)\delta\lambda^k = -\left(S_{\mathrm{s}}(\lambda^k) + S_{\mathrm{f}}(\lambda^k)\right).$$

After spatial discretization this linear problem is solved by the GMRES method.

The method requires the evaluation of $S_{\mathrm{s}}(\lambda)$, $S_{\mathrm{f}}(\lambda)$, $S_{\mathrm{s}}'(\lambda)\delta\lambda$ and $S_{\mathrm{f}}'(\lambda)\delta\lambda$, see [9] for details how to compute these quantities.

3 Finite Element Discretization on Hybrid Meshes

The spatial discretization was done by a finite element method. Let $\mathcal{M}_h$ be the original subdivision of the domain $\Omega \subset \mathbb{R}^3$ into tetrahedra, hexahedra, prisms and pyramids, which is assumed to be admissible, i.e. any two elements from $\mathcal{M}_h$ either have no intersection, or have a common face, or have a common edge, or have a common vertex. Let $\mathcal{T}_h$ be the admissible subdivision into tetrahedra, obtained in the following way: we add points at the centers of quadrilateral faces and subdivide each of them into four triangles, then add a point at the center of the element, and finally connect this center point with all the original vertices and the face center points.

As our finite element space on the hybrid mesh $\mathcal{M}_h$ we first take the standard P_1 finite element space on the underlying tetrahedral mesh $\mathcal{T}_h$ and then replace the degrees of freedom associated to the added points by averaging over neighboring vertices of the original mesh.

This extended P_1 finite element is used for discretizing the structure problem and the interface problem. For the fluid problem we also used the same finite element complemented by a pressure stabilization Petrov-Galerkin (PSPG) and a streamline upwind Petrov-Galerkin (SUPG) technique.

4 AMG for the Structure and the Fluid Sub-problems

After discretization in time and space, linear systems of the form

$$A_{\mathrm{s}h}\underline{d}_{\mathrm{s}h} = \underline{f}_{\mathrm{s}h} \quad \text{and} \quad \begin{pmatrix} A_{\mathrm{f}h} & B_{1,\mathrm{f}h}^T \\ B_{2,\mathrm{f}h} & -C_{\mathrm{f}h} \end{pmatrix} \begin{pmatrix} \underline{u}_h \\ \underline{p}_h \end{pmatrix} = \begin{pmatrix} \underline{f}_{\mathrm{f}h} \\ \underline{g}_{\mathrm{f}h} \end{pmatrix} \tag{11}$$

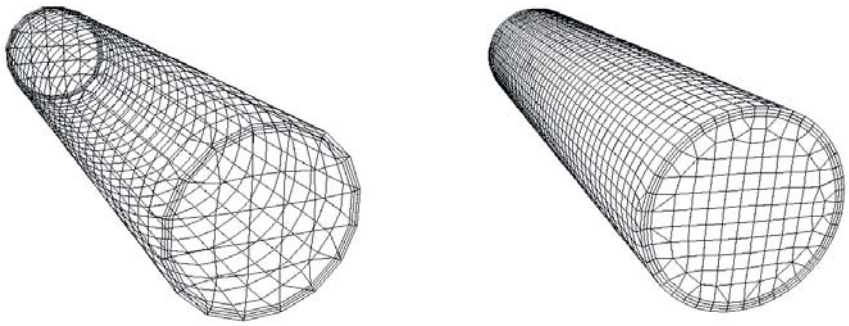
arise at each time step for the structure and the fluid sub-problems, respectively.

The first problem in (11) is symmetric positive and definite, for which the AMG solvers were studied in [3, 4], where a generalization of the classical AMG approach (see [5]) for scalar problems to systems of partial differential equations is discussed.

The system matrix of the second problem in (11) is a saddle point matrix. The AMG approach applied to this type of problem, in particular, to the Stokes or the linearized Navier-Stokes (*Oseen*) problem, stems from previous contributions in [6, 7, 8]. We extended these results to the system arising from the stabilized finite element discretization for the *Oseen* problem on hybrid meshes. In particular, we constructed a stabilized P_1–P_1 hierarchy for the AMG solver on these hybrid meshes, see [9].

5 Numerical Results

We simulate a pressure wave in a cylinder of length 5 cm and radius 5 mm at rest. The thickness of the structure is 0.5 mm. The structure is considered linear and clamped at both the inlet and outlet. The fluid viscosity is set to $\mu = 0.035$, the Lamé constants to $\mu^l = 1.15 \times 10^6$ and $\lambda^l = 1.73 \times 10^6$, the density to $\rho^{\mathrm{f}} = 1.0$ and $\rho^s = 1.2$. The fluid and structure are initially at rest and a pressure of 1.332×10^4 dyn/cm^2 is set on the inlet for a time period of 3 ms. Two meshes[1] (see surface meshes in Fig. 2 as an illustration) are used for simulations:



(a) coarse mesh (b) fine mesh

Fig. 2. Fine and coarse meshes for simulations.

For all simulations, we use the same time step size $\delta t = 1$ ms and run the simulation until the same end time $t = 20$ ms as in [1].

A relative error reduction by a factor of 10^{-5} is achieved in 2–3 outer iterations. Each of these iterations requires 6–8 GMRES iterations for a relative error reduction

[1] All meshes in our test examples were provided by Dipl.- Ing. Ferdinand Kickiger, CAE Software Solutions Wolfkersbühelstr. 23, A-3730 Eggenburg, Austria. See webpage: www.meshing.org.

by a factor of 10^{-5}. For solving the structure problem, about 10 preconditioned conjugate gradient iterations with AMG preconditioning are needed for a relative error reduction by a factor of 10^{-8}, for the fluid problem about 5 AMG iterations for a relative error reduction by a factor of 10^{-8}. Almost the same numbers of iterations were observed for the coarse and the fine mesh.

References

1. S. Deparis, M. Discacciati, G. Fourestey, and A. Quarteroni. Fluid-structure algorithms based on Steklov-poincaré operators. *Comput. Methods Appl. Mech. Eng.*, 195:5797–5812, 2006.
2. M.A. Fernández and M. Moubachir. A Newton method using exact Jacobians for solving fluid-structure coupling. *Comput. Struct.*, 83(2–3):127–142, 2005.
3. M. Griebel, D. Oeltz, and M.A. Schweitzer. An algebraic multigrid method for linear elasticity. *SIAM. J. Sci. Comput.*, 25(2):385–407, 2003.
4. S. Reitzinger. *Algebraic Multigrid Methods for Large Scale Finite Element Equations*. PhD thesis, Johannes Kepler University, Linz, 2001.
5. J.W. Ruge and K. Stüben. Algebraic multigrid (AMG). In *Multigrid Methods*, volume 5 of *Fontiers in Applied Mathematics*, pp. 73–130, SIAM, Philadelphia, PA, 1986.
6. M. Wabro. *Algebraic Multigrid Methods for the Numerical Solution of the Incompressible Navier-Stokes Equations*. PhD thesis, Johannes Kepler University, Linz, 2003.
7. M. Wabro. Coupled algebraic multigrid methods for the oseen problem. *Comput. Vis. Sci.*, 7(3–4):141–151, 2004.
8. M. Wabro. AMGe – Coarsening strategies and application to the Oseen equations. *SIAM J. Sci. Comput.*, 27:2077–2097, 2006.
9. H. Yang. *Numerical Simulations of Fluid-Structure Interaction Problems on Hybrid Meshes with Algebraic Multigrid Methods*. PhD thesis, Johannes Kepler University, Linz, 2010.

Coupled FE/BE Formulations for the Fluid–Structure Interaction

Günther Of and Olaf Steinbach

Institute of Computational Mathematics, TU Graz, A 8010 Graz, Austria, of@tugraz.at, o.steinbach@tugraz.at

Summary. We present several coupled finite and boundary element formulations for the vibro-acoustic simulation of completely immersed bodies such as submarines. All formulations are based on the different use of standard boundary integral equations. In addition to the well known symmetric coupling we discuss two different approaches which are based on the weakly singular boundary integral equation only.

1 Introduction

The simulation of the sound radiation of time-harmonic vibrating elastic structures is of main interest in many applications with the acoustic fluid being air or water. Relevant applications are the sound radiation of passenger car bodies, where the acoustic region is bounded, of partially immersed bodies such as ships, where the acoustic region is a half space, or of completely immersed bodies such as submarines with a full space acoustic region.

In this paper, we consider coupled finite and boundary element formulations for a direct simulation of a three-dimensional time-harmonic vibrating structure in a surrounding fluid [3, 7]. In particular, the time-harmonic vibrating structure in Ω_S is modeled by the Navier equations in the frequency domain,

$$-\varrho_S\omega^2 u(x) - \mu\Delta u(x) - (\lambda+\mu)\text{grad div}\, u(x) = f(x) \quad \text{for } x \in \Omega_S, \qquad (1)$$

where λ and μ are the Lamé parameters, ϱ_S is the density of the structure, ω is the frequency, and u is the unknown displacement field. Note that Ω_S is in general a bounded, multiple connected domain with an interior boundary Γ_N where Neumann boundary conditions

$$t(x) = \lambda\,\text{div}\, u(x)\, n_x + 2\mu\frac{\partial}{\partial n_x}u(x) + \mu\, n_x \times \text{curl}\, u(x) \;=\; g(x) \quad \text{for } x \in \Gamma_N \quad (2)$$

are considered, and with an exterior boundary Γ where transmission conditions are formulated for the coupling with the surrounding fluid. In particular, in the low frequency regime we use the Laplace equation

$$-\Delta p(x) = 0 \quad \text{for } x \in \Omega_F \tag{3}$$

to describe the acoustic pressure p in the unbounded domain Ω_F surrounding the structure in Ω_S. Note that p has to satisfy a radiation condition at infinity,

$$p(x) = \mathcal{O}\Big(\frac{1}{|x|}\Big) \quad \text{as } |x| \to \infty.$$

In addition to the partial differential equations (1) and (3) and the Neumann boundary conditions (2) we consider the transmission conditions on the interface $\Gamma = \overline{\Omega}_F \cap \overline{\Omega}_S$,

$$q(x) = \frac{\partial}{\partial n_x} p(x) = \varrho_F \omega^2 [u(x) \cdot n_x], \quad t(x) = -p(x) n_x \quad \text{for } x \in \Gamma, \tag{4}$$

where ϱ_F is the density of the fluid, and n_x is the exterior normal vector with respect to Ω_S.

The aim of this paper is to derive and to discuss different coupled finite and boundary element formulations for the solution of the transmission boundary value problem (1), (2), (3) and (4). Besides an efficient solution of the direct problem a main interest in applications is the determination of critical frequencies ω which correspond to eigenvalues of the coupled problem with homogeneous data, see, e.g., [1, 2] and the references given therein.

2 Integral Equations and Variational Formulations

The solution of the Laplace equation (3) in the unbounded exterior domain Ω_F is given by the representation formula for $x \in \Omega_F$, see, e.g., [5],

$$p(x) = -\frac{1}{4\pi} \int_\Gamma \frac{1}{|x-y|} q(y) ds_y + \frac{1}{4\pi} \int_\Gamma \frac{(x-y, n_y)}{|x-y|^3} p(y) ds_y. \tag{5}$$

From (5) we obtain a system of boundary integral equations given as

$$\begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} \frac{1}{2} I + K & -V \\ -D & \frac{1}{2} I - K' \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix}. \tag{6}$$

For the structural part we introduce the bilinear forms

$$a_S(u,v) := \int_{\Omega_S} \sum_{i,j=1}^{3} \sigma_{ij}(u(x)) \overline{e_{ij}(v(x))}\, dx, \quad \langle u, v \rangle_{\Omega_S} := \int_{\Omega_S} u(x) \cdot \overline{v(x)}\, dx,$$

for $u, v \in [H^1(\Omega_S)]^3$ as well as the duality pairing, for $t \in [H^{-1/2}(\Gamma)]^3$,

$$\langle t, v \rangle_\Gamma := \int_\Gamma t(x) \cdot \overline{v(x)}_{|\Gamma}\, ds_x.$$

The variational formulation of the structural problem (1) and (2) is to find the displacement field $u \in [H^1(\Omega_S)]^3$ such that

$$a_S(u,v) - \varrho_S \omega^2 \langle u, v\rangle_{\Omega_S} - \langle t, v\rangle_\Gamma = F(v) \tag{7}$$

is satisfied for all $v \in [H^1(\Omega_S)]^3$, where the linear form of the right hand side is given by

$$F(v) := \int_{\Omega_S} f(x) \cdot \overline{v(x)}\, dx + \int_{\Gamma_N} g(x) \cdot \overline{v(x)}_{|\Gamma}\, ds_x\,.$$

By using the second transmission boundary condition in (4), we can rewrite the variational formulation (7) as

$$a_S(u,v) - \varrho_S \omega^2 \langle u, v\rangle_{\Omega_S} + \langle p, v\cdot n\rangle_\Gamma = F(v) \quad \text{for all } v \in [H^1(\Omega_S)]^3, \tag{8}$$

where in addition to $u \in [H^1(\Omega_S)]^3$ also $p \in H^{1/2}(\Gamma)$ is unknown. By using the boundary integral equations as given in (6), and by using the first transmission condition in (4), we will derive a second variational equation to link the two unknowns u and p. Since such an approach is not unique, we will discuss several possible methodologies.

3 Symmetric Coupling of Finite and Boundary Elements

When inserting the first boundary integral equation as given in (6) into the variational problem (8), and by using the first transmission condition in (4), i.e.,

$$p(x) = \frac{1}{2}p(x) + (Kp)(x) - (Vq)(x), \quad q(x) = \varrho_F \omega^2 [u(x)\cdot n_x] \quad \text{for } x \in \Gamma,$$

we have to find $(u,p) \in [H^1(\Omega_S)]^3 \times H^{1/2}(\Gamma)$ satisfying

$$a_S(u,v) - \varrho_S\omega^2\langle u,v\rangle_{\Omega_S} - \varrho_F\omega^2\langle V[u\cdot n], v\cdot n\rangle_\Gamma + \langle (\frac{1}{2}I + K)p, v\cdot n\rangle_\Gamma = F(v) \tag{9}$$

for all $v \in [H^1(\Omega_S)]^3$. In addition we consider the weak formulation of the second, hypersingular, boundary integal equation in (6). Together with the first transmission condition in (4), this gives

$$\langle Dp, \pi\rangle_\Gamma + \varrho_F\omega^2(\frac{1}{2}I + K')[u\cdot n], \pi\rangle_\Gamma = 0 \quad \text{for all } \pi \in H^{1/2}(\Gamma). \tag{10}$$

From the hypersingular boundary integral equation (10) as well as from the coupled variational form (9) we conclude that the acoustic pressure p is only unique up to constants. Hence, to fix the constants we may introduce the modified hypersingular boundary integral operator via the bilinear form

$$\langle \widetilde{D}p, \pi\rangle_\Gamma := \langle Dp, \pi\rangle_\Gamma + \langle p, 1\rangle_\Gamma \langle \pi, 1\rangle_\Gamma \quad \text{for all } p, \pi \in H^{1/2}(\Gamma).$$

Instead of (10) we now consider the modified variational problem

$$\langle \widetilde{D}p, \pi\rangle_\Gamma + \varrho_F\omega^2\langle(\frac{1}{2}I + K')[u\cdot n], \pi\rangle_\Gamma \;=\; 0 \quad \text{for all } \pi \in H^{1/2}(\Gamma), \tag{11}$$

which implies the related scaling of the pressure by $\langle p, 1\rangle_\Gamma = 0$. To summarize, we have to find $(u,p) \in [H^1(\Omega_S)]^3 \times H^{1/2}(\Gamma)$ from the coupled variational problem (9) and (11). Since the modified hypersingular boundary integral operator $\widetilde{D}$ is $H^{1/2}(\Gamma)$–elliptic, we obtain from (11) the representation

$$p \;=\; -\varrho_F\omega^2\widetilde{D}^{-1}(\frac{1}{2}I + K')[u\cdot n],$$

and therefore the continuous Schur complement problem to find $u \in [H^1(\Omega_S)]^3$ such that

$$a_S(u,v) - \omega^2\Big[\varrho_S\langle u, v\rangle_{\Omega_S} + \varrho_F\langle T[u\cdot n], v\cdot n\rangle_\Gamma\Big] = F(v) \tag{12}$$

for all $v \in [H^1(\Omega_S)]^3$. Note that

$$T \;:=\; V + (\frac{1}{2}I + K)\widetilde{D}^{-1}(\frac{1}{2}I + K') \;:\; H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma) \tag{13}$$

is the symmetric and $H^{-1/2}(\Gamma)$-elliptic representation of the Poincaré–Steklov operator realizing the Neumann to Dirichlet map which is related to the Neumann boundary value problem of the Laplace equation in the unbounded exterior domain Ω_F. As a direct consequence of the mapping properties of all involved operators, we can formulate the following result.

Lemma 1. *If ω^2 is not an eigenvalue of the eigenvalue problem*

$$a_S(u,v) \;=\; \lambda\Big[\varrho_S\langle u, v\rangle_{\Omega_S} + \varrho_F\langle T[u\cdot n], v\cdot n\rangle_\Gamma\Big] \quad \textit{for all } v \in [H^1(\Omega_S)]^3,$$

then there exists a unique solution of the variational problem (12), and therefore of the coupled variational problem (9) and (11).

Next we consider a Galerkin discretization of the coupled variational formulation (9) and (11). Let $S_h^1(\Omega_S) \subset H^1(\Omega_S)$ be a conformal finite element space of, e.g., piecewise linear and continuous basis functions with respect to some admissible finite element mesh $\Omega_{S,h}$, and let $S_h^1(\Gamma)$ be some boundary element ansatz space of, e.g., piecewise linear and continuous basis functions which can be defined independently of $S_h^1(\Omega_S)$. The Galerkin discretization of the coupled variational problem (9) and (11) results in the linear system

$$\begin{pmatrix} K_S - \varrho_S\omega^2 M_S - \varrho_F\omega^2 C^\top V_h C & C^\top(\frac{1}{2}M_h + K_h) \\ (\frac{1}{2}M_h^\top + K_h^\top)C & \frac{1}{\varrho_F\omega^2}\widetilde{D}_h \end{pmatrix} \begin{pmatrix} \underline{u} \\ \underline{p} \end{pmatrix} = \begin{pmatrix} \underline{f} \\ \underline{0} \end{pmatrix} \tag{14}$$

where K_S and M_S are the finite element stiffness and mass matrices, respectively. $\widetilde{D}_h$ is the Galerkin matrix of the modified hypersingular boundary integral operator

$\widetilde{D}$. The matrix C describes the basis transformation of a piecewise linear and continuous vector function u_h to a scalar piecewise linear but discontinuous function $u_h \cdot n$ when considering a polygonal boundary mesh Γ_h. Note that V_h is the Galerkin discretization of the single layer potential V when using piecewise linear but discontinuous basis functions, while K_h and M_h are the Galerkin boundary element matrices of the double layer potential K and of the identity.

Since the Galerkin discretization $\widetilde{D}_h$ of the modified hypersingular boundary integral operator $\widetilde{D}$ is invertible, the Schur complement system of (14) is given by

$$\Big(K_S - \omega^2\Big[\varrho_S M_S + \varrho_F C^\top [V_h + (\frac{1}{2}M_h + K_h)\widetilde{D}_h^{-1}(\frac{1}{2}M_h^\top + K_h^\top)]C\Big]\Big)\underline{u} = \underline{f}. \quad (15)$$

As in the continuous case, see (12), we conclude unique solvability of the Schur complement system (14), if ω^2 is not an eigenvalue of the algebraic eigenvalue problem

$$K_S\underline{u} = \lambda\Big(\varrho_S M_S + \varrho_F C^\top [V_h + (\frac{1}{2}M_h + K_h)\widetilde{D}_h^{-1}(\frac{1}{2}M_h^\top + K_h^\top)]C\Big)\underline{u} \quad (16)$$

which is the discrete counterpart of the eigenvalue problem as considered in Lemma 1. Note that

$$T_h \;=\; V_h + (\frac{1}{2}M_h + K_h)\widetilde{D}_h^{-1}(\frac{1}{2}M_h^\top + K_h^\top)$$

is a symmetric boundary element approximation of the Poincaré–Steklov operator as defined in (13).

4 Nonsymmetric Finite and Boundary Element Coupling

Instead of the symmetric coupling of finite and boundary elements, the use of the weakly singular boundary integral equation is very popular in applications in engineering and in industry. This is due to the use of the single layer potential V and the double layer potential K only. Hence we will discuss related formulations which also allow the use of simpler collocation methods for the boundary element discretization.

For the non-sysmmetric coupling we consider two different combinations of the first boundary integral equation as given in (6), of the first transmission condition as given in (4), and of the variational formulation (8).

4.1 A Second Kind Boundary Integral Equation Approach

Inserting the first transmission condition of (4) into the first boundary integral equation in (6) gives the second kind boundary integral equation

$$(\frac{1}{2}I - K)p = -Vq = -\varrho_F\omega^2 V[u \cdot n] \quad \text{on } \Gamma. \quad (17)$$

Since $\frac{1}{2}I - K : H^{1/2}(\Gamma) \to H^{1/2}(\Gamma)$ is invertible, see, e.g. [6], we obtain

$$p = -\varrho_F \omega^2 (\frac{1}{2} I - K)^{-1} V [u \cdot n] = -\varrho_F \omega^2 T [u \cdot n],$$

where

$$T := (\frac{1}{2} I - K)^{-1} V : H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma)$$

is a second representation of the Poincaré–Steklov operator as introduced in (13). From (8) we obtain the variational formulation: find $u \in [H^1(\Omega_S)]^3$ such that

$$a_S(u, v) - \omega^2 \Big[\varrho_S \langle u, v \rangle_{\Omega_S} + \varrho_F \langle T[u \cdot n], v \cdot n \rangle_\Gamma \Big] = F(v)$$

for all $v \in [H^1(\Omega_S)]^3$, which corresponds to the variational problem (12). However, the Galerkin discretization of the variational formulation (8) and of the boundary integral equation (17) now results in the different linear system

$$\begin{pmatrix} K_S - \varrho_S \omega^2 & C^\top \\ -\overline{V} C & \frac{1}{\varrho_F \omega^2} [\frac{1}{2} \overline{M}_h - \overline{K}_h] \end{pmatrix} \begin{pmatrix} \underline{u} \\ \underline{p} \end{pmatrix} = \begin{pmatrix} \underline{f} \\ \underline{0} \end{pmatrix}. \tag{18}$$

Note that the test functions to be used in the Galerkin discretization of the second kind boundary integral equation (17) are the piecewise linear and continous basis functions of $S_h^1(\Gamma)$ as used for the approximation of the pressure p. Although, to the best of our knowledge, there is still no rigorous stability analysis available for general Lipschitz boundaries Γ, the elimination of $\underline{p}$ results in the Schur complement system

$$\Big(K_S - \omega^2 \Big[\varrho_S M_S + \varrho_F C^\top (\frac{1}{2} \overline{M}_h - \overline{K}_h)^{-1} \overline{V}_h C \Big] \Big) \underline{u} = \underline{f}, \tag{19}$$

which is uniquely solvable if ω^2 is not an eigenvalue of the related discrete eigenvalue problem

$$K_S \underline{u} = \lambda \Big(\varrho_S M_S + \varrho_F C^\top (\frac{1}{2} \overline{M}_h - \overline{K}_h)^{-1} \overline{V}_h C \Big) \underline{u}.$$

Note that

$$T_h = (\frac{1}{2} \overline{M}_h - \overline{K}_h)^{-1} \overline{V}_h$$

is a non-symmetric boundary element approximation of the Poincaré–Steklov operator T which is based on an approximate solution of the second kind boundary integral equation (17).

4.2 A First Kind Boundary Integral Equation Approach

Since the single layer potential $V : H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma)$ is invertible, we obtain from the first boundary integral equation of (6), and by using the first transmission boundary condition of (4), the relation

$$q = V^{-1} (-\frac{1}{2} I + K) p = -Sp = \varrho_F \omega^2 [u \cdot n] \quad \text{on } \Gamma,$$

where

$$S = V^{-1}(\frac{1}{2}I - K) : H^{1/2}(\Gamma) \to H^{-1/2}(\Gamma)$$

is the Steklov–Poincaré operator describing the Dirichlet to Neumann map which is related to the Laplace equation in the exterior domain. We therefore obtain

$$p = -\varrho_F \omega^2 S^{-1}[u \cdot n] = -\varrho_F \omega^2 T[u \cdot n], \quad T = S^{-1} = (\frac{1}{2}I - K)^{-1}V,$$

which obviously corresponds to the nonsymmetric approach which is based on the solution of the second kind boundary integral equation (17). Hence, unique solvability of the continuous problem follows as above. However, for a finite and boundary element discretization we consider the coupled system based on the variational formulation (8), the first boundary integral equation in (6), and the first transmission condition in (4). The Galerkin discretization of the coupled system then results in the linear system

$$\begin{pmatrix} K_S - \varrho_S \omega^2 M_S & C^\top & \\ -\varrho_F \omega^2 C & & M_h^\top \\ & \frac{1}{2}M_h - K_h & V_h \end{pmatrix} \begin{pmatrix} \underline{u} \\ \underline{p} \\ \underline{q} \end{pmatrix} = \begin{pmatrix} \underline{f} \\ \underline{0} \\ \underline{0} \end{pmatrix}. \tag{20}$$

Since the discrete single layer potential V_h is invertible, after elimination of $\underline{q}$ we obtain the reduced system

$$\begin{pmatrix} K_S - \varrho_S \omega^2 M_S & C^\top \\ C & \frac{1}{\varrho_F \omega^2} M_h^\top V_h^{-1}(\frac{1}{2}M_h - K_h) \end{pmatrix} \begin{pmatrix} \underline{u} \\ \underline{p} \end{pmatrix} = \begin{pmatrix} \underline{f} \\ \underline{0} \end{pmatrix}. \tag{21}$$

Note that

$$S_h := M_h^\top V_h^{-1}(\frac{1}{2}M_h - K_h)$$

is a non-symmetric representation of the Steklov–Poincaré operator. For stability we need to assume an appropriate choice of the boundary element spaces for an approximation of p and q, respectively, see, e.g. [4]. If S_h is invertible, the Schur complement system of (21),

$$\Big(K_S - \omega^2 \Big[\varrho_S M_S + \varrho_F C^\top S_h^{-1} C\Big]\Big)\underline{u} = \underline{f},$$

is uniquely solvable, if ω^2 is not an eigenvalue of the related eigenvalue problem

$$K_S \underline{u} = \lambda \Big(\varrho_S M_S + \varrho_F C^\top S_h^{-1} C\Big)\underline{u}.$$

5 Conclusions

The symmetric coupling of finite and boundary element methods as described in Sect. 3 admits a complete error and stability analysis, but requires the use of the hypersingular boundary integral operator D, and a Galerkin approach for the discretization of the boundary integral equations. In contrast, both nonsymmetric formulations

as given in Sect. 4 are based on the single and double layer potential operators V and K only, and allow the use of a collocation scheme for a boundary element discretization.

Challenging problems appear in the construction of efficient and robust preconditioning strategies for the solution of the resulting linear systems, in particular when considering the Helmholtz equation instead of the Laplace equation when simulating the sound radiation in the mid frequency regime. The issue of appropriate eigensolvers for the determination of critical frequencies is also of interest. For preliminary and promising results, see [1, 2, 7].

References

1. D. Brunner. *Fast Boundary Element Methods for Large–Scale Simulations of the Vibro–Acoustic Behavior of Ship–Like Structures*. Doctoral Thesis, Universität Stuttgart, 2009.
2. D. Brunner, G. Of, M. Junge, O. Steinbach, and L. Gaul. A fast BE/FE coupling scheme for partly immersed bodies. *Int. J. Numer. Methods Eng.*, 81:28–47, 2010.
3. F. Ihlenburg. Computational experience from the solution of coupled problems in ship design. In A.-M. Sändig, W. Schiehlen, and W.L. Wendland, editors, *Multifield Problems. State of the Art*, pp. 125–134, Springer Berlin, Heidelberg, New York, 2000.
4. O. Steinbach. *Stability Estimates for Hybrid Coupled Domain Decomposition Methods*, volume 1809. Springer Berlin, Heidelberg, New York, 2003.
5. O. Steinbach. *Numerical Approximation Methods for Elliptic Boundary Value Problems. Finite and Boundary Elements*. Springer, New York, NY, 2009.
6. O. Steinbach and W.L. Wendland. On C. Neumann's method for second order elliptic systems in domains with non–smooth boundaries. *J. Math. Anal. Appl.*, 262:733–748, 2001.
7. M. Wilken, G. Of, C. Cabos, and O. Steinbach. Efficient calculation of the effect of water on ship vibration. In G. Guedes Soares and P.K. Das, editors, *Analysis and Design of Marine Structures. Proceedings of MARSTRUCT 2009*, pp. 93–101, CRC Press, 2009.

Domain Decomposition Solvers for Frequency-Domain Finite Element Equations

Dylan Copeland[1], Michael Kolmbauer[2], and Ulrich Langer[2,3]

[1] Institute for Applied Mathematics and Computational Science, Texas A&M University, College Station, USA, copeland@math.tamu.edu
[2] Institute of Computational Mathematics, Johannes Kepler University, Linz, Austria, kolmbauer@numa.uni-linz.ac.at; ulanger@numa.uni-linz.ac.at
[3] Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Linz, Austria, ulrich.langer@assoc.oeaw.ac.at

Summary. The paper is devoted to fast iterative solvers for frequency-domain finite element equations approximating linear and nonlinear parabolic initial boundary value problems with time-harmonic excitations. Switching from the time domain to the frequency domain allows us to replace the expensive time-integration procedure by the solution of a simple linear elliptic system for the amplitudes belonging to the sine- and to the cosine-excitation or a large nonlinear elliptic system for the Fourier coefficients in the linear and nonlinear case, respectively. The fast solution of the corresponding linear and nonlinear system of finite element equations is crucial for the competitiveness of this method.

1 Introduction

In many practical applications, for instance, in electromagnetics and mechanics, the excitation is time-harmonic. Switching from the time domain to the frequency domain allows us to replace the expensive time-integration procedure by the solution of a simple elliptic system for the amplitudes. This is true for linear problems, but not for nonlinear problems. However, due to the periodicity of the solution, we can expand the solution in a Fourier series. Truncating this Fourier series and approximating the Fourier coefficients by finite elements, we arrive at a large-scale coupled nonlinear system for determining the finite element approximation to the Fourier coefficients. In the literature, this approach is called multiharmonic FEM or harmonic-balanced FEM, and has been used by many engineers in different applications. see, e.g. [1] and the references therein.

Reference [2] provided the first rigorous numerical analysis for the eddy current problem. The practical aspects of the multiharmonic approach, including the construction of a fast multigrid preconditioned QMR solver for the Jacobi system arising in every Newton step and the implementation in an adaptive multilevel setting, are discussed in [3] by the same authors. There was no rigorous analysis of the

multigrid preconditioned QMR solver, but the numerical results presented in this paper for academic and more practical problems indicated the efficiency of this solver.

The construction of fast solvers for such systems is very crucial for the overall efficiency of this multiharmonic approach. In this paper, we look at linear and nonlinear, time-harmonic potential problems. We construct and analyze an almost optimal preconditioned GMRes solver for the Jacobi systems arising from the Newton linearization of the large-scale coupled nonlinear system. This preconditioner is not robust with respect to the excitation frequency. In the linear case we are able to construct a robust preconditioner used in a MinRes solver. The multiharmonic approach is presented in Sect. 2, whereas the two different preconditioners and solvers are discussed in Sects. 3 and 4.

2 Frequency-Domain Finite Element Equations

Let us consider the following nonlinear, parabolic, scalar potential equation with a homogeneous Dirichlet boundary condition and an inhomogeneous initial condition as our model problem:

$$\begin{cases} \alpha \frac{\partial u}{\partial t} - \nabla \cdot (\nu(|\nabla u|)\nabla u) = f & \text{in } \Omega \times (0,T], \\ u(\mathbf{x},0) = u_0(\mathbf{x}) & \text{for } \mathbf{x} \in \overline{\Omega}, \\ u(\mathbf{x},t) = 0 & \text{for } (\mathbf{x},t) \in \partial\Omega \times [0,T], \end{cases} \tag{1}$$

where the right-hand side $f(\cdot,\cdot)$ is given by a time-harmonic excitation with the frequency ω, i.e.

$$f(\mathbf{x},t) = f^c(\mathbf{x})\cos(\omega t) + f^s(\mathbf{x})\sin(\omega t). \tag{2}$$

We assume that $\Omega \subset \mathbb{R}^3$ is a bounded Lipschitz domain, α is a given uniformly positive function in $L_\infty(\Omega)$, and $\nu : \mathbb{R}_0^+ \to \mathbb{R}^+$ is a continuously differentiable function satisfying the properties

$$0 < \nu_{\min} \le \nu(s) \le \nu_{\max} \qquad \text{for } s \ge 0, \tag{3}$$

$$\text{and} \quad s \mapsto s\nu(s) \quad \text{is Lipschitz and strongly monotone for } s \ge 0. \tag{4}$$

These conditions ensure that there exists at least a unique weak solution to the initial boundary value problem (1), see [14]. In the linear case where the coefficient ν is independent of $|\nabla u|$, the solution $u(\mathbf{x},t) = u^c(\mathbf{x})\cos(\omega t) + u^s(\mathbf{x})\sin(\omega t)$ is time-harmonic as well, and we get an elliptic boundary value problem for defining the unknown amplitudes u^c and u^s which only depend on the spatial variable $\mathbf{x}$. This is not true in the nonlinear case. However, the solution u to (1) is still periodic in time, with frequency ω. Thus, we have the Fourier series representation

$$u(\mathbf{x},t) = \sum_{k=0}^{\infty} u_k^c(\mathbf{x})\cos(k\omega t) + u_k^s(\mathbf{x})\sin(k\omega t),$$

where the Fourier coefficients are given by

$$u_k^c(\mathbf{x}) = \frac{2}{T}\int_0^T u(\mathbf{x},t)\cos(k\omega t)\,dt \quad \text{and} \quad u_k^s(\mathbf{x}) = \frac{2}{T}\int_0^T u(\mathbf{x},t)\sin(k\omega t)\,dt.$$

Here, the period is $T = 2\pi/\omega$. Similarly, the potential

$$\Psi[u](\mathbf{x},t) := \nu(|\nabla u|)\nabla u(\mathbf{x},t)$$

can be expressed as a Fourier series

$$\Psi[u](\mathbf{x},t) = \sum_{k=0}^{\infty} \Psi_k^c[u](\mathbf{x})\cos(k\omega t) + \Psi_k^s[u](\mathbf{x})\sin(k\omega t)$$

with vector-valued Fourier coefficients Ψ_k^c and Ψ_k^s. Approximating u and Ψ by the truncated series

$$u(\mathbf{x},t) \approx \tilde{u}(\mathbf{x},t) := \sum_{k=0}^{N} u_k^c(\mathbf{x})\cos(k\omega t) + u_k^s(\mathbf{x})\sin(k\omega t) \tag{5}$$

and

$$\Psi[u](\mathbf{x},t) \approx \tilde{\Psi}[\tilde{u}](\mathbf{x},t) := \sum_{k=0}^{N} \Psi_k^c[\tilde{u}](\mathbf{x})\cos(k\omega t) + \Psi_k^s[\tilde{u}](\mathbf{x})\sin(k\omega t)$$

yields the following system of nonlinear equations for the Fourier coefficients:

$$\alpha\omega \begin{pmatrix} 0 & 1 & & \\ -1 & 0 & & \\ & & \ddots & \\ & & 0 & N \\ & & -N & 0 \end{pmatrix} \begin{pmatrix} u_1^c \\ u_1^s \\ \vdots \\ u_N^c \\ u_N^s \end{pmatrix} - \nabla\cdot \begin{pmatrix} \Psi_1^c[\tilde{u}] \\ \Psi_1^s[\tilde{u}] \\ \vdots \\ \Psi_N^c[\tilde{u}] \\ \Psi_N^s[\tilde{u}] \end{pmatrix} = \begin{pmatrix} f^c \\ f^s \\ \vdots \\ 0 \\ 0 \end{pmatrix}. \tag{6}$$

Throughout this paper, we denote by $\mathbf{u} := (u_1^c, u_1^s, \ldots, u_N^c, u_N^s)^T$ the vector of $2N$ Fourier coefficients and by $\tilde{u}$ the approximation to u given by the finite series (5). We shall solve a variational problem for $\mathbf{u}$ in $H_0^1(\Omega)^{2N} := (H_0^1(\Omega))^{2N}$, where $H_0^1(\Omega)$ is the Sobolev space of order 1 on Ω, with vanishing trace on the boundary of Ω. Note that the Fourier coefficients corresponding to $k = 0$ need not be solved for due to the initial condition, cf. [4].

The finite element approximation to (6) leads to a large nonlinear system of finite element equations of the form

$$\mathbf{F_h}(\underline{\mathbf{u}}_\mathbf{h}) = \underline{\mathbf{f}}_\mathbf{h} \tag{7}$$

for determining the finite element solution

$$\mathbf{S}_h^1 := \left(\mathsf{span}\{\phi_j\}_{j=1}^{N_h}\right)^{2N} \ni \tilde{\mathbf{u}}_h \leftrightarrow \underline{\mathbf{u}}_\mathbf{h} = (\underline{u}_{1,h}^c, \underline{u}_{1,h}^s, \ldots, \underline{u}_{N,h}^c, \underline{u}_{N,h}^s)^T \in \mathbb{R}^{2N\cdot N_h}$$

to the Fourier coefficients $\mathbf{u} \in H_0^1(\Omega)^{2N}$. Here, ϕ_j are piecewise linear basis functions in $H_0^1(\Omega)$. Thus the multiharmonic approach yields a time-independent nonlinear system for the solution of which highly parallel solvers can be constructed.

Following [2] we can show that under standard regularity assumptions, the discretization error behaves like $O(h + N^{-1})$ with respect to the $L_2((0,T), H^1(\Omega))$ norm.

Solving (7) by Newton's method ($\tau_n = 1$)

$$\underline{\mathbf{u}}_{\mathbf{h}}^{n+1} = \underline{\mathbf{u}}_{\mathbf{h}}^{n} + \tau_n \underline{\mathbf{w}}_{\mathbf{h}}^{n} = \underline{\mathbf{u}}_{\mathbf{h}}^{n} + \tau_n \mathbf{F}_{\mathbf{h}}'(\underline{\mathbf{u}}_{\mathbf{h}}^{n})^{-1}(\underline{\mathbf{f}}_{\mathbf{h}} - \mathbf{F}_{\mathbf{h}}(\underline{\mathbf{u}}_{\mathbf{h}}^{n})), \tag{8}$$

we have to solve the large-scale linear system

$$\mathbf{F}_{\mathbf{h}}'(\underline{\mathbf{u}}_{\mathbf{h}}^{n})\underline{\mathbf{w}}_{\mathbf{h}}^{n} = \underline{\mathbf{r}}_{\mathbf{h}}^{n} := \underline{\mathbf{f}}_{\mathbf{h}} - \mathbf{F}_{\mathbf{h}}(\underline{\mathbf{u}}_{\mathbf{h}}^{n}), \tag{9}$$

with the Jacobi matrix $\mathbf{F}_{\mathbf{h}}{}'(\underline{\mathbf{u}}_{\mathbf{h}}^{n})$ as system matrix and the residual $\underline{\mathbf{r}}_{\mathbf{h}}^{n}$ as right-hand side.

Reference [4] show that the Jacobi-systems (9) can successfully be solved by the preconditioned GMRes method using a special domain decomposition preconditioner. We will explain the construction of this preconditioner for the corresponding linear problem in the next section, but the results remain valid for the Jacobi-systems (9) too.

In the remainder of this paper, we discuss preconditioned iterative methods for solving linear systems of the form

$$\begin{pmatrix} K_h & \sigma M_h \\ -\sigma M_h & K_h \end{pmatrix} \begin{pmatrix} \underline{u}_h^c \\ \underline{u}_h^s \end{pmatrix} = \begin{pmatrix} \underline{f}_h^c \\ \underline{f}_h^s \end{pmatrix}, \tag{10}$$

arizing from the time-domain finite element discretization of the initial-boundary value problem (1) with the time-harmonic excitation (2) in the linear case where the coefficient ν is independent of $|\nabla u|$. The coefficient σ is equal to $\alpha\omega$. Here and in the following, we assume that α is a positive constant. The stiffness matrix K_h and the mass matrix M_h are computed from the bilinear forms

$$\int_\Omega \nu(\mathbf{x}) \nabla\phi(\mathbf{x}) \cdot \nabla\psi(\mathbf{x})\, d\mathbf{x} \quad \text{and} \quad \int_\Omega \phi(\mathbf{x})\psi(\mathbf{x})\, d\mathbf{x},$$

respectively. The system matrix $\mathbf{D}_{\mathbf{h}}$ in (10) is obviously positive definite and non-symmetric (block skew-symmetric).

3 Domain Decomposition Solver

Following [13], we propose a non-symmetric two-level Schwarz preconditioner for (10) of the form

$$\mathbf{C}_{\mathbf{h}}^{-\mathbf{1}} = \mathbf{I}_{\mathbf{H}}^{\mathbf{h}} \mathbf{D}_{\mathbf{H}}^{-\mathbf{1}} \mathbf{I}_{\mathbf{h}}^{\mathbf{H}} + \beta \mathbf{B}_{\mathbf{h}}^{-\mathbf{1}}, \tag{11}$$

where $\mathbf{D}_{\mathbf{H}}$ is a coarse grid version of $\mathbf{D}_{\mathbf{h}}$, $\mathbf{I}_{\mathbf{h}}^{\mathbf{H}}$ and $\mathbf{I}_{\mathbf{H}}^{\mathbf{h}}$ are appropriate restriction and prolongation operators, $\mathbf{B}_{\mathbf{h}}$ is a symmetric positive definite (SPD) preconditioner

for the SPD part $\mathbf{A_h} = \text{blockdiag}(K_h, K_h)$ of $\mathbf{D_h}$, and β is a positive scaling constant. [5] proposed a wire-basket-based domain decomposition method that gives an effective preconditioner $\mathbf{B_h}$ for the symmetric positive definite matrix $\mathbf{A_h}$, with a condition number estimate which is independent of jumps in the coefficient and depends only polylogarithmically on H/h, see also [11]. Using this wire-basket domain decomposition preconditioner $\mathbf{B_h}$ in (11), we arrive at the following convergence estimate for the GMRes preconditioned by the Xu-Cai preconditioner (11):

Theorem 1 *Assume that the adjoint linear problem is $H^{1+s}(\Omega)^2$-coercive with some $s \in (0,1]$, and H is sufficiently small, specifically $H^s < c(1+\log(H/h))^{-2}$. Then the GMRes method preconditioned by the preconditioner (11) with the wire-basket component $\mathbf{B_h}$ converges and the convergence estimate*

$$\|\underline{\mathbf{r}}_h^m\|_{\mathbf{A_h}} \leq \left(1 - c\, c_{\log}^{-4}\left(1 + c_{\log}^2\right)^{-2}\right)^{m/2} \|\underline{\mathbf{r}}_h^0\|_{\mathbf{A_h}} := \gamma(H/h)^{m/2}\|\underline{\mathbf{r}}_h^0\|_{\mathbf{A_h}}$$

holds for the preconditioned residual $\underline{\mathbf{r}}_h^m = \mathbf{C_h^{-1}}(\underline{\mathbf{f}}_h - \mathbf{D_h}\underline{\mathbf{u}}_h^m)$ at the m-th iteration, where $c_{\log} := 1 + \log(H/h)$, $0 < \gamma(H/h) < 1$, and the constant c depends on ν and σ, but not on H and h.

The proof of this theorem can be found in [4]. In the same paper we present our numerical results which show that our preconditioned GMRes method is a quite efficient solver for the linear system (10) and can efficiently be used for solving the Jacobi-systems (9) as well. The number of iterations depends only polylogarithmically on H/h. In order to clarify the dependence on σ, [7] performed a Fourier analysis of the preconditioned matrix $\mathbf{C_h^{-1}D_h}$ for the corresponding one-dimensional problem with constant ν, where the exact SPD part $\mathbf{A_h}$ was used as $\mathbf{B_h}$, and $H = 2h$. This analysis shows that this preconditioner is not robust with respect to σ, see also the second line of Table 2. In the next section we present a robust preconditioner for the linear system (10) in an equivalent symmetric, but indefinite setting.

4 A Symmetric and Indefinite Reformulation

The non-symmetric and positive definite system (10) can be reformulated in the following equivalent form

$$\begin{pmatrix} M_h & K_h \\ K_h & -\sigma^2 M_h \end{pmatrix} \begin{pmatrix} \underline{u}_h^s \\ \frac{1}{\sigma}\underline{u}_h^c \end{pmatrix} = \begin{pmatrix} \frac{1}{\sigma}\underline{f}_h^c \\ \underline{f}_h^s \end{pmatrix} \tag{12}$$

with a symmetric but indefinite system matrix $\mathbf{D_h}$. For simplicity, we use the same notation $\mathbf{D_h}$ for the system matrix in (10) and (12). It follows from [10] that the block-diagonal preconditioner

$$\mathbf{C_h} = \frac{1}{\sigma}\begin{pmatrix} \sigma M_h + K_h & 0 \\ 0 & \sigma^2(\sigma M_h + K_h) \end{pmatrix} \tag{13}$$

is robust with respect to both the discretization parameter h and the bad parameter σ. More precisely, the condition number

$$\kappa(\mathbf{C_h}^{-1}\mathbf{D_h}) = \|\mathbf{C_h}^{-1}\mathbf{D_h}\|_{\mathbf{C_h}}\|\mathbf{D_h}^{-1}\mathbf{C_h}\|_{\mathbf{C_h}} = |\lambda_{2N_h}| \,/\, |\lambda_1| \leq c = const \quad (14)$$

can be estimated by a positive constant c that is independent of both h and σ, where the eigenvalues of the preconditioned matrix $\mathbf{C_h}^{-1}\mathbf{D_h}$ are ordered in such a way that $|\lambda_{2N_h}| \geq |\lambda_{2N_h-1}| \geq \cdots \geq |\lambda_1| > 0$. Therefore, solving

$$\mathbf{C_h}^{-1}\mathbf{D_h}\underline{\mathbf{u}}_h = \mathbf{C_h}^{-1}\underline{\mathbf{f}}_h$$

by means of the MinRes method proposed by [8], we can ensure that the preconditioned residual $\underline{\mathbf{r}}_h^{2m} = \mathbf{C_h}^{-1}\underline{\mathbf{f}}_h - \mathbf{C_h}^{-1}\mathbf{D_h}\underline{\mathbf{u}}_h^{2m}$ of the $2m$-th MinRes iterate satisfies the iteration error estimate

$$\|\underline{\mathbf{r}}_h^{2m}\|_{\mathbf{C_h}} \leq \frac{2q^m}{1-q^{2m}}\|\underline{\mathbf{r}}_h^0\|_{\mathbf{C_h}} \quad (15)$$

with $q = (\kappa(\mathbf{C_h}^{-1}\mathbf{D_h})-1)/(\kappa(\mathbf{C_h}^{-1}\mathbf{D_h})+1)$, see e.g. [12] or [6]. Thus, the number of MinRes iterations required for reducing the initial error by some fixed factor $\varepsilon \in (0,1)$ is independent of both h and σ. Of course, in practice, the diagonal blocks $\sigma M_h + K_h$ in the preconditioner (13) should be replaced by appropriate preconditioners, e.g. by appropriate domain decompostion or multigrid preconditioners, see e.g. [11].

Applying again the Fourier Analysis (FA) to our one-dimensional problem gives quantitative rates which are displayed in Table 1, for σ ranging from 10^{-10} to 10^{10}.

Table 1. Convergence rate q resulting from the FA ($\varepsilon = 10^{-5}$).

$\log_{10}\sigma$	-10	-8	-6	-4	-2	0	2	4	6	8	10
$h = 1/60$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	0.0005	0.046	0.17	0.17	0.021	0.0002	$< \varepsilon$
$h = 1/120$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	0.0005	0.046	0.17	0.17	0.072	0.0009	$< \varepsilon$
$h = 1/1{,}200$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	0.0005	0.046	0.17	0.17	0.17	0.072	0.0009
$h = 1/12{,}000$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	0.0005	0.046	0.17	0.17	0.17	0.17	0.072
$h = 1/120{,}000$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	$< \varepsilon$	0.0005	0.046	0.17	0.17	0.17	0.17	0.17

Table 2 provides the MinRes iteration numbers which are needed for reducing the initial error by the factor $\varepsilon = 10^{-5}$ for different h and σ. The second line contains the preconditioned GMRes iterations for the constellation $h = 1/60$ and $H = 1/10$, where we use the preconditioner (11) with $\mathbf{B_h} = \mathbf{A_h}$. Both the FA (Table 1) and the numerical experiments (Table 2) were performed for the one-dimensional linear problem resulting in the stiffness matrix $K_h = h^{-1}\text{tridiag}(-1,2,-1)$ and in the mass matrix $M_h = (h/6)\text{tridiag}(1,4,1)$ for the case $\nu = 1$. However, due to the estimates (14) and (15) the numerical behavior observed in our one-dimensional example is characteristic for the three-dimensional linear problem as well.

Table 2. Number of GMRes and MinRes iterations for $\varepsilon = 10^{-5}$.

$\log_{10}\sigma$	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
GMRes	1	1	1	1	1	1	2	2	2	3	4	5	9	18	36	52	52	52	52	52	52
$h = 1/60$	1	1	1	1	1	1	3	3	3	5	7	11	13	13	14	10	6	4	4	2	2
$h = 1/120$	1	1	1	1	1	1	3	3	3	5	7	11	13	13	14	12	8	4	4	2	2
$h = 1/1{,}200$	1	1	1	1	1	1	3	3	3	5	7	11	15	13	14	13	12	10	6	4	4
$h = 1/12{,}000$	1	1	1	1	1	1	3	3	3	5	7	11	15	13	14	13	12	12	11	10	6
$h = 1/120{,}000$	1	1	1	1	1	1	3	3	3	5	7	11	15	13	14	13	12	12	11	10	10

5 Conclusions, Outlook, and Acknowledgments

In this paper we have considered the harmonic and multiharmonic approach to the solution of linear and nonlinear parabolic initial-boundary value problems with harmonic excitation. We have proposed two solution strategies based on a preconditioned GMRes method for the positive definite and non-symmetric problem formulation and a preconditioned MinRes iteration method for the symmetric and indefinite reformulation of the problem. The preconditioner for the GMRes method is a two-level Schwarz preconditioner consisting of a coarse grid solver for the original non-symmetric problem and a wire-basket-based domain decomposition preconditioner for the SPD part. This iterative solver works well for both the linear system (10) arizing from the linear time-harmonic problem and the Jacobi-systems (9) arizing in every step of the Newton iteration (8) for solving the nonlinear equations (7). This preconditioner is highly parallel, but not robust with respect to the bad parameter σ. A robust preconditioner can be constructed for the linear case where ν is independent of $|\nabla u|$. The preconditioner used in the MinRes method has a block-diagonal structure and is robust with respect to both the discretization parameter h and the bad parameter σ. Of course, other iterative methods are possible like the symmetric Uzawa CG method considered in [10] or the QMR method used in [3]. Furthermore, the robust all-at-once multigrid solvers developed by [9] for solving saddle point problems can be an alternative to the preconditioned Krylov-subspace methods considered in this paper. The preconditioned GMRes and MinRes solvers presented in this paper can be generalized to nonlinear eddy current problems studied in [2] and [3].

The authors gratefully acknowledge the financial support by the Austrian Science Fund (FWF) under the grant P19255 and by the Award No. KUS-C1-016-04, made by King Abdullah University of Science and Technology.

References

1. F. Bachinger, M. Kaltenbacher, and S. Reitzinger. An efficient solution strategy for the HBFE method. In *Proceedings of the IGTE '02 Symposium Graz, Austria*, pp. 385–389, 2002.
2. F. Bachinger, U. Langer, and J. Schöberl. Numerical analysis of nonlinear multiharmonic eddy current problems. *Numer. Math.*, 100(4):593–616, 2005.

3. F. Bachinger, U. Langer, and J. Schöberl. Efficient solvers for nonlinear time-periodic eddy current problems. *Comput. Vis. Sci.*, 9(4):197–207, 2006.
4. D.M. Copeland and U. Langer. Domain decomposition solvers for nonlinear multiharmonic finite element equations. *J. Numer. Math.*, 2010. Accepted for publication (see also RICAM-Report 2009-20, RICAM, Austrian Academy of Sciences, Linz, 2009).
5. M. Dryja, B.F. Smith, and O.B. Widlund. Schwarz analysis of iterative substructuring algorithms for elliptic problems in three dimensions. *SIAM J. Numer. Anal.*, 31(6):1662–1694, 1994.
6. A. Greenbaum. *Iterative Methods for Solving Linear Systems*, volume 17 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.
7. M. Kolmbauer. A multiharmonic solver for nonlinear parabolic problems. Master's thesis, Institute for Computational Mathematics, Johannes Kepler University, Linz, 2009.
8. C.C. Paige and M.A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12(4):617–624, 1975.
9. J. Schöberl, R. Simon, and W. Zulehner. A robust multigrid method for an elliptic optimal control problem. Technical Report 2010-01, Institute for Computational Mathematics, Johannes Kepler University, Linz, 2010.
10. J. Schöberl and W. Zulehner. Symmetric indefinite preconditioners for saddle point problems with applications to pde-constrained optimization problems. *SIAM J. Matrix Anal. Appl.*, 29:752–773, 2007.
11. A. Toselli and O.B. Widlund. *Domain Decomposition Methods—Algorithms and Theory*. Springer, Berlin, 2005.
12. H. Voss. *Iterative Methods for Linear Systems of Equations*. Textbook of the 3rd International Summerschool, Jyväskylä, 1993.
13. J. Xu and X.-C. Cai. A preconditioned GMRES method for nonsymmetric or indefinite problems. *Math. Comput.*, 59(200):311–319, 1992.
14. E. Zeidler. *Nonlinear Functional Analysis and Its Applications. II/B*. Springer, New York, NY, 1990.

Deriving the X-Z Identity from Auxiliary Space Method*

Long Chen

Department of Mathematics, University of California at Irvine, Irvine, CA 92697, USA, chenlong@math.uci.edu

1 Iterative Methods

In this paper we discuss iterative methods to solve the linear operator equation

$$Au = f, \tag{1}$$

posed on a finite dimensional Hilbert space $\mathcal{V}$ equipped with an inner product $(\cdot,\cdot)$. Here $A : \mathcal{V} \mapsto \mathcal{V}$ is a symmetric positive definite (SPD) operator, $f \in \mathcal{V}$ is given, and we are looking for $u \in \mathcal{V}$ such that (1) holds.

The X-Z identity for the multiplicative subspace correction method for solving (1) is introduced and proved in [7]. Alternative proofs can be found in [1, 4]. In this paper we derive the X-Z identity from the auxiliary space method [3, 6].

A basic linear iterative method for solving (1) can be written in the following form: starting from an initial guess u^0, for $k = 0, 1, 2, \cdots$

$$u^{k+1} = u^k + B(f - Au^k). \tag{2}$$

Here the non-singular operator $B \approx A^{-1}$ will be called the *iterators*. Let $e^k = u - u^k$. The error equation of the basic iterative method (2) is

$$e^{k+1} = (I - BA)e^k = (I - BA)^k e^0.$$

Thus the iterative method (2) converges if and only if the spectral radius of the error operator $I - BA$ is less than one, i.e., $\rho(I - BA) < 1$.

Given an iterator B, we define the mapping $\Phi_B v = v + B(f - Av)$ and introduce its symmetrization $\Phi_{\overline{B}} = \Phi_{B^t} \circ \Phi_B$. By definition, we have the formula for the error operator $I - \overline{B}A = (I - B^t A)(I - BA)$, and thus

$$\overline{B} = B^t(B^{-t} + B^{-1} - A)B. \tag{3}$$

* The author is supported in part by NSF Grant DMS-0811272, and in part by NIH Grant P50GM76516 and R01GM75309. This work is also partially supported by the Beijing International Center for Mathematical Research.

Since $\overline{B}$ is symmetric, $I - \overline{B}A$ is symmetric with respect to the A-inner product $(u, v)_A := (Au, v)$. Indeed, let $(\cdot)^*$ be the adjoint in the A-inner product $(\cdot, \cdot)_A$. It is easy to show

$$I - \overline{B}A = (I - BA)^*(I - BA). \tag{4}$$

Consequently, $I - \overline{B}A$ is positive semi-definite and thus $\lambda_{\max}(\overline{B}A) \leq 1$. We get

$$\|I - \overline{B}A\|_A = \max\{|1 - \lambda_{\min}(\overline{B}A)|, |1 - \lambda_{\max}(\overline{B}A)|\} = 1 - \lambda_{\min}(\overline{B}A). \tag{5}$$

From (5), we see that $I - \overline{B}A$ is a contraction if and only if $\overline{B}$ is SPD which is also equivalent to $B^{-t} + B^{-1} - A$ being SPD in view of (3).

The convergence of the scheme Φ_B and its symmetrization $\Phi_{\overline{B}}$ is connected by the following inequality:

$$\rho(I - BA)^2 \leq \|I - BA\|_A^2 = \|I - \overline{B}A\|_A = \rho(I - \overline{B}A), \tag{6}$$

and the equality holds if $B = B^t$. Hence we shall focus on the analysis of the symmetric scheme in the rest of this paper.

The iterator B, when it is SPD, can be used as a preconditioner in the Preconditioned Conjugate Gradient (PCG) method, which admits the following estimate:

$$\frac{\|u - u^k\|_A}{\|u - u^0\|_A} \leq 2\left(\frac{\sqrt{\kappa(BA)} - 1}{\sqrt{\kappa(BA)} + 1}\right)^k (k \geq 1), \quad \left(\kappa(BA) = \frac{\lambda_{\max}(BA)}{\lambda_{\min}(BA)}\right).$$

A good preconditioner should have the properties that the action of B is easy to compute and that the condition number $\kappa(BA)$ is significantly smaller than $\kappa(A)$. We shall also discuss construction of multilevel preconditioners in this paper.

2 Auxiliary Space Method

In this section, we present a variation of the fictitious space method [3] and the auxiliary space method [6].

Let $\tilde{\mathcal{V}}$ and $\mathcal{V}$ be two Hilbert spaces and let $\Pi : \tilde{\mathcal{V}} \to \mathcal{V}$ be a surjective map. Denoted by $\Pi^t : \mathcal{V} \to \tilde{\mathcal{V}}$ the adjoint of Π with respect to the default inner products

$$(\Pi^t u, \tilde{v}) := (u, \Pi\tilde{v}) \quad \text{for all } u \in \mathcal{V}, \tilde{v} \in \tilde{\mathcal{V}}.$$

Here, to save notation, we use $(\cdot, \cdot)$ for inner products in both $\mathcal{V}$ and $\tilde{\mathcal{V}}$. Since Π is surjective, its transpose Π^t is injective.

Theorem 1. *Let $\tilde{\mathcal{V}}$ and $\mathcal{V}$ be two Hilbert spaces and let $\Pi : \tilde{\mathcal{V}} \to \mathcal{V}$ be a surjective map. Let $\tilde{B} : \tilde{\mathcal{V}} \to \tilde{\mathcal{V}}$ be a symmetric and positive definite operator. Then $B := \Pi\tilde{B}\Pi^t : \mathcal{V} \to \mathcal{V}$ is also symmetric and positive definite. Furthermore*

$$(B^{-1}v, v) = \inf_{\Pi\tilde{v}=v} (\tilde{B}^{-1}\tilde{v}, \tilde{v}). \tag{7}$$

Proof. We adapt the proof given by [7] (Lemma 2.4).

It is obvious that B is symmetric and positive semi-definite. Since $\tilde{B}$ is SPD and Π^t is injective, $(Bv, v) = (\tilde{B}\Pi^t v, \Pi^t v) = 0$ implies $\Pi^t v = 0$ and consequently $v = 0$. Therefore B is positive definite.

Let $\tilde{v}^* = \tilde{B}\Pi^t B^{-1} v$. Then $\Pi\tilde{v}^* = v$ by the definition of B. For any $\tilde{w} \in \tilde{\mathcal{V}}$

$$(\tilde{B}^{-1}\tilde{v}^*, \tilde{w}) = (\Pi^t B^{-1} v, \tilde{w}) = (B^{-1} v, \Pi\tilde{w}).$$

In particular $(\tilde{B}^{-1}\tilde{v}^*, \tilde{v}^*) = (B^{-1}v, \Pi\tilde{v}^*) = (B^{-1}v, v)$. For any $\tilde{v} \in \tilde{\mathcal{V}}$, denote by $v = \Pi\tilde{v}$. We write $\tilde{v} = \tilde{v}^* + \tilde{w}$ with $\Pi\tilde{w} = 0$. Then

$$\begin{aligned}
\inf_{\Pi\tilde{v}=v} (\tilde{B}^{-1}\tilde{v}, \tilde{v}) &= \inf_{\Pi\tilde{w}=0} (\tilde{B}^{-1}(\tilde{v}^* + \tilde{w}), \tilde{v}^* + \tilde{w}) \\
&= (B^{-1}v, v) + \inf_{\Pi\tilde{w}=0} \Big(2(\tilde{B}^{-1}\tilde{v}^*, \tilde{w}) + (\tilde{B}^{-1}\tilde{w}, \tilde{w}) \Big) \\
&= (B^{-1}v, v) + \inf_{\Pi\tilde{w}=0} (\tilde{B}^{-1}\tilde{w}, \tilde{w}) \\
&= (B^{-1}v, v).
\end{aligned}$$

□

The symmetric positive definite operator B may be used as a preconditioner for solving $Au = f$ using PCG. To estimate the condition number $\kappa(BA)$, we only need to compare B^{-1} and A.

Lemma 1. *For two SPD operators A and B, if $c_0(Av, v) \le (B^{-1}v, v) \le c_1(Av, v)$ for all $v \in \mathcal{V}$, then $\kappa(BA) \le c_1/c_0$.*

Proof. Note that BA is symmetric with respect to A. Therefore

$$\lambda_{\min}^{-1}(BA) = \lambda_{\max}((BA)^{-1}) = \sup_{u \in \mathcal{V}\setminus\{0\}} \frac{((BA)^{-1}u, u)_A}{(u, u)_A} = \sup_{u \in \mathcal{V}\setminus\{0\}} \frac{(B^{-1}u, u)}{(Au, u)}.$$

Therefore $(B^{-1}v, v) \le c_1(Av, v)$ implies $\lambda_{\min}(BA) \ge c_1^{-1}$. Similarly $(B^{-1}v, v) \ge c_0(Av, v)$ implies $\lambda_{\max}(BA) \le c_0^{-1}$. The estimate of $\kappa(BA)$ then follows. □

By Lemma 1 and Theorem 1, we have the following result.

Corollary 1. *Let $\tilde{B} : \tilde{\mathcal{V}} \to \tilde{\mathcal{V}}$ be SPD and $B = \Pi\tilde{B}\Pi^t$. If*

$$c_0(Av, v) \le \inf_{\Pi\tilde{v}=v} (\tilde{B}^{-1}\tilde{v}, \tilde{v}) \le c_1(Av, v) \quad \textit{for all } v \in \mathcal{V}, \tag{8}$$

then $\kappa(BA) \le c_1/c_0$.

Remark 1. In literature, e.g. the fictitious space lemma of [3], the condition (8) is usually decomposed as the following two conditions.

(i) For any $v \in \mathcal{V}$, there exists a $\tilde{v} \in \tilde{V}$, such that $\Pi\tilde{v} = v$ and $\|\tilde{v}\|^2_{\tilde{B}^{-1}} \le c_1\|v\|^2_A$.
(ii) For any $\tilde{v} \in \tilde{\mathcal{V}}$, $\|\Pi\tilde{v}\|^2_A \le c_0^{-1}\|\tilde{v}\|^2_{\tilde{B}^{-1}}$.

3 Auxiliary Spaces of Product Type

Let $\mathcal{V}_i \subseteq \mathcal{V}$, $i = 0, \ldots, J$, be subspaces of $\mathcal{V}$. If $\mathcal{V} = \sum_{i=0}^{J} \mathcal{V}_i$, then $\{\mathcal{V}_i\}_{i=0}^{J}$ is called a *space decomposition* of $\mathcal{V}$. Then for any $u \in \mathcal{V}$, there exists a decomposition $u = \sum_{i=0}^{J} u_i$. Since $\sum_{i=0}^{J} \mathcal{V}_i$ is not necessarily a direct sum, decompositions of u are in general not unique.

We introduce the inclusion operator $I_i : \mathcal{V}_i \to \mathcal{V}$, the projection operator $Q_i : \mathcal{V} \to \mathcal{V}_i$ in the $(\cdot, \cdot)$ inner product, the projection operator $P_i : \mathcal{V} \to \mathcal{V}_i$ in the $(\cdot, \cdot)_A$ inner product, and $A_i = A|_{\mathcal{V}_i}$. It can be easily verified that $Q_i A = A_i P_i$ and $Q_i = I_i^t$.

Given a space decomposition $\mathcal{V} = \sum_{i=0}^{J} \mathcal{V}_i$, we construct an auxiliary space of product type $\tilde{\mathcal{V}} = \mathcal{V}_0 \times \mathcal{V}_1 \times ... \times \mathcal{V}_J$, with the inner product $(\tilde{u}, \tilde{v}) := \sum_{i=0}^{J} (u_i, v_i)$. We define $\Pi : \tilde{\mathcal{V}} \to \mathcal{V}$ as $\Pi \tilde{u} = \sum_{i=0}^{J} u_i$. In operator form $\Pi = (I_0, I_1, \cdots, I_J)$. Since $\mathcal{V} = \sum_{i=0}^{J} \mathcal{V}_i$, the operator Π is surjective.

Let $R_i : \mathcal{V}_i \to \mathcal{V}_i$ be nonsingular operators, often known as smoothers, approximating A_i^{-1}. Define a diagonal matrix of operators $\tilde{R} = \mathrm{diag}(R_0, R_1, \cdots, R_J) : \tilde{\mathcal{V}} \to \tilde{\mathcal{V}}$ which is non-singular. An additive preconditioner is defined as

$$B_a = \Pi \tilde{R} \Pi^t = \sum_{i=0}^{J} I_i R_i I_i^t = \sum_{i=0}^{J} I_i R_i Q_i. \tag{9}$$

Applying Theorem 1, we obtain the following identity for preconditioner B_a.

Theorem 2. *If R_i is SPD on $\mathcal{V}_i$ for $i = 0, \ldots, J$, then B_a defined by (9) is SPD on $\mathcal{V}$. Furthermore*

$$(B_a^{-1} v, v) = \inf_{\sum_{i=0}^{J} v_i = v} \sum_{i=0}^{J} (R_i^{-1} v_i, v_i). \tag{10}$$

To define a multiplicative preconditioner, we introduce the operator $\tilde{A} = \Pi^t A \Pi$. By direct computation, the entry $\tilde{a}_{ij} = Q_i A I_j = A_i P_i I_j$. In particular $\tilde{a}_{ii} = A_i$. The symmetric operator $\tilde{A}$ may be singular with nontrivial kernel $\ker(\Pi)$, but the diagonal of $\tilde{A}$ is always non-singular. Write $\tilde{A} = \tilde{D} + \tilde{L} + \tilde{U}$ where $\tilde{D} = \mathrm{diag}(A_0, A_1, \cdots, A_j)$, $\tilde{L}$ and $\tilde{U}$ are lower and upper triangular matrix of operators, and $\tilde{L}^t = \tilde{U}$. Note that the operator $\tilde{R}^{-1} + \tilde{L}$ is invertible. We define $\tilde{B}_m = (\tilde{R}^{-1} + \tilde{L})^{-1}$ and its symmetrization as

$$\overline{\tilde{B}}_m = \tilde{B}_m^t + \tilde{B}_m - \tilde{B}_m^t \tilde{A} \tilde{B}_m = \tilde{B}_m^t (\tilde{B}_m^{-t} + \tilde{B}_m^{-1} - \tilde{A}) \tilde{B}_m. \tag{11}$$

The symmetrizied multiplicative preconditioner is defined as

$$\overline{B}_m := \Pi \overline{\tilde{B}}_m \Pi^t. \tag{12}$$

We define the diagonal matrix of operators $\overline{\tilde{R}} = \mathrm{diag}(\overline{R}_0, \overline{R}_1, \cdots, \overline{R}_J)$, where, for each $R_i, i = 0, \cdots, J$, its symmetrization is

$$\overline{R}_i = R_i^t(R_i^{-t} + R_i^{-1} - A_i)R_i.$$

Substituting $\tilde{B}_m^{-1} = \tilde{R}^{-1} + \tilde{L}$, and $\tilde{A} = \tilde{D} + \tilde{L} + \tilde{U}$ into (11), we have

$$\begin{aligned}\overline{\tilde{B}}_m &= (\tilde{R}^{-t} + \tilde{L}^t)^{-1}(\tilde{R}^{-t} + \tilde{R}^{-1} - \tilde{D})(\tilde{R}^{-1} + \tilde{L})^{-1} \\ &= (\tilde{R}^{-t} + \tilde{L}^t)^{-1}\tilde{R}^{-t}\overline{\tilde{R}}\tilde{R}^{-1}(\tilde{R}^{-1} + \tilde{L})^{-1}. \end{aligned} \tag{13}$$

It is obvious that $\overline{\tilde{B}}_m$ is symmetric. To be positive definite, from (13), it suffices to assume $\overline{\tilde{R}}$, i.e. each $\overline{R}_i$, is symmetric and positive definite which is equivalent to that operator $I - \overline{R}_i A_i$ is a contraction and so is $I - R_i A_i$.

(A) $\|I - R_i A_i\|_{A_i} < 1$ for each $i = 0, \cdots, J$.

Theorem 3. *Suppose (A) holds. Then $\overline{B}_m$ defined by (12) is SPD, and*

$$(\overline{B}_m^{-1}v, v) = \|v\|_A^2 + \inf_{\sum_{i=0}^J v_i = v} \sum_{i=0}^J \|R_i^t(A_i P_i \sum_{j=i}^J v_j - R_i^{-1}v_i)\|^2_{\overline{R}_i^{-1}}. \tag{14}$$

In particular, for $R_i = A_i^{-1}$, we have

$$(\overline{B}_m^{-1}v, v) = \|v\|_A^2 + \inf_{\sum_{i=0}^J v_i = v} \sum_{i=0}^J \|P_i \sum_{j=i+1}^J v_j\|_A^2. \tag{15}$$

Proof. Let

$$\mathscr{M} = \tilde{R}^{-t} + \tilde{R}^{-1} - \tilde{D} = \tilde{R}^{-t}\overline{\tilde{R}}\tilde{R}^{-1}, \quad \mathscr{U} = \tilde{D} + \tilde{U} - \tilde{R}^{-1}, \quad \mathscr{L} = \mathscr{U}^t.$$

then $\tilde{R}^{-1} + \tilde{L} = \mathscr{M} + \mathscr{L}$ and $\tilde{A} = \mathscr{M} + \mathscr{L} + \mathscr{U}$. We then compute

$$\begin{aligned}\overline{\tilde{B}}_m^{-1} &= (\tilde{R}^{-1} + \tilde{L})(\tilde{R}^{-t} + \tilde{R}^{-1} - \tilde{D})^{-1}(\tilde{R}^{-t} + \tilde{L}^t) \\ &= (\mathscr{M} + \mathscr{L})\mathscr{M}^{-1}(\mathscr{M} + \mathscr{U}), \\ &= \tilde{A} + \mathscr{L}\mathscr{M}^{-1}\mathscr{U} \\ &= \tilde{A} + \left[\tilde{R}^t(\tilde{D} + \tilde{U} - \tilde{R}^{-1})\right]^t \overline{\tilde{R}}^{-1}\left[\tilde{R}^t(\tilde{D} + \tilde{U} - \tilde{R}^{-1})\right]. \end{aligned}$$

For any $\tilde{v} \in \overline{\mathcal{V}}$, denoted by $v = \Pi\tilde{v}$, we have

$$(\tilde{A}\tilde{v}, \tilde{v}) = (\Pi^t A \Pi\tilde{v}, \tilde{v}) = (A\Pi\tilde{v}, \Pi\tilde{v}) = \|v\|_A^2.$$

Using component-wise formula of $\tilde{R}^t(\tilde{D} + \tilde{U} - \tilde{R}^{-1})\tilde{v}$, e.g. $((\tilde{D} + \tilde{U})\tilde{v})_i = \sum_{j=i}^J \tilde{a}_{ij}v_j = \sum_{j=i}^J A_i P_i v_j$, we get

$$(\mathscr{M}^{-1}\mathscr{U}\tilde{v}, \mathscr{U}\tilde{v}) = \sum_{i=0}^J \|R_i^t(A_i P_i \sum_{j=i}^J v_j - R_i^{-1}v_i)\|^2_{\overline{R}_i^{-1}}.$$

The identity (14) then follows. □

If we further introduce the operator $T_i = R_i A_i P_i : \mathcal{V} \to \mathcal{V}_i$, then $T_i^* = R_i^t A_i P_i$, $\overline{T}_i := T_i + T_i^* - T_i^* T_i = \overline{R}_i A_i P_i$, and $(\overline{R}_i^{-1} w_i, w_i) = (A_i \overline{T}_i^{-1} w_i, w_i) = (\overline{T}_i^{-1} w_i, w_i)_A$. Here $\overline{T}_i^{-1} := (\overline{T}_i|_{\mathcal{V}_i})^{-1} : \mathcal{V}_i \to \mathcal{V}_i$ is well defined due to the assumption (A). We then recovery the original formulation in [7]

$$(\overline{B}_m^{-1} v, v) = \|v\|_A^2 + \inf_{\sum_{i=0}^J v_i = v} \sum_{i=0}^J (\overline{T}_i^{-1} T_i^* w_i, T_i^* w_i)_A,$$

with $w_i = \sum_{j=i}^J v_j - T_i^{-1} v_i$. With these notation, we can also use (13) to recovery the formula in [1]

$$(\overline{B}_m^{-1} v, v) = \inf_{\sum_{i=0}^J v_i = v} \sum_{i=0}^J (\overline{T}_i^{-1} (v_i + T_i^* w_i), v_i + T_i^* w_i)_A.$$

4 Method of Subspace Correction

In this section, we view the method of subspace correction [5] as an auxiliary space method and provide identities for the convergence analysis.

Let $\mathcal{V} = \sum_{i=0}^J \mathcal{V}_i$ be a space decomposition of $\mathcal{V}$. For a given residual $r \in \mathcal{V}$, we let $r_i = Q_i r$ be the restriction of the residual to the subspace $\mathcal{V}_i$ and solve the residual equation in the subspaces

$$A_i e_i = r_i \quad \text{approximately by} \quad \hat{e}_i = R_i r_i.$$

Subspace corrections $\hat{e}_i$ are assembled together to give a correction in the space $\mathcal{V}$. There are two basic ways to assemble subspace corrections.

Parallel Subspace Correction (PSC)

This method is to do the correction on each subspace in parallel. In operator form, it reads

$$u^{k+1} = u^k + B_a(f - Au^k), \tag{16}$$

where

$$B_a = \sum_{i=0}^J I_i R_i I_i^t = \sum_{i=0}^J I_i R_i Q_i. \tag{17}$$

Thus the PSC are also called additive methods. Note that the formula (17) and (9) are identical and thus identity (10) is useful to estimate $\kappa(B_a A)$.

Successive Subspace Correction (SSC)

This method involves successive corrections. In operator form, it reads

$$v^0 = u^k, \quad v^{i+1} = v^i + R_i Q_i (f - A v^i), i = 0, \ldots, N, \quad u^{k+1} = v^{J+1}. \tag{18}$$

For each subspace problem, we have the operator form $v^{i+1} = v^i + R_i(f - Av^i)$, but it is not easy to write out the iterator for the space $\mathcal{V}$. We define B_m such that the error operator

$$I - B_m A = (I - R_J Q_J A)(I - R_{J-1} Q_{J-1} A)...(I - R_0 Q_0 A).$$

Therefore the SSC are also called multiplicative methods. We now derive a formulation of B_m from the auxiliary space method.

In the sequel, we consider the SSC applied to the space decomposition $\mathcal{V} = \sum_{i=0}^{J} \mathcal{V}_i$ with smoothers $R_i, i = 0, \cdots, J$. Recall that $\tilde{\mathcal{V}} = \mathcal{V}_0 \times \mathcal{V}_1 \times ... \times \mathcal{V}_J$ and $\tilde{A} = \Pi^t A \Pi$. Let $\tilde{f} = \Pi^t f$. Following [2], we view SSC for solving $Au = f$ as a Gauss-Seidel type method for solving $\tilde{A}\tilde{u} = \tilde{f}$.

Lemma 2. *Let $\tilde{A} = \tilde{D} + \tilde{L} + \tilde{U}$ and $\tilde{B} = (\tilde{R}^{-1} + \tilde{L})^{-1}$. Then the SSC for $Au = f$ with the smoothers R_i is equivalent to the Gauss–Seidel type method for solving $\tilde{A}\tilde{u} = \tilde{f}$:*

$$\tilde{u}^{k+1} = \tilde{u}^k + \tilde{B}(\tilde{f} - \tilde{A}\tilde{u}^k). \tag{19}$$

Proof. By multiplying (19) by $\tilde{R}^{-1} + \tilde{L}$ and rearranging the terms, we have

$$\tilde{R}^{-1}\tilde{u}^{k+1} = \tilde{R}^{-1}\tilde{u}^k + \tilde{f} - \tilde{L}\tilde{u}^{k+1} - (\tilde{D} + \tilde{U})\tilde{u}^k.$$

Multiplying by $\tilde{R}$, we obtain

$$\tilde{u}^{k+1} = \tilde{u}^k + \tilde{R}(\tilde{f} - \tilde{L}\tilde{u}^{k+1} - (\tilde{D} + \tilde{U})\tilde{u}^k),$$

and its component-wise formula, for $i = 0, \cdots, J$

$$\begin{aligned} u_i^{k+1} &= u_i^k + R_i(f_i - \sum_{j=0}^{i-1} \tilde{a}_{ij} u_j^{k+1} - \sum_{j=i}^{J} \tilde{a}_{ij} u_j^k) \\ &= u_i^k + R_i Q_i (f - A\sum_{j=0}^{i-1} u_j^{k+1} - A\sum_{j=i}^{J} u_j^k). \end{aligned}$$

Let

$$v^{i-1} = \sum_{j=0}^{i-1} u_j^{k+1} + \sum_{j=i}^{J} u_j^k.$$

Noting that $v^i - v^{i-1} = u_i^{k+1} - u_i^k$, we then get, for $i = 1, \cdots, J+1$

$$v^i = v^{i-1} + R_i Q_i (f - Av^{i-1}),$$

which is exactly the correction on $\mathcal{V}_i$; see (18). □

Lemma 3. *For SSC, we have*

$$B_m = \Pi \tilde{B}_m \Pi^t \quad \text{and} \quad \overline{B}_m = \Pi \overline{\tilde{B}}_m \Pi^t.$$

Proof. Let $u^k = \Pi \tilde{u}^k$. Applying Π to (19) and noting that

$$\tilde{f} = \Pi^t f, \quad \text{and} \quad \tilde{A}\tilde{u}^k = \Pi^t A u^k,$$

we then get

$$u^{k+1} = u^k + \Pi \tilde{B} \Pi^t (f - A u^k).$$

Therefore $B_m = \Pi \tilde{B}_m \Pi^t$. The formulae for $\overline{B}_m$ is obtained similarly. □

Combining Lemma 3, (5), (6), and Theorem 3, we obtain the X-Z identity.

Theorem 4 (X-Z identity). *Suppose assumption (A) holds. Then*

$$\|(I - R_J Q_J A)(I - R_{J-1} Q_{J-1} A) \ldots (I - R_0 Q_0 A)\|_A^2 = 1 - \frac{1}{1 + c_0}, \tag{20}$$

where

$$c_0 = \sup_{\|v\|_A = 1} \inf_{\sum_{i=0}^J v_i = v} \sum_{i=0}^J \|R_i^t (A_i P_i \sum_{j=i}^J v_j - R_i^{-1} v_i)\|_{R_i^{-1}}^2.$$

In particular, for $R_i = A_i^{-1}$,

$$\|(I - P_J)(I - P_{J-1}) \cdots (I - P_0)\|_A^2 = 1 - \frac{1}{1 + c_0}, \tag{21}$$

where

$$c_0 = \sup_{\|v\|_A = 1} \inf_{\sum_{i=0}^J v_i = v} \sum_{i=0}^J \|P_i \sum_{j=i+1}^J v_j\|_A^2.$$

References

1. D. Cho, J. Xu, and L. Zikatanov. New estimates for the rate of convergence of the method of subspace corrections. *Numer. Math. Theor. Methods Appl.*, 1:44–56, 2008.
2. M. Griebel and P. Oswald. On the abstract theory of additive and multiplicative Schwarz methods. *Numer. Math.*, 70:163–180, 1995.
3. S.V. Nepomnyaschikh. Decomposition and fictitious domains methods for elliptic boundary value problems. In *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations (Norfolk, VA, 1991)*, pp. 62–72, SIAM, Philadelphia, PA, 1992.
4. P.S. Vassilevski. *Multilevel Block Factorization Preconditioners: Matrix-based Analysis and Algorithms for Solving Finite Element Equations*. Springer, New York, NY, 2008.
5. J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.*, 34: 581–613, 1992.
6. J. Xu. The auxiliary space method and optimal multigrid preconditioning techniques for unstructured meshes. *Computing*, 56:215–235, 1996.
7. J. Xu and L. Zikatanov. The method of alternating projections and the method of subspace corrections in Hilbert space. *J. Am. Math. Soc.*, 15:573–597, 2002.

A Near-Optimal Hierarchical Estimate Based Adaptive Finite Element Method for Obstacle Problems

Qingsong Zou

Department of Scientific Computing and Computer Applications, Sun Yat-sen University, Guangzhou 510275, P. R. China, mcszqs@mail.sysu.edu.cn

1 Introduction

In this paper, we will derive a novel adaptive finite element method for the following symmetric, elliptic obstacle problem: Find $u \in K$ such that

$$a(u, v-u) \geq (f, v-u) \qquad \forall v \in K \tag{1}$$

where $\Omega \subset \mathbb{R}^2$ is a bounded polygonal domain with Lipschitz-continuous boundary $\partial\Omega$, $\psi \in C(\overline{\Omega})$ is a lower obstacle satisfying $\psi \leq 0$ on $\partial\Omega$, $f \in L^2(\Omega)$ is a load term and

$$K = \{v \in H_0^1(\Omega) \mid v \geq \psi \text{ a.e. in } \Omega\},$$

and

$$a(v, w) = \int_\Omega \nabla v \cdot \nabla w, \qquad v, w \in H_0^1(\Omega).$$

This problem admits a unique solution u since K is a nonempty, closed, and convex set, and $a(\cdot, \cdot)$ is $H_0^1(\Omega)$-coercive.

Adaptive solvers are now widely used in numerical simulations of lots of problems for better accuracy with minimal computational cost. The reasons for choosing adaptive method for the problem (1) are two-folded. First, the grid in the contact zone is often not necessarily as fine as that in the non-contact zone. Secondly, the solution u may have singularity in some local areas. Therefore, for obstacle problems, a finite element solution on a suitable non-uniform grid may approximate the exact solution much better than that on a uniform grid with the same number of *degrees of freedoms*. Solvers which can generate non-uniform grids adaptively according to the *problem to-be-solved* are desired.

The adaptive solver in this paper will be established based on a *near-optimal* hierarchical estimate. Note that the hierarchical a posteriori analysis can be traced back to the pioneering works [2, 7, 20] and the monographs [1, 19]. The hierarchical analysis for obstacle problems have been studied in [13, 18, 22]. The hierarchical estimate presented in this paper improves the results in [22] by estimating directly

the energy norm of the discretization error, instead of the energy functional of the discretization error in [22].

This paper is organized as follows. In Sect. 2, we present a *near-optimal* hierarchial estimate for obstacle problems. A detailed description of our adaptive method will be presented in Sect. 3. In Sect. 4, numerical experiments will be given to show that our algorithm has the optimal convergence rate.

Throughout this paper, "$A \lesssim B$" means that A can be bounded by B multiplied with a generic constant depending only on the shape regularity of the underlying grid, "$A \simeq B$" means "$A \lesssim B$" and "$B \lesssim A$".

2 A Near-Optimal Hierarchical Error Estimate

Let $\mathcal{T}$ be a conforming and shape regular triangulation of Ω with $\mathcal{N}$ and let $\mathcal{E}$ denote the set of all vertices and interior edges, respectively. We introduce the space $\mathcal{S} \subset H_0^1(\Omega)$ of piecewise linear finite elements on $\mathcal{T}$ spanned by the nodal basis $\{\phi_p \,|p \in \mathcal{N} \cap \Omega\}$. The finite element discretization of (1) reads as

$$u_{\mathcal{S}} \in K_{\mathcal{S}}: \quad a(u_{\mathcal{S}}, v - u_{\mathcal{S}}) \geq (f, v - u_{\mathcal{S}}) \quad \forall v \in K_{\mathcal{S}} \tag{2}$$

where the discrete constraints set

$$K_{\mathcal{S}} = \{v \in \mathcal{S} \,|\, v(p) \geq \psi(p) \quad \forall p \in \mathcal{N}\}.$$

Note that $K_{\mathcal{S}} \subset K$, if $\psi \in \mathcal{S}$.

We define the *residual type* functional $\sigma_{\mathcal{S}}$ by

$$\langle \sigma_{\mathcal{S}}, v\rangle = (f, v) - a(u_{\mathcal{S}}, v) = \int_\Omega fv + \sum_{E\in\mathcal{E}} \int_E j_E v, \ \forall v \in H_0^1(\Omega)$$

where $j_E = \partial_{\mathbf{n}} u_{\mathcal{S}}|_{\tau_2} - \partial_{\mathbf{n}} u_{\mathcal{S}}|_{\tau_1}$ denotes the jump of the normal flux across the common edge $E = \tau_1 \cap \tau_2$ of two triangles $\tau_1, \tau_2 \in \mathcal{T}$ and $\mathbf{n}$ denotes the normal vector on E pointing from τ_1 to τ_2. For all $E \in \mathcal{E}$, let ϕ_E be the piecewise affine function characterized by $\phi_E(p) = \delta_{x_E,p}$ for all $p \in \mathcal{N}' = \mathcal{N} \cup \{x_{E'} | E' \in \mathcal{E}\}$, here $x_{E'}$ is the midpoint of E'. We define

$$\rho_E = \langle \sigma_{\mathcal{S}}, \phi_E\rangle \|\phi_E\|^{-1}, E \in \mathcal{E}$$

and will use $|\rho_E|$ as our error indicators. Note that the similar edge-oriented indicators have been introduced in the hierarchical estimate for variational equations [2, 5], and for variational inequalities [11, 13, 18, 22]. Not all the $\rho_E, E \in \mathcal{E}$ are efficient. To determine the efficient ρ_E, we let

$$\mathcal{N}^\bullet = \{p \in \mathcal{N} \big| u_{\mathcal{S}}(p) = \psi(p) \text{ or } p \in \partial\Omega\}, \quad \mathcal{N}^\circ = \{p \in \mathcal{N} \cap \Omega \big| u_{\mathcal{S}}(p) > \psi(p)\}$$

respectively be the set of contact and non-contact nodes and

$$\mathcal{E}^\bullet = \{E \in \mathcal{E} | \mathcal{N}_E \subset \mathcal{N}^\bullet\}, \quad \mathcal{E}^\circ = \{E \in \mathcal{E} | \mathcal{N}_E \cap \mathcal{N}^\circ \neq \emptyset\}$$

be the set of contact and non-contact edges, where $\mathcal{N}_E = \mathcal{N} \cap E$ be the nodes on $E, E \in \mathcal{E}$. Moreover we define $\mathcal{E}^+ := \{E \in \mathcal{E} \big| \rho_E \geq 0\}$ and let

$$\mathcal{E}_1 = \mathcal{E} \setminus \mathcal{E}_2, \quad \mathcal{E}_2 = \mathcal{E}^\circ \cup \mathcal{E}^+.$$

The indicators $|\rho_E|, E \in \mathcal{E}_2$ are efficient.

The second type of indicators are patch-oriented quantities $\rho_p = \|h_p f\|_{0,\omega_p}, p \in \mathcal{N}$, where the patch $\omega_p = \text{supp } \phi_p$. We define

$$\mathcal{N}_2 = \{p \in \mathcal{N} | f_p \geq 0\} \cup \mathcal{N}^\circ.$$

The indicators ρ_p are efficient for all $p \in \mathcal{N}_2$.

To present efficiency and reliability of our hierarchial estimator, we need to split again the set of contact nodes $\mathcal{N}^\bullet$ into

$$\mathcal{N}^\bullet = \mathcal{N}_0^\bullet \cup \mathcal{N}_1^\bullet \cup \mathcal{N}_2^\bullet \cup \mathcal{N}_3^\bullet \cup \mathcal{N}_4^\bullet$$

where

$$\mathcal{N}_0^\bullet = \{p \in \mathcal{N}^\bullet \ \big| \ u_{\mathcal{S}}|_{\omega_p} = \psi|_{\omega_p}, \ f|_{\omega_p} \leq 0, \ j_E \leq 0 \ \forall E \in \mathcal{E}_p\}$$

is the set of the so-called *full-contact* nodes (c.f, [10, 18]) and

$$\mathcal{N}_1^\bullet = \{p \in \mathcal{N}^\bullet \ \big| \ u_{\mathcal{S}}|_{\omega_p} = \psi|_{\omega_p}, \ f_p \leq 0, \ j_E \leq 0 \ \forall E \in \mathcal{E}_p\} \setminus \mathcal{N}_0^\bullet,$$

$$\mathcal{N}_2^\bullet = \{p \in \mathcal{N}^\bullet \ \big| \ u_{\mathcal{S}}|_{\omega_p} = \psi|_{\omega_p}, \ f_p \leq 0, \ \exists \ E \in \mathcal{E}_p \text{ s.t. } j_E > 0\},$$

$$\mathcal{N}_3^\bullet = \{p \in \mathcal{N}^\bullet \big| u_{\mathcal{S}} > \psi \text{ in } \omega_p \setminus \{p\}\}, \mathcal{N}_4^\bullet = \mathcal{N}^\bullet \setminus (\mathcal{N}_0^\bullet \cup \mathcal{N}_1^\bullet \cup \mathcal{N}_2^\bullet \cup \mathcal{N}_3^\bullet).$$

We define our hierarchical estimator by

$$\eta^2 = \sum_{E \in \mathcal{E}_2} \rho_E^2 + \sum_{p \in \mathcal{N}_3^\bullet \cup \mathcal{N}_4^\bullet} \|h_p f\|_{0,\omega_p}^2$$

and the oscillation by

$$osc^2 = \sum_{p \in \mathcal{N} \setminus \mathcal{N}_0^\bullet} osc_p^2 + \sum_{p \in \mathcal{N}_3^\bullet} \|\nabla(\psi - u_{\mathcal{S}})\|_{0,\omega_p}^2$$

where the patch-oriented oscillation (c.f., [9]) is defined for all $p \in \mathcal{N}$ by

$$osc_p = \|h_p(f - f_p)\|_{0,\omega_p}.$$

Here $f_p = 0$ if $p \in \mathcal{N}_2^\bullet$ and $f_p = \frac{1}{|\omega_p|} \int_{\omega_p} f$ otherwise. Note that the oscillation defined above is smaller than that defined in [22] and it seems to be really of *high-order*.

We have the following efficiency and reliability results.

Theorem 1. *There holds the lower bound*

$$\eta \lesssim \|e\| + osc. \tag{3}$$

Moreover, if $\psi \in \mathcal{S}$, there holds the upper bound

$$\|e\| \lesssim \eta + osc. \tag{4}$$

Note that in the above theorem, the efficiency result holds for general obstacle $\psi \in C(\overline{\Omega})$ but the reliability result only holds for obstacle functions which are piecewise affine with respect to the underlying grid $\mathcal{T}$. The detailed and very complicated proof of this theorem will be given in a future paper [21].

3 An Adaptive Finite Element Method

This section is dedicated to the presentation of an adaptive finite element method for the obstacle problem (1).

The main purpose of our adaptive algorithm is to construct the sequence of triangulations $\mathcal{T}_j$, $j = 0, 1, 2, \cdots$, resulting from the jth local refinement steps of an initial triangulation $\mathcal{T}_0$. Here and throughout the paper, the subscript j will always refer to the corresponding triangulation $\mathcal{T}_j$ as, for example, in $\mathcal{N}_j,\ \mathcal{E}_j,\ \mathcal{S}_j,\ u_j,\ \psi_j$, and so on.

As a standard adaptive scheme, our adaptive algorithm consists of loops of the following four basic steps

$$Solve \ \rightarrow \ Estimate \ \rightarrow \ Mark \ \rightarrow \ Refine.$$

which will be described in the following.

Solve. To solve the discrete problem (2), we apply *monotone multigrid methods* proposed in [12] on the non-uniform grid $\mathcal{T}_j$. Our numerical implementation shows that even for non-uniform grid, this monotone multigrid method requires only $O(n_j)$ operations for each iteration, where n_j denotes the degree of freedoms of $\mathcal{T}_j$, and it converges more rapidly than the standard nonlinear Gauss–Seidel method since its convergence rate is about 1.

Given a mesh $\mathcal{T}_j$ and an initial iterate u_j^0 for the solution, the algorithm **SOLVE** computes the discrete solution

$$u_j \ := \ \mathbf{SOLVE}(\mathcal{T}_j,\ u_j^0).$$

Estimate. We use the hierarchical estimators presented in the previous section to estimate the error. For a given mesh $\mathcal{T}_j$ and the finite element approximation u_j, the subroutine **ESTIMATE** computes the edgewise hierarchical indicators ρ_E for all edges $E \in \mathcal{E}_{j2}$ and the nodalwise indicators ρ_p for all $p \in \mathcal{N}_{j3}^{\bullet} \cup \mathcal{N}_{j4}^{\bullet}$:

$$(\{\rho_E\}_{E\in\mathcal{E}_{j2}},\ \{\rho_p\}_{p\in\mathcal{N}_{j3}^{\bullet}\cup\mathcal{N}_{j4}^{\bullet}}) \ = \ \mathbf{ESTIMATE}(\mathcal{T}_j,\ u_j).$$

Mark. We use a variant of Dörfler marking strategy [8] described as below. First, we order all the quantities $\{|\rho_E|\}_{E\in\mathcal{E}_{j2}}$ and $\{\rho_p\}_{p\in\mathcal{N}_{j3}^{\bullet}\cup\mathcal{N}_{j4}^{\bullet}}$ according to their size. Secondly, proceeding from the largest to smallest quantities, we collect all entries from these two sets until they sum up to $\theta\eta_j$ where $\theta\in(0,1)$ is some given parameter. Finally, if ρ_E^2 or ρ_P^2 are contained in this collection, then all the triangles in the support of ϕ_E or ϕ_p are marked for refinement.

Given a mesh $\mathcal{T}_j$ and the indicators $(\{\rho_E\}_{E\in\mathcal{E}_{j2}},\ \{\rho_p\}_{p\in\mathcal{N}_{j3}^{\bullet}\cup\mathcal{N}_{j4}^{\bullet}})$, together with the parameter θ ,**MARK** generates a subset $\widetilde{\mathcal{T}}_j$ of $\mathcal{T}_j$:

$$\widetilde{\mathcal{T}}_j \ = \ \mathbf{MARK}(\theta,\ \mathcal{T}_j, (\{\rho_E\}_{E\in\mathcal{E}_{j2}},\ \{\rho_p\}_{p\in\mathcal{N}_{j3}^{\bullet}\cup\mathcal{N}_{j4}^{\bullet}})).$$

Refine. We will use the so-called *newest vertex bisection* techniques to refine the mesh $\mathcal{T}_j$: first we label one vertex of each triangle in $\mathcal{T}_j$ as the *newest vertex*, the opposite edge of the newest vertex is called *reference edge*. After being labeled, each the element $\tau\in\widetilde{\mathcal{T}}_j$ is then bisected to two new children elements by connecting the newest vertex to the midpoint of the reference edge. After all the marked triangles are bisected, more bisections are necessary to eliminate the hanging nodes (cf.,[3, 4, 15]). It is worth mentioning that here, each marked triangle is refined only once and consequently, the interior node property[14, 17] has been circumvented.

Given a mesh $\mathcal{T}_j$ and a marked set $\widetilde{\mathcal{T}}_j$, **REFINE** constructs the conforming and shape regular triangulation $\mathcal{T}_{j+1}$:

$$\mathcal{T}_{j+1} \ = \ \mathbf{REFINE}(\mathcal{T}_j,\ \widetilde{\mathcal{T}}_j).$$

Now we are ready to present our adaptive finite element methods for (1) which consists of the loops of the above four subroutines **SOLVE**, **ESTIMATE**, **MARK**, and **REFINE**, consecutively. Given an initial triangulation $\mathcal{T}_0$, a tolerance $\varepsilon>0$ and a parameter $0<\theta<1$, our adaptive solver can be described as below:

$u_{\mathrm{FE}} = \mathbf{AFEM4OP}(\mathcal{T}_0,\varepsilon,\theta)$

Set $u_0=0$, for $j=1,\ldots$, do the following :

1. Set $u_j^0 \ = \ u_{j-1}$, then $u_j = \mathbf{SOLVE}(\mathcal{T}_j,\ u_j^0)$.
2. $(\{\rho_E\}_{E\in\mathcal{E}_{j2}},\ \{\rho_p\}_{p\in\mathcal{N}_3^{\bullet}\cup\mathcal{N}_4^{\bullet}}) \ = \ \mathbf{ESTIMATE}(\mathcal{T}_j,\ u_j)$.
3. Compute η_j. If $\eta_j\le\varepsilon$, $u_{\mathrm{FE}}=u_j$, stop. Otherwise, go to Step 4.
4. $\widetilde{\mathcal{T}}_j \ = \ \mathbf{MARK}(\theta,\ \mathcal{T}_j, (\{\rho_E\}_{E\in\mathcal{E}_{j2}},\ \{\rho_p\}_{p\in\mathcal{N}_3^{\bullet}\cup\mathcal{N}_4^{\bullet}}))$.
5. $\mathcal{T}_{j+1} \ = \ \mathbf{REFINE}(\mathcal{T}_j,\ \widetilde{\mathcal{T}}_j)$. Set $j=j+1$, go to Step 1.

4 Numerical Experiments

In our numerical experiments, we will test two examples to verify if our algorithm **AFEM4OP** has a quasi-optimal convergence rate in terms of the number of degrees of freedom. Note that for adaptive finite element methods for elliptic PDEs, the optimal convergence rate has been obtained both theoretically and numerically in recent papers such as [6, 16, 17].

Example 1 Constant Obstacle. Let the obstacle function $\psi \equiv 0$, the domain $\Omega = (-1,1)^2$, and the radially symmetric right-hand side

$$f(x) = \begin{cases} -8r^2, & |x| > r \\ -8(2|x|^2 - r^2)), & |x| \le r \end{cases}.$$

This problem has the unique radially symmetric exact solution

$$u(x) = (\max\{r^2 - |x|^2, 0\})^2.$$

For simplicity, we select $r = 0.7$ in our numerical computations. Then the circle $\{x \in \Omega | r = 0.7\}$ is the free boundary of the problem which decompose the domain to the contact zone $\{x \in \Omega || r > 0.7\}$ and the non-contact zone $\{x \in \Omega || r < 0.7\}$.

In our numerical experiments, the initial triangulation $\mathcal{T}_0$ consisting of four congruent triangles. Selecting $\theta = 0.5$ in **AFEM4OP**, we obtain the sequence of triangulations $\mathcal{T}_j, j = 0, 1, \ldots, 14$. The left picture of Fig. 1 presents the discretization error

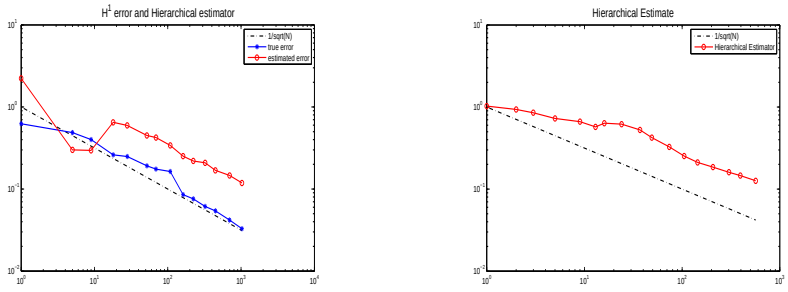


Fig. 1. *Left*: the exact error and the hierarchial estimator for Example 1, *Right*: the estimator for Example 2.

$\|u - u_{\mathcal{S}_j}\|$ and the hierarchical estimator η_j over the number $n_j = \#\mathcal{T}_j$. Obviously, both the exact error and the hierarchical estimator have an optimal convergence rate of $\mathcal{O}(n_j^{-1/2})$.

Example 2 Lipschitz Obstacle. We consider (1) with $\Omega = \{x \in \mathbb{R}^2 \mid |x_1| + |x_2| < 1\}$, the right hand side $f = -5$ and the Lipschitz obstacle

$$\psi(x) = -\mathrm{dist}(x, \partial\Omega).$$

As in the first example, we apply the algorithm AFEM4OP to obtain a sequence of triangulations $\mathcal{T}_j$, $j = 1, 2, \ldots, 18$ based upon the initial triangulation $\mathcal{T}_0$ consisting of four congruent triangles. The adaptive parameter $\theta = 0.35$. As no exact solution is available, the final approximate solution u_{18} is depicted in the left picture of Fig. 2 while the right picture shows the grid in the 18th iteration step. The

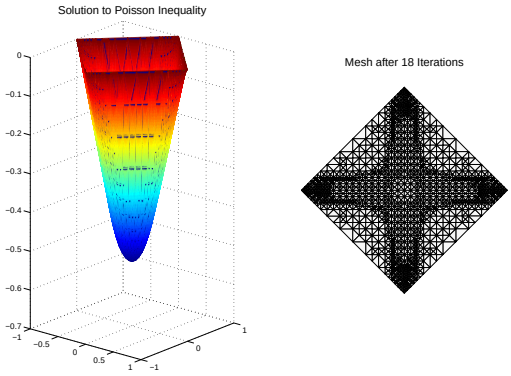


Fig. 2. Approximate solution u_{18} and the grid in the 18th adaptive iteration step.

triangulation is locally refined in the neighborhood of the free boundary which is in agreement with the corresponding lack of regularity. The hierarchical estimator is presented in the right picture of 1. We still observe that $\eta_{\mathcal{T}_j} = \mathcal{O}(n_j^{-1/2})$.

References

1. M. Ainsworth and J.T. Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Wiley, New York, NY, 2000.
2. R.E. Bank and R.K. Smith. A posteriori error estimates based on hierarchical bases. *SIAM J. Numeric Anal.*, 30:921–935, 1993.
3. E. Bänsch. Local mesh refinement in 2 and 3 dimensions. *IMPACT Comput. Sci. Eng.*, 3: 181–191, 1991.
4. P. Binev, W. Dahmen, and R. DeVore. Adaptive finite element methods with convergence rates. *Numer. Math.*, 97(2):219–268, 2004.
5. F.A. Bornemann, B. Erdmann, and R. Kornhuber. A posteriori error estimates for elliptic problems in two and three space dimensions. *SIAM J. Numer. Anal.*, 33:1188–1204, 1996.
6. J.M. Cascon, C. Kreuzer, R.H. Nochetto, and K.G. Siebert. Quasi-optimal convergence rate for an adaptive finite element method. *SIAM J. Numer. Anal.*, 46(5):2524–2550, 2008.
7. P. Deuflhard, P. Leinen, and H. Yserentant. Concepts of an adaptive hierarchical finite element code. *IMPACT Comput. Sci. Eng.*, 1:3–35, 1989.
8. W. Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM J. Numer. Anal.*, 33:1106–1124, 1996.
9. W. Dörfler and R.H. Nochetto. Small data oscillation implies the saturation assumption. *Numer. Math.*, 91:1–12, 2002.
10. F. Fierro and A. Veeser. A posteriori error estimators for regularized total variation of characteristic functions. *SIAM J. Numer. Anal.*, 41(6):2032–2055 (electronic), 2003. ISSN 1095-7170.
11. R. Kornhuber. A posteriori error estimates for elliptic variational inequalities. *Comput. Math. Appl.*, 31:49–60, 1996.

12. R. Kornhuber. *Adaptive Monotone Multigrid Methods for Nonlinear Variational Problems*. Teubner, Stuttgart, 1997.
13. R. Kornhuber and Q. Zou. Efficient and reliable hierarchical error estimates for the discretization error of elliptic obstacle problems. *Math. Comput.*, in press, 2010.
14. K. Mekchay and R. Nochetto. Convergence of adaptive finite element methods for general second order linear elliptic PDE. *SIAM J. Numer. Anal.*, 43:1803–1827, 2005.
15. W.F. Mitchell. *Unified Multilevel Adaptive Finite Element Methods for Elliptic Problems*. PhD thesis, University of Illinois at Urbana-Champaign, 1988.
16. P. Morin, R.H. Nochetto, and K.G. Siebert. Data oscillation and convergence of adaptive fem. *SIAM J. Numer. Anal.*, 38:466–488, 2000.
17. P. Morin, R.H. Nochetto, and K.G. Siebert. Convergence of adaptive finite element methods. *SIAM Rev.*, 44:631–658, 2002.
18. K.G. Siebert and A. Veeser. A unilaterally constrained quadratic minimization with adaptive finite elements. *SIAM J. Optim.*, 18:260–289, 2007.
19. R. Verfürth. *A Review of a Posteriori Error Estimation and Adaptive Mesh–Refinement Techniques*. Wiley-Teubner, Chichester, 1996.
20. O.C. Zienkiewicz, J.P. De, S.R. Gago, and D.W. Kelly. The hierarchical concept in finite element analysis. *Comput. Struct.*, 16:53–65, 1983.
21. Q. Zou. A near-optimal hierarchial analysis for elliptic obstacle problems. *In preparation*.
22. Q. Zou, A. Veeser, R. Kornhuber, and C. Gräser. Hierarchical error estimates for the energy functional in obstacle problems. *Numer. Math.*, submitted, 2009.

Efficient Parallel Preconditioners for High-Order Finite Element Discretizations of $H(grad)$ and $H(\mathrm{curl})$ Problems

Junxian Wang[1], Shi Shu[1] and Liuqiang Zhong[2]

[1] School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, P.R. China, xianxian.student@sina.com; shushi@xtu.edu.cn
[2] School of Mathematics Sciences, South China Normal University, Guangzhou 510631, P.R. China, zhonglq@xtu.edu.cn

Summary. In this paper, we study preconditioning techniques for both $H(grad)$ and $H(\mathbf{curl})$ problems. For an $H(grad)$ elliptic problem discretized by high-order finite elements with a hierachical basis of k-th order, we design and analyse an parallel AMG preconditioner based on a two-level method and a block Gauss–Seidel smoothing technique. For an $H(\mathbf{curl})$ elliptic problem discretized by high-order edge finite elements, we design a parallel solver based on an auxiliary space preconditioner. Numerical experiments show that the number of iteration for the corresponding PCG methods does not depend on the mesh size, and depend weakly on the order of the finite element space and jumps of the coefficients.

Key words: high-order finite element, jump coefficients, parallel preconditioner, block Gauss-Seidel smoother, auxiliary space preconditioner

1 Introduction

Many problems in fields such as computational electromagnetics and computational fluid dynamics can be essentially be transformed into computations of $H(grad)$ and $H(\mathbf{curl})$ elliptic problems. The finite element method is one of the most commonly used numerical methods for solving these kinds of problems. Efficient parallel algorithms are of great importance because the resulting discrete systems are usually large and the corresponding condition numbers grow rapidly with the refinement of the mesh. High-order finite element methods are of great practical interests because of their better approximation to the analytical solution. But the resulting algebraic systems of equations are more difficult to solve than those derived from linear finite elements.

The classical algebraic multigrid (AMG) method is quite efficient for solving the algebraic system resulting from the discretization of $H(grad)$ elliptic problems discretized by linear finite elements. But it does not work well for the systems obtained

by the high-order finite elements. The reason is that the algebraic mesh graph corresponding to the stiffness matrix obtained by the high-order finite element is quite different from the geometric mesh graph. There exist some improved AMG methods for solving the linear system resulting from high-order finite elements ([2, 4, 6, 7]). But, so far, they are mainly restricted to the smooth coefficient case, and the number of iterations depends on the order of the finite element space. For an $H(\mathbf{curl})$ elliptic problems, [3] designed an auxiliary space preconditioner for the algebraic systems resulting from the linear edge elements of the first type of Nédélec elements. Reference [9] further studied the preconditioners for high-order edge finite element discretizations. Recently, the study of parallel algorithms have attracted more and more researchers' attention, and the corresponding numerical softwares has developed rapidly. The high performance parallel preconditioner package HYPRE is one of the most popular numerical software in the world. BoomerAMG and AMS in HYPRE are two efficient solvers for the $H(grad)$ and $H(\mathbf{curl})$ elliptic problems, discretized by linear elements, respectively.

In this paper, we first consider an $H(grad)$ elliptic problem with jump coefficients. Here we discuss the parallel PCG method for solving the algebraic system derived from high-order finite element based on hierachical bases (HBk). Using a two-level method and a block Gauss–Seidel smoothing technique, we have designed an efficient parallel AMG preconditioner and a corresponding PCG method, and implemented them using HYPRE. Then for an $H(\mathbf{curl})$ elliptic problem discretized by high-order edge finite elements, we have designed a parallel PCG algorithm, whose key idea (see [8]) is to change the construction of the parallel preconditioner for the $H(\mathbf{curl})$ system into the design of several preconditioners for the $H(grad)$ system.

The numerical results indicate that these new algorithms have good algorithmic scalability and the number of iterations does not depend on the mesh size, and depend weakly on the order of the finite element space and the jumps of the coefficients.

2 A Parallel Preconditioner for the $H(grad)$ System

Let Ω be a simply connected polyhedron in $\mathbb{R}^3$. We consider the variational problem: Find $u \in H_0^1(\Omega)$, such that

$$a(u, v) = (f, v),\ \forall v \in H_0^1(\Omega), \tag{1}$$

where $a(u,v) = \int_\Omega \beta \nabla u \cdot \nabla v dx$, $(f,v) = \int_\Omega f v dx$, and $\beta(x)$ is a bounded positive function, which may contain large discontinuous jumps, and $f \in L^2(\Omega)$.

We discretize the continuous variational problem (1) with k-th order hierachical basis $\{\psi^k, k = 1, \cdots, n_k\}$ proposed in [1], where $n_k = dim(Z_h^k)$ and Z_h^k is the k-th order Lagrangian finite element space. Furthermore we denote the corresponding algebraic system as

$$A_k u_k = f_k. \tag{2}$$

2.1 A Parallel AMG Preconditioner

Let np be the number of processors, A_j^k, u_j^k and f_j^k be the restrictions of A_k, u_k and f_k on the j-th processor, respectively, and n_j^k be the number of rows of A_j^k. Here, we always assume that the indices of first n_j^1 rows of A_j^k corresponds to the indices of the linear element basis. For the l-th component of u_j^k, we call l the local index on the j-th processor, and let the location in u_k be the global index. We also say that the global index belongs to the j-th processor.

We use a two-level method for solving (2), which translates the solving of linear system discretized by high order finite element into the solving of the one corresponding to the linear finite element. We need to generate the parallel matrix of linear finite element A_1 that consists of elements whose row and column indices is correspond to the linear element basis of A_k.

We change the traditional smoother to the block Gauss–Seidel smoother, since the iteration number of a two level-method with the traditional smoother depends on the order of the finite element space. Next, we generate the needed datum for the block Gauss–Seidel smoother. And we need to introduce some notation related to the j-th processor.

(i) Let $W_j = \{1, \cdots, n_j^k\}$ be the local index set for the j-th processor, and V_j be the corresponding global index set. For any $i \in W_j$, we denote

$$S_i = \{l | A_j^k(i,l) > \theta \max_{m \neq i_g} |A_j^k(i,m)|\}, \tag{3}$$

as the i-th block index set, where $\theta \in [0,1]$, $A_j^k(i,m)$ is an element of A_j^k in row i and column m, i_g is the global index of i.

(ii) Mapping array: $g(i), i = 1, \cdots, n_j^k$ where $g(i) = 0$ *or* i, which indicates whether a block smoothing is needed for the $g(i)$-th block index set.

(iii) The expanded matrix $\tilde{A}_j^k$ on the j-th processor is defined as the submatrix formed by those rows of A_k whose indices belong to $\bigcup_{g(i)\neq 0} S_{g(i)}$. The expanded vector $\tilde{f}_j^k$ can be obtained similarly.

Then we can generate the mapping array $g(i)$ and the block index set $S_{g(i)}$ $(for\ g(i) > 0)$, $i = 1, \cdots, n_j^k$ according to some suitable rule as in Algorithm 1.

Algorithm 1 *(generate the mapping array and the block index set)*

Step 1 *Initialize the mapping array* $g(i) = 0,\ i = 1, \cdots, n_j^k$.

Step 2 *Let* $g(i) = i$, *and generate* $S_{g(i)}$ *according to (3),* $i = 1, \cdots, n_j^1$.

Step 3 *Let* $H = V_j \backslash (V_j \cap \bigcup_{i=1}^{n_j^1} S_{g(i)})$,

Step 4 *For a given* $i \in H$, *find its corresponding local index* $i_l \in W_j$. *Set* $g(i_l) = i_l$, *and generate* $S_{g(i_l)}$.

Step 5 *If* $H := H \backslash (H \cap \bigcup_{g(i)\neq 0} S_{g(i)})$ *is nonempty, then goto Step 4.*

In Algorithm 2, we generate the expanded matrix $\tilde{A}_j^k$ and the expanded vector $\tilde{f}_j^k$ on the j-th processor in order to reduce the communication when the block Gauss–Seidel iteration has been carried out.

Algorithm 2 *(generate $\tilde{A}_j^k$ and $\tilde{f}_j^k$)*

Step 1 *Set $\tilde{A}_j^k := A_j^k$. Let $\tilde{S} = \bigcup_{g(i)\neq 0} S_{g(i)} = V_j \bigcup \tilde{S}_0$, where $\tilde{S}_0$ denotes the set of global indices belonging to those neighboring processors.*

Step 2 *Loop through all the neighboring processors, let $\tilde{S}_0^m$ be the set of those element in $\tilde{S}_0$ which is on the current neighboring processor m.*

2.1 *Form $A_m^0(f_m^0)$ by those rows of $A_m^k(f_m^k)$ whose global indices belongs to $\tilde{S}_0^m$. Then send $A_m^0(f_m^0)$ to processor j.*

2.2 *On processor j, receive $A_m^0(f_m^0)$ and append to $\tilde{A}_j^k(\tilde{f}_j^k)$.*

Step 3 *Generate the mapping array $p(n), n = 1, \cdots, |\tilde{S}|$, which is from the set of row numbers of $\tilde{A}_j^k$ to its corresponding global index set.*

Now, using $\tilde{A}_j^k$ and $\tilde{f}_j^k$, we can generate all the block diagonal matrices $A_{j,k}^{l,d}$, nondiagonal matrices $A_{j,k}^{l,nd}$ and right hand side vectors $f_{j,k}^{l,d}$ on processor j when $g(l) \neq 0$. Then, we construct the parallel block Gauss–Seidel smoothing algorithm as follows.

Algorithm 3 *(block Gauss–Seidel iteration)*

Assume that $\tilde{u}_j^k$ is the result from the block Gauss–Seidel iteration on the current processor j. Let S_{max} be the maximum iteration number.

Step 1 *Let the initial guess be $\tilde{u}_j^k = 0$ and $sm = 0$.*

Step 2 *Let $sm = sm+1$. If $sm > 1$, then update $\tilde{u}_j^k$ for those components belonging to its neighboring processors.*

Step 3 *For any $l \in \{1, \cdots, n_j^k\}$ with $g(l) \neq 0$, solve $A_{j,k}^{l,d} u_{j,k}^{l,d} = f_{j,k}^{l,d} - A_{j,k}^{l,nd} \tilde{\tilde{u}}_j^k$ for $u_{j,k}^{l,d}$, where $\tilde{\tilde{u}}_j^k$ is a zero extension of $\tilde{u}_j^k$, and update $\tilde{u}_j^k$ by $u_{j,k}^{l,d}$.*

Step 4 *If $sm < S_{max}$, then goto step 2.*

Step 5 *Use those components of $\tilde{u}_j^k$, whose indices belong to processor j, to form the vector $\tilde{u}_j^{k,s}$, then gather $\tilde{u}_j^{k,s}$ to form the parallel solution $\tilde{u}_k^{(s)}$.*

Based on the above block Gauss–Seidel smoother, we develop the corresponding parallel two-level method in Algorithm 4 for solving (2). Let it_{max} denote the maximum iteration number, and tol denote the stopping criteria.

Algorithm 4 *(Two-Level method)*

Step 1 *For a given initial guess $u_k^{(0)}$, use HYPRE to get an initial residual vector $r_k^{(0)} = f_k - A_k u_k^{(0)}$, and let $res0 = \|r_k^{(0)}\|$, $l = 1$.*

Step 2 *Presmoother: Use Algorithm 3 for (2) with $S_{max} = m_1$ and the initial guess $u_k^{(l-1)}$ to get the parallel iterative vector u_t.*

Step 3 *Correction*

3.1 Use HYPRE to get the vector $r_k = f_k - A_k u_t$*, and get the parallel vector* f_1 *from the first* n_j^1 *components of* r_k *on the* j*-th processor.*

3.2 Use BoomerAMG to solve $A_1 e = f_1$*, and obtain the solution* e_t*.*

3.3 Add e_t *and* u_t *to get a new parallel vector* $\tilde{u}_t$*.*

Step 4 Postsmoother: Use Algorithm 3 for (2) with $S_{max} = m_1$ *and initial guess* $\tilde{u}_t$ *to get the parallel iterative vector* $u_k^{(l)}$*.*

Step 5 Use HYPRE to get $res = \|f_k - A_k u_k^{(l)}\|$*. If* $\frac{res}{res0} < tol$ *or* $l = it_{max}$*, then exit. Otherwise, set* $l := l + 1$*, and goto step2.*

Combining the above algorithms, we obtain a new parallel AMG method for solving the system (2) resulting from the $H(grad)$ problem.

In Algorithm 4, by setting $it_{max} = 1$, we can obtain a TLB-AMG-p preconditioner for solving (2), and denote the PCG method using TLB-AMG-p as a preconditioner for TLB-AMG-CG-p.

2.2 Numerical Experiments

Consider the model problem (1) with $\Omega = (0,1)^3$, $\beta = \beta_0$ in Ω_J and $\beta = 1$ in $\Omega \backslash \Omega_J$, where β_0 is a positive constant, and Ω_J is a subdomain of Ω.

We decompose Ω into $n \times n \times n$ hexahedra $\Omega_k (k = 1, \cdots, n^3)$. All the hexahedra constitute the desired domain decomposition. To get the final triangulation of Ω, we decompose each hexahedra mentioned above into $m \times m \times m$ smaller hexahedra and divide each smaller hexahedra into 6 tetrahedra. All the tetrahedra constitute the triangulation $\mathcal{T}_h = n(m)$ of the domain Ω.

We like to remark that each subdomain Ω_k corresponds to one processor.

Example 1. (A floating subdomain case) Let $n = 3$ and $\Omega_J = (\frac{1}{3}, \frac{2}{3})^3$.

Example 2. (No floating subdomain case) Let $n = 2$ and $\Omega_J = (0, \frac{1}{2})^3$.

We apply TLB-AMG-CG-p to solve the system (2) corresponding to Example 1. In our test, we set $S_{max} = 2$ in Algorithm 3, one V-cycle in BoomerAMG, $\theta = 0.0$ in (3).

Both in Tables 1 and 2, the data below " k" is the order of finite element basis, the " $n(m)$" row denotes the tetrahedron meshes, the datum in the row of " β_0" are the values of the coefficient β in Ω_J.

The numerical results for Example 1 are presented in Table 1. A relative reduction of the preconditioned norm of the residual vector by a factor of 10^{-8} was used as termination criterion.

Table 1. No. of iterations for Example 1 with TLB-AMG-CG-p.

$n(m)$	3(4)			3(6)		
β_0	10^{-5}	1	10^5	10^{-5}	1	10^5
$k=2$	4	4	5	4	5	6
$k=3$	5	5	6	5	5	6
$k=4$	5	6	9	5	5	9

From Table 1, we find that the iteration number of TLB-AMG-CG-p does not depend on the mesh size, depends weakly on the order of basis function and the jumps of the coefficients for a floating subdomain case.

In Table 2, we present the numerical results obtained by BoomerAMG-CG(with BoomerAMG as a preconditioner) for Example 2.

Table 2. No. of iterations for Example 2 with BoomerAMG-CG.

$n(m)$	$2(4)$			$2(6)$		
β_0	1	10^3	10^5	1	10^3	10^5
$k=2$	65	101	88	72	43	42
$k=3$	16	16	16	15	15	15
$k=4$	29	28	27	27	25	25
$k=5$	>200	>200	>200	>200	>200	>200

In view of Table 2, we find that BoomerAMG-CG is sensitive to the mesh size, the order of basis functions and the jumps of the coefficients for the non-floating subdomain case. So our TLB-AMG-CG-p is more robust and efficient.

3 A Parallel Preconditioner for the $H(\mathrm{curl})$ Problem

Consider the variational problem: Find $\boldsymbol{u} \in H_0(\mathbf{curl}; \Omega)$, such that

$$a(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v}),\ \forall \boldsymbol{v} \in H_0(\mathbf{curl}; \Omega), \tag{4}$$

where $a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega (\alpha \nabla \times \boldsymbol{u} \cdot \nabla \times \boldsymbol{v} + \beta \boldsymbol{u} \cdot \boldsymbol{v}) dx,\ (\boldsymbol{f}, \boldsymbol{v}) = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} dx, \alpha(\boldsymbol{x}), \beta(\boldsymbol{x})$ are two bounded positive functions, and $\boldsymbol{f} \in (L^2(\Omega))^3$.

Use the finite element space $\boldsymbol{V}_h^{k,j} (k \geq 1, j = 1, 2)$ for problem (4), where $\boldsymbol{V}_h^{k,1}$ and $\boldsymbol{V}_h^{k,2}$ are the k^{th}-order element spaces of the first family and second family of Nédélec elements, respectively(see [5]), then we obtain a linear system

$$A_h^{k,j} U_h^{k,j} = F_h^{k,j}. \tag{5}$$

Below we develop a parallel PCG algorithm for solving (5).

3.1 A Parallel Preconditioner for (5)

Let $B_h^{k,j}$ be the HX-ho preconditioner for $A_h^{k,j}$ (see [8]). As an example, we take $k = 1$ and $j = 2$. Let $I_h^l : \boldsymbol{V}_h^{1,1} \mapsto Z_{h,l} (l = 1, 2, 3)$, $I_h^{g,2} : \boldsymbol{V}_h^{1,2} \mapsto \nabla Z_h^2$ be the corresponding interpolation matrices where $(Z_h^1)^3 := \sum_{l=1}^3 Z_{h,l}$. Given a vector y, we can give an algorithm for solving $w = B^{1,2} y$ as Algorithm 5.

Algorithm 5 *(HX-ho preconditioner)*

Step 1 (Setup):

1.1 Form (in parallel) matrices I_h^l, $I_h^{g,2}$ and get the matrix $A^{1,1}$ with $A^{1,2}$;

1.2 Generate matrices $A_h^l = I_h^l A^{1,1}(I_h^l)^T, l = 1,2,3$ *and* $A_h^{g,2} = I_h^{g,2} A^{1,2} (I_h^{g,2})^T$.

Step 2 (Solve):

2.1 Use zero as an initial guess, and apply m *times parallel symmetric Gauss–Seidel iteration for* $A^{1,2}\tilde{u} = y$ *to obtain the solution* $\tilde{u}$*;*

2.2 Generate a vector y_1 *by extracting those* y *components which belong to* $\boldsymbol{V}_h^{1,1}$*. Compute* $u_l = B_h^l I_h^l y_1 (l = 1,2,3)$ *where* B_h^l *can be chosen as the TLB-AMG-p* ($k = 1$) *preconditioner of* A_h^l*, calculate the vector* u_l' *by zero extension of* $(I_h^l)^T u_l$*;*

2.3 Compute $\tilde{u}_{g,2} = B_h^{g,2} I_h^{g,2} y$ *where* $B_h^{g,2}$ *can be chosen as the TLB-AMG-p* ($k = 2$) *preconditioner of* $A_h^{g,2}$*;*

2.4 Construct $w = \tilde{u} + \sum_{i=1}^{3} u_l' + (I_h^{g,2})^T \tilde{u}_{g,2}$.

We denote HX-ho-CG-p for the PCG method using Algorithm 5 as a preconditioner. Below we present some examples to show the efficiency and robustness of HX-ho-CG-p.

3.2 Numerical Results

We consider the model problem (4) with $\Omega = (0,1)^3$, $\alpha = \beta = \beta_0$ in Ω_J and $\alpha = \beta = \beta_0$ in $\Omega \backslash \Omega_J$, where β_0 is a positive constant, and $\Omega_J \subset \Omega$.

Example 3. (No floating subdomain case) $\Omega_J = (0, \frac{1}{3})^3, \beta_0 = 10^5$.

Example 4. (A floating subdomain case) $\Omega_J = (\frac{1}{3}, \frac{2}{3})^3, \beta_0 = 10^5$.

The numerical results for the two examples are reported in Table 3, where the datum in the column "np" denote the number of processors, the number $n(m)$ denote the tetrahedral meshes, and the datum left in the column of "$n(m)$" are the corresponding iteration number of HX-ho-CG-p.

A relative reduction of the preconditioner norm of the residual vector by a factor of 10^{-6} was used as termination criterion in HX-ho-CG-p. In our test, we set $m = 3$ in Algorithm 5 and the block Gauss–Seidel iteration is changed to 15 Jacobi iterations with weight $\omega = 0.45$ as a smoother in TLB-AMG-p.

Table 3. No. of iterations for the two examples with HX-ho-CG-p.

np	Example 3			Example 4		
	3(2)	3(4)	3(8)	3(2)	3(4)	3(8)
1	15	16	16	19	21	19
4	16	17	17	24	22	21
8	17	17	17	29	22	21

From Table 3, we find that the new algorithm HX-ho-CG-p has good algorithmic scalability and that the number of iterations is independent of the mesh size, and weakly dependent on the jumps of the coefficients.

Acknowledgments The authors are partially supported by the National Natural Science Foundation of China(Grant No. 10771178), the National Basic Research Program of China(Grant No. 2005CB321702), Key Project of Chinese Ministry of Education(Grant No. 208093). The first author is supported by Hunan Provincial Innovation Foundation For Postgraduate(Grant No. CX2009B121)

References

1. M. Ainsworth and J. Coyle. Hierarchic finite element bases on unstructured tetrahedral meshes. *Int. J. Numer. Methods Eng.*, 58:2103–2130, 2003.
2. J.J. Heys, T.A. Manteuffel, S.F. McCormick, and L.N. Olson. Algebraic multigrid for higher-order finite elements. *J. Comput. Phys.*, 204(2):520–532, 2005.
3. R. Hiptmair and J. Xu. Nodal auxiliary spaces preconditions in $H(\mathbf{curl})$ and $H(div)$ spaces. *SIAM J. Numer. Anal.*, 45(6):2483–2509, 2007.
4. Y. Huang, S. Shu, and X. Yu. Preconditioning higher order finite element systems by algebraic multigrid method of linear elements. *J. Comput. Math.*, 24(5):657–664, 2006.
5. P. Monk. *Finite Element Methods for Maxwell Equations*. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford, Colorado, USA 2003.
6. J. Ruge. AMG for higher-order discretizations of second-order elliptic problems. In *Eleventh Copper Mountain Conference on Multigrid Methods*, 2003.
7. S. Shu, D. Sun, and J. Xu. An algebraic multigrid method for higher-order finite element discretizations. *Computing*, 77(4):347–377, 2006.
8. J. Xu, L. Chen, and R. Nochetto. *Optimal multilevel methods for $H(grad)$, $H(\mathbf{curl})$ and $H(div)$ systems on graded and unstructured grids*. Summer School, Peking University, July 13, 2009–August 8, 2009, 2009.
9. L. Zhong, S. Shu, D. Sun, and L. Tan. Preconditioners for higher order edge finite element discretizations of Maxwell's equations. *Sci. China Ser. A*, 51(8):1537–1548, 2008.

Part III

Contributed Presentations

A Simple Uniformly Convergent Iterative Method for the Non-symmetric Incomplete Interior Penalty Discontinuous Galerkin Discretization

Blanca Ayuso[1] and Ludmil T. Zikatanov[2]

[1] Departamento de Matemáticas, Instituto de Ciencias Matemáticas
CSIC-UAM-UC3M-UCM, Universidad Autónoma de Madrid
Madrid 28049, Spain, blanca.ayuso@uam.es
[2] Department of Mathematics, Penn State University,
University Park, PA 16802, USA, ltz@math.psu.edu

We introduce a uniformly convergent iterative method for the systems arising from *non-symmetric* IIPG linear approximations of second order elliptic problems. The method can be viewed as a block Gauß–Seidel method in which the blocks correspond to restrictions of the IIPG method to suitably constructed subspaces. Numerical tests are included, showing the uniform convergence of the iterative method in an energy norm.

1 Introduction

In recent years, domain decomposition preconditioners and multilevel methods have been developed for the efficient solution of the linear systems that arise from Discontinuous Galerkin (DG) discretizations of elliptic problems (see [5] and the references therein). While most works deal with symmetric DG methods, very little is known for preconditioning the non-symmetric ones. However, designing efficient solvers for the resulting non-symmetric linear systems is of interest since they could be used as building blocks for preconditioning DG discretizations of non-symmetric PDEs (such as convection-diffusion problems). An important distinction between non-symmetric and symmetric DG schemes (even for discretizations of selfadjoint elliptic problems) is that the convergence analysis of the iterative methods is much more involved. As shown numerically in [1], the symmetric part of the preconditioned matrix of non-symmetric DG schemes (and in particular for the IIPG discretization considered here) can be indefinite and so the classical convergence theory for GMRES (see [6]) cannot be applied and new theoretical tools are needed.

In this paper, we design an efficient solver by using a space decomposition of the DG space introduced in [3]. We also extend some of the results from that work to the case of variable diffusion coefficient. The proposed iterative method is a successive subspace correction method for the non-symmetric IIPG discretization. We

demonstrate via numerical experiments uniform convergence with respect to both the penalty parameter and the number of degrees of freedom (dofs). In addition, as we will discuss, the method considered here is more efficient than those proposed and analyzed in [3]. However, the convergence of such method, although numerically evident, is much more difficult to analyze and we do not present such analysis here.

2 Interior Penalty Discontinuous Galerkin Methods

Given $f \in L^2(\Omega)$, we consider the following model problem

$$-\nabla \cdot (\mathbb{K}\nabla u) = f \quad \text{in } \Omega\,, \quad u = 0 \quad \text{on } \partial\Omega\,, \tag{1}$$

where $\Omega \subset \mathbb{R}^d$, $d = 2, 3$ is a convex polygon or polyhedron and $\mathbb{K} \in (L^\infty(\Omega))^{2d}_s$ is a given piecewise constant permeability symmetric positive-definite tensor satisfying: $0 < \mathfrak{c}_0\|\boldsymbol{\xi}\|^2 \le \boldsymbol{\xi}^t\mathbb{K}(x)\boldsymbol{\xi} \le \mathfrak{c}_1\|\boldsymbol{\xi}\|^2 \quad \forall \boldsymbol{\xi} \in \mathbb{R}^d\ \forall\, x \in \Omega$.
Let $\mathcal{T}_h$ be a shape-regular family of partitions of Ω into d-dimensional simplexes T (triangles if $d = 2$ and tetrahedrons if $d = 3$) and let $h = \max_{T\in\mathcal{T}_h} h_T$ with h_T denoting the diameter of T for each $T \in \mathcal{T}_h$. We assume $\mathcal{T}_h$ does not contain hanging nodes and $\mathbb{K}$ is constant on each $T \in \mathcal{T}_h$.
Let V^{DG} denote the discontinuous finite element space defined by:

$$V^{DG} = \left\{ u \in L^2(\Omega) \ : \ u_{|_T} \ \in \ \mathbb{P}^1(T)\ \forall T \in \mathcal{T}_h \right\},$$

where $\mathbb{P}^1(T)$ denotes the space of linear polynomials on T. We denote by $\mathcal{E}_h^o$ and $\mathcal{E}_h^\partial$ the sets of all interior faces and boundary faces (edges in $d = 2$), respectively, and we set $\mathcal{E}_h = \mathcal{E}_h^o \cup \mathcal{E}_h^\partial$. Following [2], we define the *average* and *jump* trace operators. Let T^+ and T^- be two neighboring elements, and $\mathbf{n}^+$, $\mathbf{n}^-$ be their outward normal unit vectors, respectively ($\mathbf{n}^\pm = \mathbf{n}_{T^\pm}$). Let $\zeta^\pm$ and $\boldsymbol{\tau}^\pm$ be the restriction of ζ and $\boldsymbol{\tau}$ to $T^\pm$. We set:

$$\begin{aligned} 2\{\zeta\} &= (\zeta^+ + \zeta^-), \quad [\![\,\zeta\,]\!] = \zeta^+\mathbf{n}^+ + \zeta^-\mathbf{n}^- && \text{on } E \in \mathcal{E}_h^o, \\ 2\{\boldsymbol{\tau}\} &= (\boldsymbol{\tau}^+ + \boldsymbol{\tau}^-), \quad [\![\,\boldsymbol{\tau}\,]\!] = \boldsymbol{\tau}^+ \cdot \mathbf{n}^+ + \boldsymbol{\tau}^- \cdot \mathbf{n}^- && \text{on } E \in \mathcal{E}_h^o, \end{aligned}$$

and on $E \in \mathcal{E}_h^\partial$ we set $[\![\,\zeta\,]\!] = \zeta\mathbf{n}$ and $\{\boldsymbol{\tau}\} = \boldsymbol{\tau}$. We will also use the notation

$$(u, w)_{\mathcal{T}_h} = \sum_{T\in\mathcal{T}_h} \int_T uw dx \qquad \langle u, w\rangle_{\mathcal{E}_h} = \sum_{E\in\mathcal{E}_h} \int_E uw \quad \forall\, u, w, \in V^{DG}.$$

The approximation to the solution of (1) reads:

$$\text{Find} \quad u \in V^{DG} \quad \text{such that} \quad \mathcal{A}(u, w) = (f, w)_{\mathcal{T}_h}\,, \quad \forall\, w \in V^{DG}. \tag{2}$$

$\mathcal{A}(\cdot,\cdot)$ is the bilinear form of the IIPG method (see [8]):

$$\mathcal{A}(u, w) = (\mathbb{K}\nabla u, \nabla w)_{\mathcal{T}_h} - \langle\{\mathbb{K}\nabla u\}, [\![\, w\,]\!]\rangle_{\mathcal{E}_h} + \langle S_E\{\mathbb{K}[\![\, u\,]\!]\}, [\![\, w\,]\!]\rangle_{\mathcal{E}_h}\,, \tag{3}$$

where $S_E = \alpha_E h_E^{-1}$ with $\alpha_E \geq \alpha^* > 0$ for all $E \in \mathcal{E}_h$ and h_E the length of the edge E in $d = 2$ and the diameter of the face E in $d = 3$. We denote by α^* a *fixed* value of the penalty parameter for which $\mathcal{A}(\cdot,\cdot)$ is coercive. In the numerical experiments we take α^* to be larger than (but close to) the critical value for which $\mathcal{A}(\cdot,\cdot)$ is only semidefinite. We point out that when α^* is close to that critical value one may expect less accurate DG solution due to the fact that the discrete problem is not stable. We present interative methods which are uniformly convergent independently of the value of α^* as long as, α^* is only just large enough to ensure coercivity of the bilinear forms, even though the discretizations in this case lead to "nearly singular" linear systems. This of course includes the cases of stable discretizations resulting in accurate DG solutions, and hence the methods that we propose are aplicable to the cases interesting from practical point of view. In general, α_E might vary from one face to another, but we assume that the possible variations on α are uniformly bounded, namely

$$\frac{\alpha_{\max}}{\alpha_{\min}} \approx 1, \quad \alpha_{\min} := \min_{E\in\mathcal{E}_h} \alpha_E \ , \quad \alpha_{\max} := \max_{E\in\mathcal{E}_h} \alpha_E \ . \tag{4}$$

Together with $\mathcal{A}(\cdot,\cdot)$, we consider also the bilinear form that results by computing all the integrals in (3) with the mid-point quadrature rule:

$$\mathcal{A}_0(u,w) = (\mathbb{K}\nabla u, \nabla w)_{\mathcal{T}_h} - \langle\{\mathbb{K}\nabla u\}, [\![w]\!]\rangle_{\mathcal{E}_h} + \langle S_E \mathcal{P}_E^0(\{\mathbb{K}[\![u]\!]\}), [\![w]\!]\rangle_{\mathcal{E}_h} \ , \tag{5}$$

where $\mathcal{P}_E^0 : L^2(E) \longrightarrow \mathbb{P}^0(E)$ is for each $E \in \mathcal{E}_h$, the L^2-orthogonal projection onto the constants: $\mathcal{P}_E^0(u) := \frac{1}{|E|}\int_E u, \forall\, u \in L^2(E)$. Continuity and coercivity can be shown for (3) in the DG energy norm (see [2]), which is equivalent to the norm induced by the symmetric part of $\mathcal{A}(\cdot,\cdot)$;

$$\|u\|_{\widetilde{\mathcal{A}}} := \widetilde{\mathcal{A}}(u,u) \quad \widetilde{\mathcal{A}}(u,w) = \frac{\mathcal{A}(u,w) + \mathcal{A}(w,u)}{2} \quad \forall\, u, w \in V^{DG} \ .$$

We will also use the equivalent form of the IIPG method obtained via the *weighted residual* approach introduced in [4]:

$$\begin{aligned}
\mathcal{A}(u,w) &= \langle [\![\mathbb{K}\nabla u]\!], \{w\}\rangle_{\mathcal{E}_h^o} + \langle [\![u]\!], S_E\{\mathbb{K}[\![w]\!]\}\rangle_{\mathcal{E}_h} && \forall u, w \in V^{DG}, \\
\mathcal{A}_0(u,w) &= \langle [\![\mathbb{K}\nabla u]\!], \{w\}\rangle_{\mathcal{E}_h^o} + \langle [\![u]\!], S_E\mathcal{P}_E^0(\{\mathbb{K}[\![w]\!]\})\rangle_{\mathcal{E}_h} && \forall u, w \in V^{DG},
\end{aligned}$$

where we have already discarded the term $(-\div(\mathbb{K}\nabla u), w)_{\mathcal{T}_h}$, since $u \in V^{DG}$ is linear and $\mathbb{K}$ is constant on each $T \in \mathcal{T}_h$, and therefore $(-\div(\mathbb{K}\nabla u), w)_{\mathcal{T}_h} = 0$. Next result guarantees the *spectral equivalence* of $\mathcal{A}_0(\cdot,\cdot)$ and $\mathcal{A}(\cdot,\cdot)$.

Lemma 1. *Let $\mathcal{A}(\cdot,\cdot)$ and $\mathcal{A}_0(\cdot,\cdot)$ be the bilinear forms of the IIPG method defined in (3) and (5). Then, there exist $c_2 > 0$ depending only on the shape regularity of $\mathcal{T}_h$ and $c_0 = c_0(\alpha_{\max}, \mathfrak{c}_1) > 0$, such that:*

$$c_2\mathcal{A}_0(u,u) \leq \mathcal{A}(u,u) \leq c_0(\alpha_{\max}, \mathfrak{c}_1)\mathcal{A}_0(u,u) \quad \forall u \in V^{DG}.$$

3 Space Decomposition

In this section we introduce a new basis which provides a natural subspace splitting of the linear V^{DG}-space. We will show that if such basis is used the linear system associated to (2) has special properties that allow for simple derivation of efficient iterative methods for the non-symmetric IIPG method.

Let $\varphi_{E,T}$ denote the canonical Crouzeix-Raviart (CR) basis function on T, dual to the degree of freedom at the mass center m_E of the face E, and extended by zero outside T. Note that $\varphi_{E,T} \in L^2(\Omega)$ and the support of $\varphi_{E,T}$ is T. Then, for any $u \in V^{DG}$:

$$u(x) = \sum_{T\in\mathcal{T}_h} \sum_{E\in\partial T} u_T(m_E)\varphi_{E,T}(x) = \sum_{E\in\mathcal{E}_h} u^+(m_E)\varphi_E^+(x) + \sum_{E\in\mathcal{E}_h^o} u^-(m_E)\varphi_E^-(x).$$

with $\varphi_E^{\pm}(x) := \varphi_{E,T^{\pm}}(x)$. Let V^{CR} be the classical Crouziex-Raviart space:

$$V^{CR} = \left\{ v \in L^2(\Omega) \,:\, v_{|_T} \in \mathbb{P}^1(T)\, \forall T \in \mathcal{T}_h \text{ and } \mathcal{P}_E^0[\![v]\!] = 0 \;\forall\, E \in \mathcal{E}_h^o \right\}.$$

Note that $v = 0$ at the midpoint m_E of each $E \in \mathcal{E}_h^{\partial}$. We also define the space

$$\mathcal{Z} = \left\{ z \in L^2(\Omega) \,:\, z_{|_T} \in \mathbb{P}^1(T)\, \forall T \in \mathcal{T}_h \text{ and } \mathcal{P}_E^0\{v\} = 0 \;\forall\, E \in \mathcal{E}_h^o \right\}. \quad (6)$$

Observe that functions from $\mathcal{Z}$ have nonzero jumps on each internal face and so they can be deemed as highly oscillatory. Defining now:

$$\varphi_E^{CR} = \varphi_{E,T^+} + \varphi_{E,T^-} = 2\{\varphi_{E,T^{\pm}}\} \quad \forall\, E \in \mathcal{E}_h^o \quad E = T^+ \cap T^-,$$

$$\begin{cases} \psi^z_{E,T^{\pm}} = \varphi_{E,T^{\pm}} - \varphi_{E,T^{\mp}} & \forall\, E \in \mathcal{E}_h^o \quad E = T^+ \cap T^-, \\ \psi^z_{E,T} = \varphi_{E,T} & \forall\, E \in \mathcal{E}_h^{\partial}, \quad E = T \cap \partial\Omega, \end{cases}$$

we obtain a decomposition of the functions $\{\varphi_{E,T}\}$ which provides following representation for the spaces V^{CR} and $\mathcal{Z}$:

$$\begin{array}{rcl} V^{CR} & = & \operatorname{span}\{\varphi_E^{CR}\}_{E\in\mathcal{E}_h^o} \\ \mathcal{Z} & = & \operatorname{span}\{\psi^z_{E,T}\}_{E\in\mathcal{E}_h^o} \oplus \operatorname{span}\{\psi^z_{E,T}\}_{E\in\mathcal{E}_h^{\partial}}. \end{array} \quad (7)$$

Next result summarizes these observations.

Proposition 1. *For any $u \in V^{DG}$ there exists a unique $v \in V^{CR}$ and a unique $z \in \mathcal{Z}$ such that $u = v + z$, that is: $V^{DG} = V^{CR} \oplus \mathcal{Z}$.*

Thus, for all $u \in V^{DG}$ we have $u = v + z$ with unique v and z, given by

$$v = \sum_{E\in\mathcal{E}_h^o} \mathcal{P}_E^0(\{u\})\varphi_E^{CR}(x), \qquad z = \sum_{E\in\mathcal{E}_h} \mathcal{P}_E^0\left(\frac{[\![u]\!]\cdot \mathbf{n}^+}{2}\right)\psi^z_{E,T^+}(x).$$

Next Lemma gives a $\mathcal{A}$-"orthogonality" property of the subspace splitting.

Lemma 2. *Let $u \in V^{DG}$ be such that $u = v + z$ with $v \in V^{CR}$ and $z \in \mathcal{Z}$. Let $\mathcal{A}_0(\cdot,\cdot)$ be the bilinear form defined in* (5)*. Then,*

$$\mathcal{A}_0(v, z) = 0 \qquad \forall\, v \in V^{CR}, \quad \forall\, z \in \mathcal{Z}.$$

3.1 Matrix Representation of the DG Bilinear Forms

We denote by A the discrete operator $(Au, w) = \mathcal{A}(u, w)$ (resp. $(A_0 u, w) = \mathcal{A}_0(u, w)$). Let $\mathbb{A}$ (resp. $\mathbb{A}_0$) be the matrix representation of A (resp. A_0) in certain basis. The solution of (2) amounts to the solution of the linear system

$$\mathbb{A}\mathbf{u} = \mathbf{f}, \tag{8}$$

where $\boldsymbol{u}, \boldsymbol{f}$ are the vector representations of the unknown u and the source f. If the basis (7) is used for all these representations, we have:

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{v} \end{bmatrix}, \qquad \mathbb{A}_0 = \begin{bmatrix} \mathbb{A}_0^{zz} & \mathbf{0} \\ \mathbb{A}_0^{vz} & \mathbb{A}_0^{vv} \end{bmatrix}, \qquad \mathbb{A} = \begin{bmatrix} \mathbb{A}^{zz} & \mathbb{A}^{zv} \\ \mathbb{A}^{vz} & \mathbb{A}^{vv} \end{bmatrix}. \tag{9}$$

The blocks $\mathbb{A}^{zz}$, $\mathbb{A}_0^{zz}$ and $\mathbb{A}^{vv}$, $\mathbb{A}_0^{vv}$ correspond, respectively, to the stiffness matrices that result when approximating the solution to (1) with the IIPG method restricted to the $\mathcal{Z}$ and V^{CR} subspaces. The block lower triangular form of $\mathbb{A}_0$ in (9) is a consequence of Lemma 2. The solution of (8) with the block matrix $\mathbb{A}$ (or $\mathbb{A}_0$) as in (9) will certainly involve solutions of smaller systems with $\mathbb{A}^{zz}$ (or $\mathbb{A}_0^{zz}$) and $\mathbb{A}^{vv}$ (or $\mathbb{A}_0^{vv}$). Since, such systems are also solved in every iteration in the method we propose, we next comment on methods for their solution:

Solution in V^{CR}: Restricting the IIPG to the V^{CR} space, we get:

$$\mathcal{A}_0(v, \varphi) = (\mathbb{K}\nabla v, \nabla\varphi)_{\mathcal{T}_h} = \sum_{T\in\mathcal{T}_h} (\mathbb{K}\nabla v, \nabla\varphi)_T \qquad \forall\, v\,, \varphi \in V^{CR}\,,$$
$$\mathcal{A}(v, \varphi) = (\mathbb{K}\nabla v, \nabla\varphi)_{\mathcal{T}_h} + \langle S_E[\![\, v\,]\!], \{\mathbb{K}[\![\,\varphi\,]\!]\}\rangle_{\mathcal{E}_h^o} \qquad \forall v, \varphi \in V^{CR}\,.$$

Hence, both $\mathbb{A}_0^{vv}$ and $\mathbb{A}^{vv}$ are s.p.d. Moreover, note that $\mathcal{A}_0$ is the standard non-conforming CR finite element method for the solution of (1). From the spectral equivalence in Lemma 1, any system with $\mathbb{A}_0^{vv}$ or $\mathbb{A}^{vv}$ can be efficiently solved by using any of the known solvers for the CR approximation of (1); as those proposed in [7] or [9].

Solution in the $\mathcal{Z}$-space: Using the weighted residual formulation together with the definition (6) of the $\mathcal{Z}$ space, it follows that $\forall\ z,\ \psi \in \mathcal{Z}$:

$$\mathcal{A}_0(z, \psi) = \langle S_E \mathcal{P}_E^0([\![\, z\,]\!]), \{\mathbb{K}[\![\,\psi\,]\!]\}\rangle_{\mathcal{E}_h} \quad \mathcal{A}(z, \psi) = \langle S_E[\![\, z\,]\!], \{\mathbb{K}[\![\,\psi\,]\!]\}\rangle_{\mathcal{E}_h}\,. \tag{10}$$

Thus, restricted to $\mathcal{Z}$, both $\mathcal{A}_0$ and $\mathcal{A}$ are symmetric and coercive (since we have set $\alpha_E \geq \alpha^* > 0$ for all $E \in \mathcal{E}_h$). Therefore the blocks $\mathbb{A}_0^{zz}$ and $\mathbb{A}^{zz}$ are both s.p.d. Next Lemma establishes upper and lower bounds on their eigenvalues showing that $\mathbb{A}_0^{zz}$ and $\mathbb{A}_{zz}$ are well conditioned.

Lemma 3. *Let $\mathcal{Z}$ be the space defined in* (6)*. Then for all $z \in \mathcal{Z}$, it holds*

$$c_1(\alpha_{\min})h^{-2}\|\mathbb{K}^{1/2}z\|_{0,\mathcal{T}_h}^2 \leq \mathcal{A}_0(z, z) \leq \mathcal{A}(z, z) \leq c_2(\alpha_{\max})h^{-2}\|\mathbb{K}^{1/2}z\|_{0,\mathcal{T}_h}^2\,,$$

where c_1 and c_2 depend on the mesh regularity and $\alpha_{\min}, \alpha_{\max}$ are as in (4)*.*

The proof of similar result can be found in [3]. By virtue of this lemma and (4), denoting by κ the condition number, we have that $\kappa(\mathbb{A}^{zz}) = O(1)$ and $\kappa_2(\mathbb{A}_0^{zz}) = O(1)$ independent of the mesh size. Clearly, then a linear system with $\mathbb{A}^{zz}$ can be solved using the method of Conjugate Gradients (CG) and the number of CG iterations needed for achieving a prescribed tolerance is also independent of the mesh size. It is also easy to show that the matrix $\mathbb{A}_0^{zz}$ is diagonal, and hence $\mathbb{A}_0^{zz}$ can be also used as a preconditioner for $\mathbb{A}^{zz}$.

4 A Uniformly Convergent Iterative Method

The general setting is a linear iterative algorithm with a given $\mathcal{B}(\cdot,\cdot) \approx \mathcal{A}(\cdot,\cdot)$:

Algorithm 1 *Given initial guess u_0, let u_k, $k \geq 0$ be the current approximation to the solution. The next iterate u_{k+1} is then defined by*

1. *Solve* $\mathcal{B}(e_k, w) = (f, w)_{\mathcal{T}_h} - \mathcal{A}(u_k, w) \quad \forall\, w \in V^{DG}$.
2. *Update* $u_{k+1} = u_k + e_k$.

In [3], the uniform convergence of Algorithm 1 is shown with $\mathcal{B} = \widetilde{\mathcal{A}}$, the symmetric part of $\mathcal{A}$. Here we propose more efficient (in terms of computational work) algorithm. It is suggested by the fact that $\mathbb{A}_0$ is lower triangular and the symmetric parts of $\mathcal{A}(\cdot,\cdot)$ and $\mathcal{A}_0(\cdot,\cdot)$ are spectrally equivalent. This suggest to take the "block lower triangular part" of $\mathcal{A}(\cdot,\cdot)$ as $\mathcal{B}(\cdot,\cdot)$, namely:

$$\mathcal{B}(z + v, \psi^z + \varphi) := \mathcal{A}(z, \psi^z) + \mathcal{A}(z, \varphi) + \mathcal{A}(v, \varphi), \tag{11}$$

$\forall\, v, \varphi \in V^{CR}$ and $\forall\, z, \psi^z \in \mathcal{Z}$. The restrictions of this bilinear form on V^{CR} and $\mathcal{Z}$ are easy to find and the resulting iterative method can be then written in terms of solution of the problems on the subspaces as follows:

Algorithm 2 *Let u_0 be a given initial guess. For $k \geq 0$, and given $u_k = z_k + v_k$, the next iterate $u_{k+1} = z_{k+1} + v_{k+1}$ is defined via the two steps:*

1. *Solve* $\mathcal{A}(z_{k+1}, \psi^z) = (f, \psi^z)_{\mathcal{T}_h} - \mathcal{A}(v_k, \psi^z) \quad \forall\, \psi^z \in \mathcal{Z}$.
2. *Solve* $\mathcal{A}(v_{k+1}, \varphi) = (f, \varphi)_{\mathcal{T}_h} - \mathcal{A}(z_{k+1}, \varphi) \quad \forall\, \varphi \in V^{CR}$.

Observe that algorithm 2 requires two solutions of smaller s.p.d problems: one solution in $\mathcal{Z}$-space (step 1 of the algorithm 2), and one solution in V^{CR}-space (step 2 of algorithm 2). The solution of the subproblems on $\mathcal{Z}$ and on V^{CR} can be done in an efficient way as discussed in the previous section.

5 Numerical Results

We present a set of numerical experiments aimed at assessing the performance of the proposed iterative method. We consider the model problem (1) on the unit square in $\mathbb{R}^2$ triangulated with a family of unstructured triangulations $\mathcal{T}_h$. In the tables given

below $J = 1$ corresponds to the coarsest grid and each refined triangulation on level J, $J = 2, \ldots, 4$ is obtained by subdividing each of the triangles forming the grid on level $(J-1)$ into four congruent triangles. We set the permeability coefficient $\mathbb{K} = \mathbb{I}$, to ease the comparison with the iterative methods for IIPG given in [3], where the symmetric part of $\mathcal{A}$, which we denote by $\widetilde{\mathcal{A}}$, is used as preconditioner. The coarsest grid has $n_1 = 480$ dofs and the finest ($J = 4$) has approximately $n_4 = 30,720$ dofs. We have set $\alpha = K\alpha^*$, and $\alpha^* = 0.9$ for $J = 1$ and $\alpha^* = 1.3$ for $J \geq 2$. We denote by $\mathbb{B}$ the matrix representation of $\mathcal{B}(\cdot,\cdot)$, as given in (11), and by $\widetilde{\mathbb{A}}$ the one of $\widetilde{\mathcal{A}}(\cdot,\cdot)$. The latter being s.p.d. induces a norm $\mathbb{R}^{n_J}$ denoted here by $\|\cdot\|_{\widetilde{\mathbb{A}}}$. The corresponding matrix norm is denoted below by the same symbol.

In Table 1 are given the rates of convergence measured in $\|\cdot\|_{\widetilde{\mathbb{A}}}$-norm and in Table 2 the asymptotic convergence rates (the spectral $\rho(\mathbb{I} - \mathbb{B}^{-1}\mathbb{A})$).

Table 1. Rate of convergence: $\|\mathbb{I} - \mathbb{B}^{-1}\mathbb{A}\|_{\widetilde{\mathbb{A}}}$ for different levels and different values of the penalty parameter $\alpha = K\alpha^*$.

	$J=1$	$J=2$	$J=3$	$J=4$
$K=1$	0.541	0.530	0.571	0.595
$K=2$	0.529	0.566	0.574	0.579
$K=4$	0.576	0.610	0.616	0.619
$K=8$	0.616	0.641	0.645	0.648

Table 2. Asymptotic convergence rate: $\rho(\mathbb{I} - \mathbb{B}^{-1}\mathbb{A})$ for different levels and different values of the penalty parameter $\alpha = K\alpha^*$.

	$J=1$	$J=2$	$J=3$	$J=4$
$K=1$	0.448	0.465	0.469	0.470
$K=2$	0.451	0.465	0.469	0.470
$K=4$	0.454	0.466	0.469	0.470
$K=8$	0.456	0.467	0.470	0.470

The conclusion that we may draw from the above experiments is that the convergence rate in $\|\cdot\|_{\widetilde{\mathbb{A}}}$ norm is uniform with respect to the mesh size, and deteriorates when increasing the value of the penalty parameter. The asymptotic convergence rate is uniformly bounded with respect to both K and the mesh size. The numerical tests clearly show that the $\|\cdot\|_{\widetilde{\mathbb{A}}}$ norm could be used to theoretically analyze the convergence behavior of this iterative method. However, as we have already mentioned, obtaining quantitative theoretical results that reflect and match the convergence rates presented in Table 1 could be quite involved and is subject of a current research.

Acknowledgments First author was supported by MEC grants MTM2008-03541 and HI2008-0173. The work of the second author was supported in part by the National Science Foundation NSF-DMS 0511800 and NSF-DMS 0810982.

References

1. P.F. Antonietti and B. Ayuso. Schwarz domain decomposition preconditioners for discontinuous Galerkin approximations of elliptic problems: non-overlapping case. *Math. Model. Numer. Anal.*, 41(1):21–54, 2007.
2. D.N. Arnold, F. Brezzi, B. Cockburn, and L. Donatella Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, 39(5):1749–1779 (electronic), 2001/02.
3. B. Ayuso de Dios and L. Zikatanov. Uniformly convergent iterative methods for discontinuous Galerkin discretizations. *J. Sci. Comput.*, 40(1–3):4–36, 2009.
4. F. Brezzi, B. Cockburn, L.D. Marini, and E. Süli. Stabilization mechanisms in discontinuous Galerkin finite element methods. *Comput. Methods Appl. Mech. Eng.*, 195(25–28): 3293–3310, 2006.
5. M. Dryja, J. Galvis, and M. Sarkis. BDDC methods for discontinuous Galerkin discretization of elliptic problems. *J. Complex.*, 23(4–6):715–739, 2007.
6. S.C. Eisenstat, H.C. Elman, and M.H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.*, 20(2):345–357, 1983.
7. M. Sarkis. Nonstandard coarse spaces and Schwarz methods for elliptic problems with discontinuous coefficients using non-conforming elements. *Numer. Math.*, 77(3):383–406, 1997.
8. S. Sun and M.F. Wheeler. Symmetric and nonsymmetric discontinuous Galerkin methods for reactive transport in porous media. *SIAM J. Numer. Anal.*, 43(1):195–219 (electronic), 2005.
9. P.S. Vassilevski and J. Wang. An application of the abstract multilevel theory to nonconforming finite element methods. *SIAM J. Numer. Anal.*, 32(1):235–248, 1995.

A Study of Prolongation Operators Between Non-nested Meshes

Thomas Dickopf[1]* and Rolf Krause[2]

[1] Institute for Numerical Simulation, University of Bonn,
53115 Bonn, Germany, dickopf@ins.uni-bonn.de

[2] Institute of Computational Science, University of Lugano,
6904 Lugano, Switzerland, rolf.krause@usi.ch

Summary. For a class of multilevel preconditioners based on non-nested meshes, we study numerically several selected prolongation and restriction operators. Robustness with respect to the mesh size and to jumps in the coefficients is demonstrated.

Key words: multilevel preconditioners, geometric and algebraic multigrid methods, finite elements, non-nested meshes, prolongation

1 Introduction

The topic of this paper is a preconditioning strategy based on non-nested meshes for linear problems arising from finite element discretizations. Using a set of suitable prolongation and restriction operators, we give an explicit construction of a nested space hierarchy with corresponding bases. The analysis of the resulting multilevel preconditioner can be carried out in a natural way by looking at the original non-nested spaces and the connecting operators.

The present approach has the advantage, as compared with (purely) algebraic multigrid methods, that the little geometric information entering the setup leads to a very efficient multilevel hierarchy. Both grid and operator complexity are particularly small. Moreover, in our numerical experiments, we observed robustness of the developed semi-geometric method with respect to jumps in the coefficients; its performance does also not deteriorate for systems of partial differential equations.

Aside from [1, 6, 14], our semi-geometric framework is distinctly motivated by the literature on domain decomposition methods for unstructured meshes, e. g., [4, 5], and the auxiliary space method [15]. We refer to [8] for a more detailed review. To our knowledge, the present paper is the first one to include a numerical comparison of different, to a greater or lesser extent sophisticated candidates for the prolongation between non-nested finite element spaces. We examine operators from or at least motivated by [2, 4, 5, 7, 8, 10, 11].

* Supported by Bonn International Graduate School in Mathematics

2 Multilevel Preconditioners Based on Non-nested Meshes

Let $\Omega \subset \mathbb{R}^d$ be a Lipschitz domain of dimension $d \in \{2,3\}$. For a right hand side $f \in H^{-1}(\Omega)$ and a positive function $\alpha \in L^\infty(\Omega)$ bounded away from zero, we consider the variational model problem

$$u \in H_0^1(\Omega): \quad a(u,v) := (\alpha \nabla u, \nabla v)_{L^2(\Omega)} = f(v), \quad \forall\, v \in H_0^1(\Omega). \tag{1}$$

For a Galerkin discretization of (1), let $(\mathcal{T}_l)_{l\in\mathbb{N}}$ be a family of *non-nested* shape regular meshes of domains $(\Omega_l)_{l\in\mathbb{N}}$. For a fixed finest level $L \geq 2$, assume for simplicity that $\Omega_L = \Omega$ and, in addition, $\Omega_l \supset \Omega$ for all $l \in \{0,\dots,L-1\}$. Let $h_l : \Omega_l \to \mathbb{R}_+$ be a suitable function, e. g., piecewise constant, reflecting the local mesh size of $\mathcal{T}_l$. We denote the set of nodes of $\mathcal{T}_l$ by $\mathcal{N}_l$ and abbreviate $N_l := |\mathcal{N}_l|$. At each level l, we consider the space X_l of Lagrange conforming finite elements of first order with incorporated homogeneous Dirichlet boundary conditions and denote its nodal basis functions as $(\lambda_p^l)_{p\in\mathcal{N}_l}$ with $\lambda_p^l(q) = \delta_{pq}$, $p,q \in \mathcal{N}_l$. Finally, set $\omega_p := \operatorname{supp}(\lambda_p^l)$ for $p \in \mathcal{N}_l$.

Now, the goal is to develop an efficient method for the iterative solution of the ill-conditioned discrete problem

$$u \in X_L: \quad A_L u = f_L, \tag{2}$$

with the stiffness matrix A_L associated with X_L, namely $(A_L)_{pq} := a(\lambda_p^L, \lambda_q^L)$ for $p,q \in \mathcal{N}_L$, and the right hand side f_L given by $(f_L)_p := f(\lambda_p^L)$. Here and in the following, we do not aspire to distinguish strictly between an operator or function and its representation with respect to a finite element basis.

We introduce a rather simple multilevel preconditioner C_L. The delicate point, though, is the construction of an appropriate hierarchy of spaces from the originally unrelated spaces $(X_l)_{l=0,\dots,L}$. For this purpose, let the spaces $(X_l)_{l=0,\dots,L}$ be connected by the prolongation operators $(\mathit{\Pi}_{l-1}^l)_{l=1,\dots,L}$, namely

$$\mathit{\Pi}_{l-1}^l : X_{l-1} \to X_l, \quad \forall\, l \in \{1,\dots,L\}.$$

A closer examination of selected linear operators $(\mathit{\Pi}_{l-1}^l)_{l=1,\dots,L}$ will be the key issue of this paper. We construct a nested sequence of spaces $(V_l)_{l=0,\dots,L}$ via $V_L := X_L$, $V_{L-1} := \mathit{\Pi}_{L-1}^L X_{L-1}$, and further

$$V_l := \mathit{\Pi}_{L-1}^L \cdots \mathit{\Pi}_l^{l+1} X_l, \quad \forall\, l \in \{0,\dots,L-2\}.$$

That way, the images of the operators determine the space hierarchy.

With the nodal bases of the finite element spaces X_{l-1} and X_l a matrix representation of $\mathit{\Pi}_{l-1}^l$ in $\mathbb{R}^{N_l \times N_{l-1}}$ can be computed for $l \in \{1,\dots,L\}$. For convenience, we set $\widetilde{\lambda}_q^L = \lambda_q^L$ for $q \in \mathcal{N}_L$. Then, a basis $(\widetilde{\lambda}_p^l)_{p\in\mathcal{N}_l}$ of V_l for $l \in \{0,\dots,L-1\}$ can recursively be defined by

$$\widetilde{\lambda}_q^l := \sum_{p\in\mathcal{N}_{l+1}} (\mathit{\Pi}_l^{l+1})_{pq} \widetilde{\lambda}_p^{l+1}, \quad \forall\, q \in \mathcal{N}_l.$$

In this manner, basis functions at level $l-1$ are nothing but linear combinations of basis functions at level l induced by the operator Π_{l-1}^{l}; they are piecewise linear with respect to the finest mesh $\mathcal{T}_L$. In particular, one can easily see that the matrix $\Pi_{l-1}^{l} \in \mathbb{R}^{N_l \times N_{l-1}}$ may be regarded as an algebraic representation of the natural embedding of the novel spaces V_{l-1} into V_l.

Then, as customary in a variational approach, the coarse level matrices with respect to the bases $(\widetilde{\lambda}_p^l)_{p\in\mathcal{N}_l}$ are assembled by Galerkin restriction in the following setup phase of the multilevel hierarchy.

Algorithm 1 (Setup semi-geometric multigrid method)
Choose type *of prolongation operator according to Sect. 3.*
setupSGM $(\text{type}, (\mathcal{T}_l)_{l=0,\dots,L})$ {
  for $(l = L, \dots, 1)$ {
    Compute prolongation operator: Π_{l-1}^{l}
    Compute coarse level operator: $A_{l-1} = (\Pi_{l-1}^{l})^T A_l \, \Pi_{l-1}^{l}$
  }
}

If A_L is symmetric positive definite and if $(\Pi_{l-1}^{l})_{l=1,\dots,L}$ have full rank, the respective coarse level operators $(A_l)_{l=0,\dots,L-1}$ are symmetric positive definite, too. In particular, the standard smoothing operators such as ν steps of the (symmetric) Gauß–Seidel or the Jacobi method, denoted by $(S_l^{\nu})_{l=1,\dots,L}$ in the following, are assumed to satisfy a smoothing property in V_l.

Algorithm 2 (Semi-geometric $\mathcal{V}(\nu,\nu)$-cycle)
For (the residual) $r \in V_l$ *compute the value* $C_l r$ *as follows.*
SGM $(l, \Pi_{l-1}^{l}, A_l, S_l^{\nu}, r)$ {
  if $(l = 0)$
    Solve exactly: $C_0 r \leftarrow A_0^{-1} r$
  else {
    Pre-smoothing step: $x \leftarrow S_l^{\nu}(r)$
    Coarse level correction:
      Restriction: $\widetilde{r} \leftarrow (\Pi_{l-1}^{l})^T (r - A_l x)$
      Recursive call: $\widetilde{x} \leftarrow \text{SGM}\ (l-1, \Pi_{l-2}^{l-1}, A_{l-1}, S_{l-1}^{\nu}, \widetilde{r})$
      Prolongation: $x \leftarrow x + \Pi_{l-1}^{l} \widetilde{x}$
    Post-smoothing step: $C_l r \leftarrow x + S_l^{\nu}(r - A_l x)$
  }
  return $C_l r$
}

The condition number of the preconditioned operator, $\kappa(C_L A_L)$, may be analyzed using the well-known result by Bramble, Pasciak, Wang, and Xu [3, Theorem 1] achieved at the beginning of the nineties. However, we emphasize that the relevant estimates, more precisely, the existence of H^1-stable operators $\mathcal{Q}_l^V : V_L \to V_l$, $l \in \{0, \dots, L-1\}$ satisfying suitable L^2-approximation properties, follow from assumptions on the original spaces $(X_l)_{l=0,\dots,L}$ and the prolongation operators $(\Pi_{l-1}^{l})_{l=1,\dots,L}$ rather than on the spaces $(V_l)_{l=0,\dots,L}$. Note that the possible dependence of the results on the number of levels is not ruled out. For the details we refer to [8].

Theorem 1 (Quasi-optimal preconditioning [8, Theorem 3.5]). *Let $\Pi_{l-1}^l : X_{l-1} \to X_l$, $l \in \{1, \dots, L\}$, be H^1-stable prolongation operators with the L^2-approximation properties*

$$\|h_l^{-1}(v - \Pi_{l-1}^l v)\|_{L^2(\Omega)} \lesssim \|v\|_{H^1(\Omega)}, \quad \forall\, v \in X_{l-1}. \tag{3}$$

Assume there are H^1-stable mappings $\mathcal{Q}_l^X : X_L \to X_l$, $l \in \{0, \dots, L-1\}$, satisfying the analogous L^2-approximation properties. If, additionally, the operators $(S_l^\nu)_{l=1,\dots,L}$ have suitable smoothing properties, then the multilevel method C_L defined by Algorithms 1 and 2 is a quasi-optimal preconditioner, i. e., $\kappa(C_L A_L) \lesssim 1$ uniformly with respect to the mesh size.

3 Looking for Suitable Prolongation Operators

The presented preconditioner based on non-nested meshes is related to both agglomeration multigrid methods [6] and aggregation-based algebraic multigrid methods [1, 12, 14]. The difference is that in our semi-geometric approach the coarsening reflected by the "agglomerates" or "aggregates", respectively, and thus the structures of the coarse level operators are in large part determined by the meshes $(\mathcal{T}_l)_{l=0,\dots,L}$. Still, the second ingredient to the setup in Algorithm 1 is a set of prolongation operators $(\Pi_{l-1}^l)_{l=1,\dots,L}$. It turns out that the little geometric information that enters the method is enough to generate an efficient space hierarchy with relatively smooth coarse level functions. Especially, no additional prolongation smoother [14] is needed.

The paradigm one should keep in mind is that, in the multigrid context, the L^2-projection is a natural way to prolongate a coarse level correction to a finer mesh. As the evaluation of the discrete L^2-projection is computationally inefficient in case of non-nested meshes, one has to seek an alternative.

In this section, some selected (intuitive and more elaborate) candidates for the prolongation operators $(\Pi_{l-1}^l)_{l=1,\dots,L}$ are specified. This is done in preparation for the numerical studies in the last section of the paper; for a more detailed review we refer to [8]. First, we consider the most elementary operator. In the literature on domain decomposition methods for unstructured meshes, the *standard finite element interpolation* $\mathcal{I}_{l-1}^l : \mathcal{C}^0(\Omega) \supset X_{l-1} \to X_l$, $u \mapsto \mathcal{I}_{l-1}^l u := \sum_{p \in \mathcal{N}_l} u(p)\lambda_p^l$, has been proposed to be used with non-nested coarse meshes. Different proofs of the H^1-stability and the L^2-approximation property (3) can be found in [4, 5, 13] in the context of partition lemmas.

Whereas the above mapping operates on continuous functions only, the rest of the operators comprise a weighting and are thus well-defined on appropriate Lebesgue spaces. The *Clément quasi-interpolation operator* first introduced in the influential paper [7] is defined by

$$\mathcal{R}_{l-1}^l : L^2(\Omega) \supset X_{l-1} \to X_l, \quad u \mapsto \mathcal{R}_{l-1}^l u := \sum_{p \in \mathcal{N}_l} (Q_p u)(p)\lambda_p^l, \tag{4}$$

with the L^2-projections Q_p onto the local polynomial spaces $\mathbb{P}_r(\omega_p)$ of degree $r \in \mathbb{N}$. It is probably most famous for its frequent use in proofs of the reliability of a posteriori error estimators. In Sect. 4, we employ $\mathcal{R}_{l-1}^l$ with $r = 0$.

The following *L^2-quasi-projection operator* has been analyzed in [2] to construct approximation operators replacing the L^2-projection from the space $H^1(\Omega)$ to the discrete spaces X_l. It is defined by

$$\widehat{\mathcal{Q}}_{l-1}^l : L^2(\Omega) \supset X_{l-1} \to X_l, \quad u \mapsto \widehat{\mathcal{Q}}_{l-1}^l u := \sum_{p \in \mathcal{N}_l} \frac{(\lambda_p^l, u)_{L^2(\Omega)}}{\int_{\omega_p} \lambda_p^l} \lambda_p^l. \tag{5}$$

Note that this is the operator obtained from the discrete L^2-projection by lumping the mass matrix associated with X_l.

In [8], we have investigated a new operator, primarily motivated by [10, 11]. For its definition, choose a set of functions $(\psi_p^l)_{p \in \mathcal{N}_l}$ with $\psi_p^l \in \mathcal{C}^0(\omega_p)$ for all $p \in \mathcal{N}_l$ such that $(\psi_p^l, \lambda_q^l)_{L^2(\Omega)} = \delta_{pq} \int_{\omega_p} \lambda_p^l$, $\forall\, p, q \in \mathcal{N}_l$. Then, the *pseudo-$L^2$-projection operator* $\mathcal{P}_{l-1}^l : L^2(\Omega) \supset X_{l-1} \to X_l$ is defined by a Petrov–Galerkin variational formulation with trial space X_l and test space $Y_l := \operatorname{span}\{\psi_p^l |\ p \in \mathcal{N}_l\} \not\subset \mathcal{C}^0(\Omega)$ yielding the representation formula

$$\mathcal{P}_{l-1}^l u = \sum_{p \in \mathcal{N}_l} \frac{(\psi_p, u)_{L^2(\Omega)}}{\int_{\omega_p} \lambda_p^l} \lambda_p^l. \tag{6}$$

For this last operator, the authors have proved the H^1-stability and the L^2-approximation property in case of shape regular meshes. Therefore, the multilevel preconditioner C_L defined by Algorithms 1 and 2 is quasi-optimal with the choice $\Pi_{l-1}^l = \mathcal{P}_{l-1}^l$ for $l \in \{1, \ldots, L\}$; see [8, Theorem 5.7]. Note that the present considerations do not yield estimates which are independent of the number of levels L, though.

Let us remark that, if the meshes $\mathcal{T}_{l-1}$ and $\mathcal{T}_l$ are nested, the operators $\mathcal{R}_{l-1}^l$ and $\widehat{\mathcal{Q}}_{l-1}^l$ do not coincide with the L^2-projection, which is the same as the finite element interpolation in this case; see [8]. In contrast, the operator $\mathcal{P}_{l-1}^l$ is always a projection, especially the L^2-projection in the nested case.

To evaluate (4), (5) or (6) in the setup phase (Algorithm 1) exactly, one has to compute the intersections of elements in consecutive meshes. In practice, good results may be obtained by an approximate numerical integration via a quadrature rule solely based on the finer mesh.

Without loss of generality, we may assume that the prolongation operators do not contain any zero columns; otherwise the respective coarse degrees of freedom are not coupled to the original problem (2) and should be removed in Algorithm 1. As a measure for the efficiency of the multilevel hierarchy itself, in addition to iteration counts or convergence rates, we put forward the notions of grid complexity $\mathcal{C}_{\text{gr}}$ and operator complexity $\mathcal{C}_{\text{op}}$ defined by

$$\mathcal{C}_{\text{gr}} := \sum_{l=0}^{L} N_l / N_L, \quad \mathcal{C}_{\text{op}} := \sum_{l=0}^{L} n_l / n_L, \tag{7}$$

that are common in the AMG literature. Here, n_l is the number of non-zero entries in A_l, $l \in \{0, \dots, L\}$. A prevalent technique to keep $\mathcal{C}_{\text{gr}}$ and $\mathcal{C}_{\text{op}}$ small (and the application of Algorithm 2 efficient) is truncation of the prolongation operators by deleting the entries that are smaller than $\varepsilon_{\text{tr}} = 0.2$ times the maximal entry in the respective row and rescaling afterwards; see [12].

4 Numerical Results

Because of the geometric nature of the coarsening procedure, it is important to analyze its dependence on the meshes. This is done, in each single study, by choosing an independently generated fine mesh $\mathcal{T}_L$ (of varying mesh size) approximating the unit ball. In the fashion of an auxiliary space method [15], we use nested coarse meshes $(\mathcal{T}_l)_{l<L}$ associated with the unit cube (structured; 189, 1,241 and 9,009 nodes, respectively) and standard interpolation between the levels $l < L$. Note that the different fine meshes yield different coarse spaces $(V_l)_{l<L}$ although the respective coarse meshes are unchanged.

We report on the convergence of the conjugate gradient method (until the residual norm is below 10^{-16}) preconditioned by the semi-geometric $\mathcal{V}(3,3)$-cycle (Algorithm 2) with symmetric Gauß–Seidel smoothing. In view of the affinity of our method to aggregation-based AMG, it is reasonable to examine whether an over-relaxation of the coarse level correction can improve the convergence; see [1, 12]. For the model problem (1) and $d = 3$, we identify the scaling factors 1.0, 1.1, 1.1, 1.2 for the four operators, respectively, in Fig. 1. But note that over-correction is not really necessary. The results of our experiments can be found in Table 1 for a constant coefficient $\alpha = 1$ and in Table 2 for a coefficient function α constant on each element in $\mathcal{T}_L$ with a randomly chosen value 1 or 10,000. For each operator, we give the number of iteration steps and an approximation of the asymptotic convergence rate. The last column (with caption "none") always contains the values for the one-level pcg with the symmetric Gauß–Seidel method as preconditioner.

A semi-geometric approach has the best chance of generating a very efficient multilevel hierarchy. This can be verified by noting that the complexities (7) are quite small in all numerical studies, namely down to $\mathcal{C}_{\text{gr}} = 1.035$ and $\mathcal{C}_{\text{op}} = 1.070$, at the same time with convergence rates of 0.0713 and 0.0747, respectively, for the scalar problem with 204,675 nodes; see Table 3.

Finally, let us remark that the convergence behavior does not deteriorate for systems of PDEs; see Table 4. This is in contrast to most algebraic multigrid methods; we refer to [9] for a discussion and an exemplary comparison of different algorithms. Certainly, one reason for this robustness is the fact that we treat the different physical unknowns separately, e. g., the (scalar) displacement in direction of a chosen basis of $\mathbb{R}^d$ in case of (linear) elasticity problems. Thus, the coarse level hierarchy is the same in each component. In the present linear elastic example, we observe a superior performance of the projections $\mathcal{I}$ and $\mathcal{P}$ over the operators $\mathcal{R}$ and $\widehat{\mathcal{Q}}$.

Table 1. Convergence of the pcg for a constant coefficient α. In this and the other convergence studies (see Tables 2 and 4), we give the number of needed pcg iterations and an approximate asymptotic convergence rate for $\Pi = \mathcal{I}, \mathcal{R}, \widehat{\mathcal{Q}}, \mathcal{P}$. Both quantities appear to be reasonably bounded.

#dof	$\mathcal{I}$		$\mathcal{R}$		$\widehat{\mathcal{Q}}$		$\mathcal{P}$		None	
47,348	11	0.0287	11	0.0243	11	0.0230	12	0.0394	91	0.5832
53,460	12	0.0398	11	0.0255	12	0.0345	12	0.0330	97	0.6418
64,833	12	0.0381	12	0.0362	12	0.0288	12	0.0279	102	0.6434
72,525	13	0.0486	12	0.0333	12	0.0311	12	0.0288	106	0.6576
87,244	13	0.0503	13	0.0437	13	0.0452	12	0.0313	114	0.6954
127,787	14	0.0572	15	0.0469	13	0.0421	14	0.0416	125	0.6792
204,675	16	0.0761	16	0.0684	16	0.0737	16	0.0713	146	0.7358

Table 2. Convergence of the pcg for α randomly jumping between 1 and 10,000.

#dof	$\mathcal{I}$		$\mathcal{R}$		$\widehat{\mathcal{Q}}$		$\mathcal{P}$		None	
47,348	15	0.0405	14	0.0235	14	0.0270	14	0.0296	98	0.6222
53,460	15	0.0387	14	0.0286	14	0.0277	14	0.0305	103	0.6767
64,833	15	0.0393	15	0.0357	15	0.0388	15	0.0408	109	0.6810
72,525	17	0.0653	15	0.0399	15	0.0342	15	0.0374	110	0.6470
87,244	16	0.0454	16	0.0439	16	0.0408	16	0.0412	121	0.7225
127,787	18	0.0586	17	0.0524	17	0.0553	17	0.0504	130	0.6880
204,675	20	0.0831	20	0.0790	20	0.0802	20	0.0747	152	0.7206

Table 3. Grid and operator complexity depend on the prolongation type and the problem size. As we keep the coarse meshes fixed (but not the coarse spaces) throughout the presented studies, both $\mathcal{C}_{\text{gr}}$ and $\mathcal{C}_{\text{op}}$ decrease with increasing problem size.

#elem.	#dof	$\mathcal{C}_{\text{gr}}(\mathcal{I})$	$\mathcal{C}_{\text{op}}(\mathcal{I})$	$\mathcal{C}_{\text{gr}}(\mathcal{R})$	$\mathcal{C}_{\text{op}}(\mathcal{R})$	$\mathcal{C}_{\text{gr}}(\widehat{\mathcal{Q}})$	$\mathcal{C}_{\text{op}}(\widehat{\mathcal{Q}})$	$\mathcal{C}_{\text{gr}}(\mathcal{P})$	$\mathcal{C}_{\text{op}}(\mathcal{P})$
262,365	47,348	1.144	1.364	1.144	1.490	1.144	1.430	1.143	1.338
297,620	53,460	1.128	1.320	1.128	1.418	1.128	1.373	1.127	1.296
361,907	64,833	1.106	1.263	1.106	1.330	1.106	1.298	1.105	1.242
405,256	72,525	1.095	1.233	1.095	1.291	1.095	1.263	1.095	1.214
490,617	87,244	1.079	1.190	1.079	1.230	1.079	1.210	1.079	1.173
719,951	127,787	1.055	1.128	1.055	1.149	1.055	1.138	1.055	1.117
1,161,926	204,675	1.035	1.076	1.033	1.085	1.033	1.080	1.035	1.070

Table 4. Linear elastic problem on the unit ball with Poisson ratio 0.3. Some differences in the performance of the prolongation operators may be observed.

#dof	$\mathcal{I}$		$\mathcal{R}$		$\widehat{\mathcal{Q}}$		$\mathcal{P}$		None	
142,044	17	0.0915	24	0.1969	23	0.1754	19	0.1194	137	0.7398
160,380	17	0.0974	25	0.1936	23	0.1667	19	0.1319	146	0.7936
194,499	18	0.1068	28	0.2215	26	0.1767	20	0.1473	152	0.7501
217,575	18	0.1101	29	0.2297	26	0.2107	21	0.1447	160	0.8159
261,732	19	0.1285	31	0.2382	27	0.2095	21	0.1600	175	0.8334
383,361	20	0.1439	32	0.2547	28	0.2279	22	0.1692	186	0.8395
614,025	24	0.1895	34	0.2411	31	0.2267	24	0.1969	217	0.8760

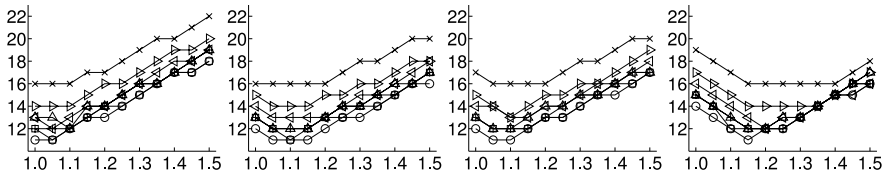


Fig. 1. Over-relaxation of the coarse level correction $\Pi_{l-1}^{l}\widetilde{x}$ in Algorithm 2. Study of the number of pcg iterations depending on the scaling factor for the choices $\Pi = \mathcal{I}$, $\mathcal{R}$, $\widehat{\mathcal{Q}}$, $\mathcal{P}$ (from *left to right*). Each *line* represents a different problem size.

References

1. R. Blaheta. Algebraic multilevel methods with aggregations: an overview. In I. Lirkov, S. Margenov, and J. Waśniewski, editors, *Large-Scale Scientific Computing*, volume 3743 of *Lecture Notes in Computer Science*, pp. 3–14. Springer, 2006.
2. J.H. Bramble, J.E. Pasciak, and P.S. Vassilevski. Computational scales of Sobolev norms with applications to preconditioning. *Math. Comput.*, 69(230):463–480, 2000.
3. J.H. Bramble, J.E. Pasciak, J. Wang, and J. Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Math. Comput.*, 57(195):23–45, 1991.
4. X. Cai. The use of pointwise interpolation in domain decomposition methods with non-nested meshes. *SIAM J. Sci. Comput.*, 16(1):250–256, 1995.
5. T. Chan, B. Smith, and J. Zou. Overlapping Schwarz methods on unstructured meshes using non-matching coarse grids. *Numer. Math.*, 73(2):149–167, 1996.
6. T. Chan, J. Xu, and L. Zikatanov. An agglomeration multigrid method for unstructured grids. In J. Mandel, C. Farhat, and X.-C. Cai, editors, *Domain Decomposition Methods 10*, volume 218 of *Contemporary Mathematics*, pp. 67–81. AMS, Providence, RI, 1998.
7. P. Clément. Approximation by finite element functions using local regularization. *RAIRO Anal. Numér.*, 9(R-2):77–84, 1975.
8. T. Dickopf and R. Krause. A pseudo-L^2-projection for multilevel methods based on non-nested meshes. INS Preprint No. 0908 (University of Bonn) and ICS Preprint No. 2009-04 (University of Lugano), 2009.
9. M. Griebel, D. Oeltz, and M.A. Schweitzer. An algebraic multigrid method for linear elasticity. *SIAM J. Sci. Comput.*, 25(2):385–407, 2003.
10. L.R. Scott and S. Zhang. Finite element interpolation of nonsmooth functions satisfying boundary conditions. *Math. Comput.*, 54(190):483–493, 1990.
11. O. Steinbach. On a generalized L^2-projection and some related stability estimates in Sobolev spaces. *Numer. Math.*, 90(4):775–786, 2002.
12. K. Stüben. An introduction to algebraic multigrid. In U. Trottenberg et al., editors, *Multigrid*, pp. 413–532. Academic Press, London, 2001.
13. A. Toselli and O. Widlund. *Domain Decomposition Methods – Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, Berlin 2005.
14. P. Vaněk, M. Brezina, and J. Mandel. Convergence of algebraic multigrid based on smoothed aggregation. *Numer. Math.*, 88(3):559–579, 2001.
15. J. Xu. The auxiliary space method and optimal multigrid preconditioning techniques for unstructured grids. *Computing*, 56(3):215–235, 1996.

A Parallel Schwarz Method for Multiple Scattering Problems

Daisuke Koyama

The University of Electro-Communications, Chofu, Japan, koyama@im.uec.ac.jp

1 Introduction

Multiple scattering of waves is one of the important research topics in scientific and industrial fields. A number of numerical methods have been developed to compute waves scattered by several obstacles, e.g., acoustic waves scattered by schools of fish, water waves by ocean structures, and elastic waves by particles in composite materials.

In this paper, we focus on the acoustic scattering, which is described as solutions of boundary value problems of the Helmholtz equation in an unbounded domain. In order to compute such solutions numerically, the following method is well known [2, 6]: one introduces an artificial boundary and imposes an artificial boundary condition on it to reduce the original problem on the unbounded domain to a problem on a bounded domain enclosed by the artificial boundary. Recently, [4] have developed a new method by extending the method above to the multiple scattering problem. In their method, one introduces several disjoint artificial boundaries, each of which surrounds one of the obstacles, and imposes an exact non-reflecting boundary condition on such artificial boundaries. This boundary condition is called the multiple DtN (Dirichlet-to-Neumann) boundary condition.

In this paper, we parallelize their method by a parallel nonoverlapping Schwarz method due to [5]. The original unbounded domain is then decomposed into bounded subdomains, each of which is surrounded by one of the artificial boundaries, and the remaining unbounded subdomain. A particular feature of this method is including a problem in the unbounded subdomain, imposing Sommerfeld's radiation condition. This problem is reduced to a certain problem on the multiple artificial boundaries by the natural boundary reduction due to [2].

The remainder of this paper is organized as follows. In Sect. 2 we introduce the exterior Helmholtz problem, and present a parallel Schwarz algorithm and its convergence theorem, which is proved by the energy method due to [1] in Sect. 5. We introduce the multiple DtN operator associated with the problem on the unbounded subdomain in Sect. 3. We describe how to reduce the problem on the unbounded subdomain to the problem on the multiple artificial boundaries in Sect. 4.

2 Exterior Helmholtz Problem and Schwarz Method

We consider the following exterior Helmholtz problem:

$$\begin{cases} -\Delta u - k^2 u = f & \text{in } \Omega_\infty, \\ u = 0 & \text{on } \bigcup_{j=1}^J \partial\mathcal{O}_j, \\ \lim\limits_{r \longrightarrow +\infty} r^{\frac{1}{2}} \left(\dfrac{\partial u}{\partial r} - \mathrm{i}ku \right) = 0, \end{cases} \tag{1}$$

where k is a positive constant, $\mathcal{O}_j$ $(1 \leq j \leq J)$ are bounded open sets of $\mathbb{R}^2$, $\Omega_\infty := \mathbb{R}^2 \setminus \left(\bigcup_{j=1}^J \overline{\mathcal{O}_j} \right)$, and f is a given datum. Assume that Ω_∞ is connected, f has a compact support, and $\partial\mathcal{O}_j$ $(1 \leq j \leq J)$ are of class C^∞. Problem (1) has a unique solution belonging to $H^2_{\mathrm{loc}}(\overline{\Omega_\infty})$ for every compactly supported $f \in L^2(\Omega_\infty)$ (see [7]), where

$$H^m_{\mathrm{loc}}(\overline{\Omega_\infty}) := \{ u \mid u \in H^m(B) \ \text{ for all bounded open set } B \subset \Omega_\infty \} \quad (m \in \mathbb{N}).$$

2.1 Domain Decomposition

Suppose that for each $1 \leq j \leq J$, there exists a ball B_j with radius a_j and center c_j such that $\overline{\mathcal{O}_j} \subset B_j$, $\operatorname{supp} f \subset \bigcup_{j=1}^J B_j$, and $\overline{B_j} \cap \overline{B_l} = \emptyset$ if $j \neq l$. We introduce, for every $1 \leq j \leq J$, artificial boundaries: $\Gamma_j := \left\{ x \in \mathbb{R}^2 \mid |x - c_j| = a_j \right\}$ and bounded domains: $\Omega_j := B_j \setminus \overline{\mathcal{O}_j}$. We further introduce the following unbounded domain: $\Omega_0 := \mathbb{R}^2 \setminus \left(\bigcup_{j=1}^J \overline{B_j} \right)$. Then we can decompose Ω_∞ into subdomains Ω_0, $\Omega_1, \ldots, \Omega_J$: $\overline{\Omega_\infty} = \bigcup_{j=0}^J \overline{\Omega_j}$, and we have $\Omega_j \cap \Omega_l = \emptyset$ if $j \neq l$.

2.2 A Parallel Schwarz Method

To solve problem (1), we consider the following parallel Schwarz method of Lions:

(1) Choose u_0^0 and u_j^0 $(1 \leq j \leq J)$.
(2) For $n = 1, 2, \ldots$, solve

$$\begin{cases} -\Delta u_0^n - k^2 u_0^n = 0 & \text{in } \Omega_0, \\ -\dfrac{\partial u_0^n}{\partial r_j} - \mathrm{i}ku_0^n = -\dfrac{\partial u_j^{n-1}}{\partial r_j} - \mathrm{i}ku_j^{n-1} & \text{on } \Gamma_j \quad (1 \leq j \leq J), \\ \lim\limits_{r \longrightarrow +\infty} r^{1/2} \left(\dfrac{\partial u_0^n}{\partial r} - \mathrm{i}ku_0^n \right) = 0, \end{cases} \tag{2}$$

and for $1 \leq j \leq J$,

$$\begin{cases} -\Delta u_j^n - k^2 u_j^n = f & \text{in } \Omega_j, \\ u_j^n = 0 & \text{on } \partial\mathcal{O}_j, \\ \dfrac{\partial u_j^n}{\partial r_j} - \mathrm{i}ku_j^n = \dfrac{\partial u_0^{n-1}}{\partial r_j} - \mathrm{i}ku_0^{n-1} & \text{on } \Gamma_j. \end{cases} \tag{3}$$

Here, for each $1 \le j \le J$, (r_j, θ_j) are polar coordinates with origin c_j, and then the normal derivative on Γ_j is expressed by $\partial/\partial r_j$. As is well known, problems (2) and (3), respectively, have a unique solution.

Theorem 1. *Let u, u_0^n, and u_j^n $(1 \le j \le J)$ be the solutions of problems (1), (2), and (3), respectively. If*

$$\left.\frac{\partial u_0^0}{\partial r_j} - \mathrm{i}ku_0^0\right|_{\Gamma_j} \quad \text{and} \quad \left.-\frac{\partial u_j^0}{\partial r_j} - \mathrm{i}ku_j^0\right|_{\Gamma_j} \in H^{1/2}(\Gamma_j) \quad (1 \le j \le J), \tag{4}$$

then we have $u_0^n \longrightarrow u$ in $L^2(\Gamma)$ and $u_j^n \longrightarrow u$ in $H^1(\Omega_j)$ $(1 \le j \le J)$ as $n \longrightarrow +\infty$, where $\Gamma := \bigcup_{j=1}^J \Gamma_j$.

3 Multiple DtN Operator

We introduce the multiple DtN operator

$$S:\ H^{1/2}(\Gamma) \left(\cong \prod_{j=1}^J H^{1/2}(\Gamma_j)\right) \longrightarrow H^{-1/2}(\Gamma) \left(\cong \prod_{j=1}^J H^{-1/2}(\Gamma_j)\right)$$

defined by

$$S:\ p = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_J \end{bmatrix} \longrightarrow \begin{bmatrix} -\frac{\partial u}{\partial r_1}\Big|_{\Gamma_1} \\ -\frac{\partial u}{\partial r_2}\Big|_{\Gamma_2} \\ \vdots \\ -\frac{\partial u}{\partial r_J}\Big|_{\Gamma_J} \end{bmatrix},$$

where $p_j := p|_{\Gamma_j}$ $(1 \le j \le J)$, and u is the solution of the following problem:

$$\begin{cases} -\Delta u - k^2 u = 0 & \text{in } \Omega_0, \\ u = p & \text{on } \Gamma, \\ \lim\limits_{r \longrightarrow +\infty} r^{1/2}\left(\dfrac{\partial u}{\partial r} - \mathrm{i}ku\right) = 0. \end{cases}$$

To represent S in an implicit form involving some operators which can be analytically represented, we introduce, for each $1 \le j \le J$, an operator

$$\mathcal{G}_j:\ H^{1/2}(\Gamma_j) \longrightarrow H^1_{\mathrm{loc}}(\overline{D_j})$$

defined by $\mathcal{G}_j[\varphi_j] := w_j$, where $w_j \in H^1_{\mathrm{loc}}(\overline{D_j})$ is the solution of the following problem:

$$\begin{cases} -\Delta w_j - k^2 w_j = 0 & \text{in } D_j, \\ w_j = \varphi_j & \text{on } \Gamma_j, \\ \lim\limits_{r_j \longrightarrow +\infty} r_j^{\frac{1}{2}}\left(\dfrac{\partial w_j}{\partial r_j} - \mathrm{i}kw_j\right) = 0, \end{cases} \tag{5}$$

where $D_j := \{x \in \mathbb{R}^2 \mid |x - c_j| > a_j\}$.

Theorem 2. *If $u \in H^1_{\mathrm{loc}}(\overline{\Omega_0})$ satisfies*

$$\begin{cases} -\Delta u - k^2 u = 0 \quad \text{in } \Omega_0, \\ \lim\limits_{r \longrightarrow +\infty} r^{\frac{1}{2}} \left(\dfrac{\partial u}{\partial r} - \mathrm{i}ku \right) = 0, \end{cases}$$

then there uniquely exists a $\varphi \in H^{1/2}(\Gamma)$ such that

$$u = \sum_{j=1}^{J} \mathcal{G}_j[\varphi_j] \quad \text{in } \Omega_0. \tag{6}$$

Proof. See [4]. □

Taking the traces and the normal derivatives $\partial/\partial r_j$ of both sides of (6) on Γ_j $(1 \le j \le J)$, we get

$$u = \varphi_j + \sum_{l \neq j} \mathcal{P}_{jl}[\varphi_l] \quad \text{on } \Gamma_j \tag{7}$$

and

$$\frac{\partial u}{\partial r_j} = -\mathcal{S}_j[\varphi_j] - \sum_{l \neq j} \mathcal{T}_{jl}[\varphi_l] \quad \text{on} \quad \Gamma_j, \tag{8}$$

respectively, where

$$\mathcal{P}_{jl}[\varphi_l] := \mathcal{G}_l[\varphi_l]|_{\Gamma_j}, \quad -\mathcal{T}_{jl}[\varphi_l] := \frac{\partial}{\partial r_j} \mathcal{G}_l[\varphi_l] \bigg|_{\Gamma_j} \quad (1 \le j \ne l \le J),$$

$$-\mathcal{S}_j[\varphi_j] := \frac{\partial}{\partial r_j} \mathcal{G}_j[\varphi_j] \bigg|_{\Gamma_j} \quad (1 \le j \le J).$$

We here remark that $\mathcal{S}_j$ is the single DtN operator associated with (5). Deleting φ_j from (7) and (8), we can see that the multiple DtN operator S can be represented as follows:

$$S = TC^{-1}, \tag{9}$$

where

$$C := \begin{bmatrix} I & \mathcal{P}_{12} & \cdots & \mathcal{P}_{1J} \\ \mathcal{P}_{21} & I & \cdots & \mathcal{P}_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{P}_{J1} & \mathcal{P}_{J2} & \cdots & I \end{bmatrix}, \qquad T := \begin{bmatrix} \mathcal{S}_1 & \mathcal{T}_{12} & \cdots & \mathcal{T}_{1J} \\ \mathcal{T}_{21} & \mathcal{S}_2 & \cdots & \mathcal{T}_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{T}_{J1} & \mathcal{T}_{J2} & \cdots & \mathcal{S}_J \end{bmatrix}.$$

As we all know, we can obtain an analytical representation of $\mathcal{G}_j$ by separation of variables, and hence, from this representation, we can derive analytical representations of the operators $\mathcal{S}_j$, $\mathcal{T}_{jl}$, and $\mathcal{P}_{jl}$.

Note that we have $C \in \mathrm{Isom}(H^{1/2}(\Gamma),\ H^{1/2}(\Gamma))$, the proof of which will appear elsewhere.

4 How to Solve Problem (2)

If u_0^n is the solution of problem (2), then we have

$$-\frac{\partial u_0^n}{\partial r_j} = Su_0^n|_{\Gamma_j} \quad (1 \le j \le J).$$

Hence we can reduce problem (2) to the following problem on Γ: find $p \in H^{1/2}(\Gamma)$ such that

$$Sp - \mathrm{i}kp = \lambda, \tag{10}$$

where $\lambda \in H^{-1/2}(\Gamma)$ and $\lambda = -\partial u_j^{n-1}/\partial r_j - \mathrm{i}ku_j^{n-1}$ on Γ_j. This process is often called the *natural boundary reduction* (cf. [2]). The solution p of this problem gives the trace on Γ of the solution u_0^n of problem (2).

By (9), we have

$$(S - \mathrm{i}kI)p = \lambda \Longleftrightarrow (T - \mathrm{i}kC)C^{-1}p = \lambda.$$

Hence, we can solve (10) by executing the following two processes:

- Solve $(T - \mathrm{i}kC)\varphi = \lambda$.
- Compute $p = C\varphi$.

Equation $(T - \mathrm{i}kC)\varphi = \lambda$ can be written in the following variational form: find $\varphi \in H^{1/2}(\Gamma)$ such that

$$\begin{aligned} &\langle \mathcal{S}_j\varphi_j, \psi_j\rangle_j + \sum_{l\neq j}\int_{\Gamma_j} \mathcal{T}_{jl}\,[\varphi_l]\,\overline{\psi_j}\,d\Gamma_j \\ &\quad -\mathrm{i}k\left(\int_{\Gamma_j}\varphi_j\overline{\psi_j}\,d\Gamma_j + \sum_{l\neq j}\int_{\Gamma_j}\mathcal{P}_{jl}\,[\varphi_l]\,\overline{\psi_j}\,d\Gamma_j\right) \\ &= \langle \lambda_j, \psi_j\rangle_j \quad \forall\psi_j \in H^{1/2}(\Gamma_j), \quad 1 \le \forall j \le J, \end{aligned} \tag{11}$$

where $\langle\cdot,\cdot\rangle_j$ is the duality pairing between $H^{-1/2}(\Gamma_j)$ and $H^{1/2}(\Gamma_j)$.

We can practically compute (11) by using the analytical representations of the operators $\mathcal{S}_j$, $\mathcal{T}_{jl}$, and $\mathcal{P}_{jl}$. Discretizing (11) by a FEM, we need to solve a linear system whose matrix is full and of order the number of nodes on Γ.

5 Proof of Theorem 1

We can prove Theorem 1 by the energy technique due to [1]. Let u, u_0^n, and u_j^n $(1 \le j \le J)$ be the solutions of problems (1), (2), and (3), respectively. Put $e_j^n := u - u_j^n$ $(0 \le j \le J)$. We should keep in mind that if (4) is satisfied, then $e_0^n \in H^2_{\mathrm{loc}}(\overline{\Omega_0})$ and $e_j^n \in H^2(\Omega_j)$ $(1 \le j \le J)$ for all $n \in \mathbb{N}$.

We now define a *pseudo energy* E^n by

$$E^n := \int_\Gamma |Se_0^n - \mathrm{i}ke_0^n|^2\,d\Gamma + \sum_{j=1}^J \int_{\Gamma_j}\left|\frac{\partial e_j^n}{\partial r_j} - \mathrm{i}ke_j^n\right|^2 d\Gamma_j.$$

Lemma 1. *$\{E^n\}_{n=1}^{\infty}$ is a decreasing sequence.*

Proof. Because e_j^n satisfies the homogeneous equation in Ω_j and the homogeneous boundary condition on $\partial\mathcal{O}_j$, we have

$$\mathrm{Im}\left\{\int_{\Gamma_j}\frac{\partial e_j^n}{\partial r_j}\overline{e_j^n}\,d\Gamma_j\right\}=0 \tag{12}$$

for every $n\in\mathbb{N}$ and for each $1\le j\le J$. Hence, for every $n\in\mathbb{N}$, we have the following *energy equality*:

$$E^{n+1}=E^n+4k\,\mathrm{Im}\left\{\int_{\Gamma}Se_0^n\overline{e_0^n}\,d\Gamma\right\}. \tag{13}$$

Since e_0^n also satisfies the homogeneous equation in Ω_0, we have

$$\mathrm{Im}\left\{\int_{\Gamma}Se_0^n\overline{e_0^n}\,d\Gamma\right\}=-kR\sum_{\mu=-\infty}^{\infty}\mathrm{Im}\left\{\frac{H_\mu^{(1)'}(kR)}{H_\mu^{(1)}(kR)}\right\}\left|e_{0,\mu}^n(R)\right|^2 \tag{14}$$

for an arbitrary positive number R satisfying $\Gamma\subset\{x\in\mathbb{R}^2\,|\,|x|<R\}$, where $H_\mu^{(1)}$ is the first kind Hankel function of order μ, and $e_{0,\mu}^n(R)$ is the μth Fourier coefficient of e_0^n on $\Gamma^R:=\{x\in\mathbb{R}^2\,|\,|x|=R\}$. As we all know, we have

$$\mathrm{Im}\left\{\frac{H_\mu^{(1)'}(kR)}{H_\mu^{(1)}(kR)}\right\}>0 \tag{15}$$

for all $\mu\in\mathbb{Z}$. Therefore, combining (13), (14) and (15) completes the proof of Lemma 1. □

Proposition 1. *We have $S-\mathrm{i}kI\in\mathrm{Isom}(H^{1/2}(\Gamma),\,H^{-1/2}(\Gamma))$.*

Proof. We can prove from the two facts that $S-\mathrm{i}kI$ is a bounded linear operator from $H^{1/2}(\Gamma)$ onto $H^{-1/2}(\Gamma)$, and that the following problem:

$$\begin{cases}-\Delta u-k^2u=0 & \text{in }\Omega_0,\\ -\dfrac{\partial u}{\partial r_j}-\mathrm{i}ku=\lambda & \text{on }\Gamma_j\quad(1\le j\le J),\\ \lim\limits_{r\longrightarrow+\infty}r^{1/2}\left(\dfrac{\partial u}{\partial r}-\mathrm{i}ku\right)=0\end{cases}$$

has a unique solution belonging to $H^1_{\mathrm{loc}}(\overline{\Omega_0})$ for every $\lambda\in H^{-1/2}(\Gamma)$. □

Proof of Theorem 1

Proposition 1 assures us that there exists a positive constant C such that

$$\|e_0^{n+1}\|_{H^{1/2}(\Gamma)}\le C\|\lambda\|_{H^{-1/2}(\Gamma)}, \tag{16}$$

where $\lambda_j=-\partial e_j^n/\partial r_j-\mathrm{i}ke_j^n$. Using (12) and Lemma 1, we have

$$\|\lambda\|^2_{L^2(\Gamma)} = \sum_{j=1}^{J} \int_{\Gamma_j} \left| \frac{\partial e_j^n}{\partial r_j} - \mathrm{i}k e_j^n \right|^2 d\Gamma_j \le E^1. \tag{17}$$

From (16) and (17), $\{e_0^n\}$ is a bounded sequence in $H^{1/2}(\Gamma)$, and hence $\{e_0^n\}$ has a subsequence $\{e_0^{n_l}\}$ such that $e_0^{n_l} \longrightarrow e_0$ in $H^{1/2}(\Gamma)$ weakly. This indicates that for all $q \in H^{1/2}(\Gamma)$,

$$\int_\Gamma S e_0^{n_l} \overline{q}\, d\Gamma = \langle e_0^{n_l},\, S^* q \rangle \longrightarrow \langle e_0,\, S^* q \rangle = \langle S e_0,\, q \rangle, \tag{18}$$

where S^* is the DtN operator corresponding to the *incoming* radiation condition, and $\langle \cdot,\, \cdot \rangle$ is the duality pairing between $H^{-1/2}(\Gamma)$ and $H^{1/2}(\Gamma)$.

On the other hand, from Lemma 1, (14), and (15), we have

$$\begin{aligned} E^1 &\ge \int_\Gamma |S e_0^n - \mathrm{i}k e_0^n|^2\, d\Gamma \\ &= \int_\Gamma \left\{ |S e_0^n|^2 + k^2 |e_0^n|^2 \right\} d\Gamma - 2k\, \mathrm{Im} \left\{ \int_\Gamma S e_0^n \overline{e_0^n}\, d\Gamma \right\} \\ &\ge \int_\Gamma |S e_0^n|^2\, d\Gamma. \end{aligned}$$

This implies that there exists a subsequence of $\{S e_0^{n_l}\}$, still denoted by $\{S e_0^{n_l}\}$, which converges in $L^2(\Gamma)$ weakly. Thus, we can see from (18) that $S e_0 \in L^2(\Gamma)$ and $S e_0^{n_l} \longrightarrow S e_0$ in $L^2(\Gamma)$ weakly. Further, by the compact imbedding of $H^{1/2}(\Gamma)$ in $L^2(\Gamma)$, $e_0^{n_l} \longrightarrow e_0$ in $L^2(\Gamma)$ strongly, and hence we have

$$\int_\Gamma S e_0^{n_l} \overline{e_0^{n_l}}\, d\Gamma \longrightarrow \int_\Gamma S e_0 \overline{e_0}\, d\Gamma.$$

Now, from (13), we get

$$E^{n+1} = E^1 + 4k \sum_{m=1}^{n} \mathrm{Im} \left\{ \int_\Gamma S e_0^m \overline{e_0^m}\, d\Gamma \right\},$$

and hence $\mathrm{Im} \left\{ \int_\Gamma S e_0^m \overline{e_0^m}\, d\Gamma \right\} \longrightarrow 0$. Thereby we have $\mathrm{Im} \left\{ \int_\Gamma S e_0 \overline{e_0}\, d\Gamma \right\} = 0$. This implies $e_0 = 0$. Therefore, we can conclude that $u_0^n \longrightarrow u$ in $L^2(\Gamma)$.

We can show that $u_j^n \longrightarrow u$ in $H^1(\Omega_j)$ $(1 \le j \le J)$ in the same way as in the proof of Theorem 2.6 in [1]. □

6 Concluding Remarks

We demonstrated the convergence of the parallel Schwarz method of Lions for multiple scattering problems. Many techniques of acceleration of the convergence of the Schwarz method have been developed (see [3, 8]). The investigation of acceleration techniques is yet to be done for the Schwarz method presented in this paper.

References

1. B. Després. Domain decomposition method and the Helmholtz problem. In *Mathematical and Numerical Aspects of Wave Propagation Phenomena (Strasbourg, 1991)*, pp. 44–52. SIAM, Philadelphia, PA, 1991.
2. K. Feng. Finite element method and natural boundary reduction. In *Proceeding of the International Congress of Mathematicians*, Warsaw, Poland, 1983.
3. M.J. Gander, F. Magoulès, and F. Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24(1):38–60, 2002.
4. M.J. Grote and C. Kirsch. Dirichlet-to-Neumann boundary conditions for multiple scattering problems. *J. Comput. Phys.*, 201(2):630–650, 2004.
5. P.-L. Lions. On the Schwarz alternating method. III: a variant for nonoverlapping subdomains. In T.F. Chan, R. Glowinski, J. Périaux, and O. Widlund, editors, *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations , held in Houston, Texas, March 20–22, 1989*, SIAM, Philadelphia, PA, 1990.
6. M. Masmoudi. Numerical solution for exterior problems. *Numer. Math.*, 51:87–101, 1987.
7. R.S. Phillips. On the exterior problem for the reduced wave equation. In *Partial Differential Equations (Proc. Sympos. Pure Math., Vol. XXIII, Univ. California, Berkeley, CA, 1971)*, pp. 153–160, AMS, Providence, RI, 1973.
8. D. Tromeur-Dervout. Aitken-Schwarz method: acceleration of the convergence of the Schwarz method. In F. Magoulès and T. Kako, editors, *Domain Decomposition Methods: Theory and Applications*, chapter 2, pp. 37–64, Gakkotosho, Tokyo, 2006.

Numerical Method for Antenna Radiation Problem by FDTD Method with PML

Takashi Kako[1] and Yoshiharu Ohi[2]

[1] The University of Electro-Communications, Department of Computer Science, Chofu, Tokyo 182-8585, Japan, kako@im.uec.ac.jp
[2] The University of Electro-Communications, Department of Computer Science, Chofu, Tokyo 182-8585, Japan, ohi@sazae.im.uec.ac.jp

Summary. In the numerical simulation of electromagnetic wave radiation from an antenna, the antenna is assumed to be a perfectly conducting obstacle. It was shown numerically that the antenna can be effectively modeled by a highly conducting region occupied by it. The Finite Difference Time Domain method combined with Perfectly Matched Layer gives a flexible numerical methodology for this problem. We apply the method to analyze several radiation problems with different types of antennas such as a birdcage and the Yagi types where the delta gap type power supply model is adopted. For treating an unbounded outer region numerically, we apply a newly developed technique to discretize the PML region with little artificial reflection. Theoretical justification of this procedure for a 1D case was presented in DD17, and effectiveness of this technique was also demonstrated numerically for 2D and 3D cases. We observe a good 3D numerical performance of the method and confirm its usefulness though theoretical justification remains as a future problem.

1 FDTD Method and PML

In this paper, we consider a numerical method for electromagnetic wave propagation in an unbounded domain. The standard numerical method for computing an electromagnetic wave is the FDTD (Finite Difference Time Domain) method introduced by [6]. To solve the problem in the unbounded domain, one must truncate the outer unbounded domain appropriately. For this purpose the PML (Perfectly Matched Layer) which was firstly introduced by [1] is popularly used, where one introduces an artificial magnetic conductivity σ^* in this region. When we discretize the equations in the PML, some artificial reflections are observed in a solution by the original scheme of Berenger. We firstly review a new discretization scheme with fewer reflection introduced by the present authors, [3] for a 1D problem and also applied for 2D and 3D problems. This scheme applied to the 1D problem does not cause any artificial reflection at least in the constant σ^* region. Although we have not proved mathematically the non-existence of the artificial reflection for 2D and 3D cases, we have succeeded in validating the method for these cases numerically.

Secondly, we develop a 3D numerical method to simulate propagation of an RF (Radio-Frequency) wave emitted by various antennas such as the Yagi antennas and birdcage coil antennas used for an MRI (Magnetic Resonance Imaging) device.

In the FDTD method, a finite difference method with a space-time staggered mesh is used for discretization of the Maxwell equation. To solve the problem in an unbounded region, we employ the PML and introduce an artificial absorption term σ^* in the equation to attenuate the wave there. In order to make the computational domain finite we impose a perfectly reflecting boundary condition on the outermost boundary of the PML. The additional boundary condition may introduce a further extra artificial reflection, but the reflection is supposed to be controllable within a negligible level in most applications. The Maxwell equation in non-PML region is written as

$$\frac{\partial}{\partial t}E(t,x) = -\frac{\sigma(x)}{\epsilon}E(t,x) + \frac{1}{\epsilon}\nabla \times H(t,x), \tag{1}$$

$$\frac{\partial}{\partial t}H(t,x) = -\frac{1}{\mu}\nabla \times E(t,x), \tag{2}$$

and in the PML region as

$$\frac{\partial}{\partial t}E(t,x) = -\frac{\sigma(x)}{\epsilon}E(t,x) + \frac{1}{\epsilon}\nabla \times H(t,x), \tag{3}$$

$$\frac{\partial}{\partial t}H(t,x) = -\frac{\sigma^*(x)}{\mu}H(t,x) - \frac{1}{\mu}\nabla \times E(t,x), \tag{4}$$

with $E = (E_x, E_y, E_z)$ the electric field, $H = (H_x, H_y, H_z)$ the magnetic field, ϵ the electric permittivity, μ the magnetic permeability, σ the electric conductivity and σ^* the artificial magnetic conductivity. In the followings, we assume without loss of generality, that $\epsilon = \mu = 1$, and also impose an impedance matching condition, $\sigma^* = \sigma$.

By introducing σ^*, the solution in the non-PML region does not change numerically in 1D, 2D and 3D cases, and we can prove it theoretically in 1D case (see [3]).

In both PML and non-PML regions, in accordance with the idea of [1], we split the variables E and H into two components as $E_x = E_{xy} + E_{xz}$, $H_x = H_{xy} + H_{xz}$ and so on. By using these variables the Maxwell equation is expressed as

$$\frac{\partial}{\partial t}E_{xy}(t,x) = -\sigma_y(x)E_{xy}(t,x) + \frac{\partial(H_{zx}(t,x) + H_{zy}(t,x))}{\partial y}, \tag{5}$$

$$\frac{\partial}{\partial t}E_{xz}(t,x) = -\sigma_z(x)E_{xz}(t,x) - \frac{\partial(H_{yz}(t,x) + H_{yx}(t,x))}{\partial z}, \tag{6}$$

and similar equations derived by permutating x, y, z cyclically and changing the roles of E and H. The formulation for one dimensional case is seen in [3].

New FDTD discretization scheme for the 3D equations is

$$E_{xy}^{n+1}(i+\frac{1}{2},j,k) = A_{xy}E_{xy}^{n}(i+\frac{1}{2},j,k) + B_{xy}\{H_{zx}^{n+\frac{1}{2}}(i+\frac{1}{2},j+\frac{1}{2},k) + H_{zy}^{n+\frac{1}{2}}(i+\frac{1}{2},j+\frac{1}{2},k) - H_{zx}^{n+\frac{1}{2}}(i+\frac{1}{2},j-\frac{1}{2},k) - H_{zy}^{n+\frac{1}{2}}(i+\frac{1}{2},j-\frac{1}{2},k)\}, \quad (7)$$

with the coefficients,

$$A_{xy} = e^{-\sigma_y(j)\Delta t} \quad \text{and} \quad B_{xy} = \frac{\Delta t}{\Delta y}e^{-\sigma_y(j)\Delta t/2}, \quad (8)$$

and so on. In the previous standard scheme by Berenger, the corresponding coefficients are

$$A_{xy} = e^{-\sigma_y(j)\Delta t}, \quad \text{and} \quad B_{xy} = \frac{1-e^{\sigma_y(j)\Delta t}}{\sigma_y(j)\Delta y}. \quad (9)$$

There is another simplified scheme where the coefficients are given as

$$A_{xy} = \frac{1-\frac{\sigma_y(j)\Delta t}{2}}{1+\frac{\sigma_y(j)\Delta t}{2}} \quad \text{and} \quad B_{xy} = \frac{\Delta t}{\Delta y(1+\frac{\sigma_y(j)\Delta t}{2})}. \quad (10)$$

We show comparison of performance among these schemes by numerical examples in Fig. 1. It can be concluded that our new scheme is superior to others.

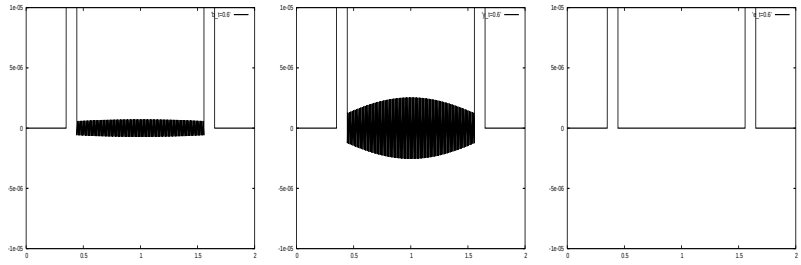


Fig. 1. Comparison of reflection waves at $t = 0.6$ by Berenger's scheme (*left*), simplified scheme (*middle*) and new scheme (*right*).

To check the validity of our method in 3D case, we show a time evolution of the absolute value of the Poynting's vector at an observation point for an initial value problem with a delta function like initial profile. Figure 2 shows that the artificial reflection from the PML region is negligible, although we observe some small reflection wave from the PML region as well as from the outermost boundary.

2 Basic Formulation of Antenna Problem

Among variety of electromagnetic radiation and scattering problems the antenna problem is a special case where a scatterer or an obstacle is a low dimensional singular object.

The antenna is usually modeled as a perfectly conducting obstacle, which constitutes a lower dimensional region in the computational domain such as a flat plate or a parabola panel as 2D region and a line or an array of lines as 1D region.

To make the numerical simulation, we need to calculate the electric current density profile on the line antennas. For this purpose, Pocklington's integral equation for electric current is already known in the case of a straight line antenna, for which the standard numerical methodology is the moment method. There are, however, several demerits of the method, i.e., it is effective only for the time harmonic problem and is not so easy to extend it to more general antenna configurations.

On the other hand, the new method developed by us is free from these demerits as our methodology is based on the FDTD method combined with the PML for solving time dependent problems, and treats the antenna as an electrically highly conductive region, and the current density on the antenna can be calculated afterwards if necessary.

A typical example of an array of line antennas is the Yagi antenna consisting of a power supplier, reflectors and guiders (see [2]). In Fig. 3, the energy density profiles of the wave on x–y plane and x–z plane are shown. The computational region is approximately $2.2 \times 2.2 \times 2.2$ with the PML having the thickness of $6h = 6 \times 2^{-6} \approx 0.1$ with the mesh size $h = 2^{-6}$. The antenna lengths of the supplier, the reflector and two guiders are $29h$, $31h$, $29h$ and $27h$, respectively. At the midpoint of the supplier, we assume a delta gap power supply, i.e., an external source which supplies a time harmonic electric field $\sin(2\pi ft)/h$ with frequency $f = 1.0$ on one mesh point. The spatial mesh size h is $h = 2^{-6} = 1/64$ as stated before and the temporal mesh size τ is $\tau = h/2 = 2^{-7} = 1/128$. Figure 3 shows four spatial profiles of electromagnetic energy density of the radiating wave from a Yagi antenna at time $t = t_0$, $t_0 + (1/4)f$, $t_0 + (1/2)f$ and $t_0 + (3/4)f$ with sufficiently large $t_0 = 5.0 \gg f = 1.0$. One of the most interesting and important problems is to arrange the components of the line antennas so that the best performance of electromagnetic wave radiation to the desirable direction is attained. We leave this problem to be solve in our future study.

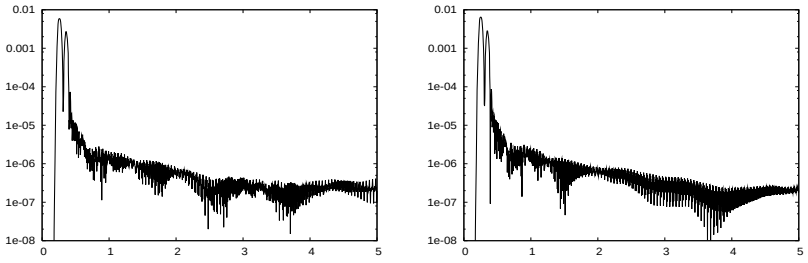


Fig. 2. Time evolution of the absolute value of Poynting's vector for an initial value problem.

3 Application to MRI Problem

As an application of our method, we show an example of the electromagnetic radiation and scattering problems appearing in MRI (Magnetic Resonance Imaging) which is an imaging technology based on NMR (Nuclear Magnetic Resonance). There are many researches on human susceptibility related to a mobile phone (see for example, [4, 5]) because use of electromagnetic wave of radio frequency range is considered to have some unfavorable heating effect on the human body. Though the calculation of SAR (Specific Absorption Rate) is important for this purpose, only a few studies have been carried out on this problem up to now concerning MRI. Hence it is challenging to develop a methodology for the estimation of SAR concerning MRI.

For this purpose we first simulate numerically the propagation of the electromagnetic wave excited by a birdcage coil in MRI device by the FDTD method with the PML. Then by putting a phantom of a human body inside the birdcage coil we estimate SAR in the phantom, by which the possible change of SAR under different coil configuration can be studied.

In Fig. 4 we show examples of numerical simulation on heating of a phantom in MRI with birdcage antennas. The size of computational domain is $2.2 \times 2.2 \times 2.2\,\mathrm{m}^3$ and the thickness of PML is 0.1 m, and the frequency of power supply is 64 MHz. In this example, some specific parts of the phantom body are heated more in comparison with other parts. For example, we observed several typical phenomena, i.e., SAR becomes higher at the positions nearest to the coil as a head and a waist, and also at edges of the body, especially at the edges of convex shape as a head, a shoulder and

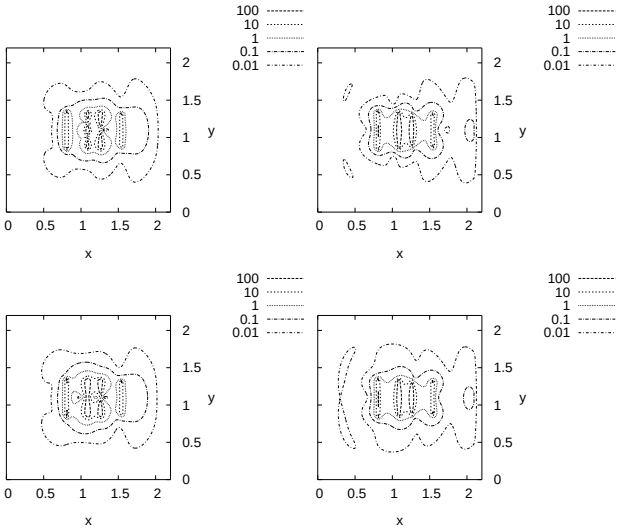


Fig. 3. Examples of numerical simulation on spatial profiles of electromagnetic energy density for a Yagi antennas at different times.

a waist. Naturally, we see that the SAR decreases with increasing the coil dimension (length and radius) and increasing the distance between the body and the coil. The detailed analysis including the optimization of the antenna coil configurations and others based on this methodology is our future problem.

4 Summary and Future Problems

We now summarize our study as follows:

(i) We tested the efficiency of our methodology through a basic antenna configuration such as the Yagi antennas.
(ii) We applied our new scheme to the 3D MRI problem with a source birdcage coil antenna, and computed SAR for a phantom body inside the coil.

Future problems are

(i) the optimal design of various line antennas by using appropriate optimization algorithm (such as gradient method and/or GA);
(ii) the investigation of possible variation of SAR when we increase or decrease the number of leading wires connecting two circular wires of birdcage coil;
(iii) the study of the effect of static background magnetic field configuration as well as the effect of the way of impressing a source voltage through background electric circuit;
(iv) the usage of more realistic CAD models of a human body in its geometric shape and with physical and/or physiological parameters.

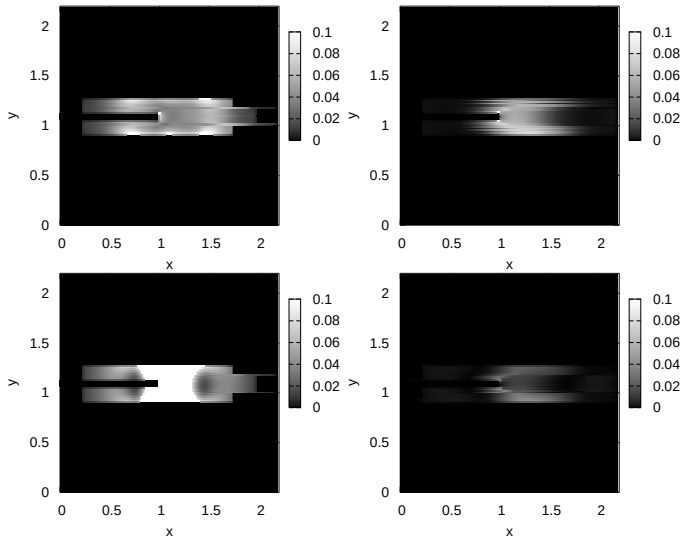


Fig. 4. Examples of numerical simulation on heating of a phantom in MRI with bird cage antennas.

References

1. J.-P. Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.*, 114(2): 185–200, 1994.
2. E. Jones and W.T. Joines. Design of Yagi-Uda antennas using genetic algorithms. *IEEE Trans. Antennas Propag.*, 45(9): 1386–1392, 1997.
3. T. Kako and Y. Ohi. Numerical method for wave propagation problem by FDTD method with PML. In *Domain Decomposition Methods in Science and Engineering XVII, volume 60 of Lecture Notes in Computational Science and Engineering*, 551–558, 2007.
4. K. Sugimura. *Principle of MRI and Imaging Methodology*. Medical view Co. Ltd, 2000.
5. T. Uno. *Electromagnetic Wave and Antenna Analysis by FDTD Method*. Corona Publishing Co. Ltd, Tokyo, 1998.
6. K.S. Yee. Numerical solution of initial boundary value problems involving Maxwell's equation in isotropic media. *IEEE Trans. Antennas Propag.*, 14(3): 302–307, 1966.

On Domain Decomposition Algorithms for Contact Problems with Tresca Friction

Julien Riton[1], Taoufik Sassi[1], and Radek Kučera[2]

[1] LMNO, University of Caen, Caen, France, riton.julien@math.unicaen.fr; Taoufik.Sassi@math.unicaen.fr
[2] Technical University of Ostrava, Ostrava, Czech Republic, radek.kucera@vsb.cz

1 Introduction

Development of numerical methods for the solution of contact problems is a challenging task whose difficulty lies in the non-linear conditions for non-penetration and friction. Recently, many authors proposed to use various numerical algorithms combined with multigrid or domain decomposition techniques; see, e.g., the primal-dual active set algorithm [8], the non-smooth multiscale method [10], or the augmented Lagrangian based algorithm [3]. Another alternative consists in the formulation of suitable iterations solving the elasticity equations for each sub-body separately with certain boundary conditions [5]. In [1], the authors proposed a Dirichlet-Neumann algorithm which takes into account the natural interface for frictionless contact problems. Another improvement has led to a Neumann–Neumann algorithm in which they added two Neumann sub-problems in order to ensure the continuity of normal stresses [2]. Later, various numerical implementations of this approach was given in [7, 9]. In this contribution, we extend the algorithm to two-body contact problems with Tresca friction. The advantage consists in decoupling the non-penetration and friction conditions between the bodies so that they are treated separately by smaller subproblems that may be solved in parallel. With respect to existing (global) algorithms, our method is suitable in situations when material or geometrical qualities of the bodies are considerably different. By numerical experiments, we illustrate that the algorithm is mesh independent for a suitable choice of parameters.

2 Contact Problems with Tresca Friction

Let us consider two elastic bodies occupying bounded domains $\Omega^\alpha \in \mathbb{R}^2$, $\alpha = 1, 2$. Each boundary $\Gamma^\alpha := \partial\Omega^\alpha$ is assumed piecewise continuous and composed of three disjoint, non-empty parts Γ_u^α, Γ_ℓ^α, and Γ_c^α. Each body Ω^α is fixed on Γ_u^α and subject to surface tractions $\boldsymbol{\phi}^\alpha \in \mathbf{L}^2(\Gamma_\ell^\alpha)$. The body forces are denoted by $\boldsymbol{f}^\alpha \in \mathbf{L}^2(\Omega^\alpha)$. In the initial configuration, the bodies possess the common contact interface $\Gamma_c := \Gamma_c{}^1 = \Gamma_c{}^2$, where the unilateral contact with Tresca friction is

considered. The problem consists in finding the displacement field $\boldsymbol{u} = (\boldsymbol{u}^1, \boldsymbol{u}^2)$, $\boldsymbol{u}^\alpha := \boldsymbol{u}_{|\Omega^\alpha}$, and the stress tensor $\sigma = (\sigma(\boldsymbol{u}^1), \sigma(\boldsymbol{u}^2))$ such that for $\alpha = 1, 2$:

$$\left.\begin{aligned} \mathbf{div}\sigma(\mathbf{u}^\alpha) + \boldsymbol{f}^\alpha &= \mathbf{0} \ \text{ in } \ \Omega^\alpha, \\ \sigma(\boldsymbol{u}^\alpha)\boldsymbol{n}^\alpha &= \boldsymbol{\phi}^\alpha \ \text{ on } \ \Gamma_\ell^\alpha, \\ \boldsymbol{u}^\alpha &= \mathbf{0} \ \text{ on } \ \Gamma_u^\alpha \end{aligned}\right\} \tag{1}$$

and σ is related to the strain tensor $e(\boldsymbol{u}^\alpha) = \left(\nabla \boldsymbol{u}^\alpha + (\nabla \boldsymbol{u}^\alpha)^T\right)/2$ by Hooke's law for linear isotropic materials:

$$\sigma_{ij}(\boldsymbol{u}^\alpha) = \mathsf{E}_{ijkh}^\alpha e_{kh}(\boldsymbol{u}^\alpha),$$

where $\mathsf{E} = (\mathsf{E}_{ijkh}^\alpha)_{1\le i,j,k,h\le 2} \in (L^\infty(\Omega^\alpha))^{16}$ is the fourth-order tensor satisfying the symmetry and ellipticity conditions.

We will use the usual notation for the normal and tangential components of the displacement and stress vectors on Γ_c:

$$u_N^\alpha = \boldsymbol{u}^\alpha.\boldsymbol{n}^\alpha, \ \ u_T^\alpha = \boldsymbol{u}^\alpha.\boldsymbol{t}^\alpha, \ \ \sigma_N^\alpha = (\sigma(u^\alpha)\boldsymbol{n}^\alpha).\boldsymbol{n}^\alpha, \ \ {\sigma_T}^\alpha = (\sigma(\boldsymbol{u}^\alpha)\boldsymbol{t}^\alpha).\boldsymbol{n}^\alpha,$$

where $\boldsymbol{n}^\alpha$ denotes the unit outer normal vector to Γ_c^α and $\boldsymbol{t}^\alpha$ is the unit tangential vector satisfying $\boldsymbol{t}^\alpha.\boldsymbol{n}^\alpha = 0$ and $\boldsymbol{t}^1 = -\boldsymbol{t}^2$. On Γ_c, the unilateral contact law is given by

$$\sigma_N^1 = \sigma_N^2 := \sigma_N, \qquad \sigma_T^1 = \sigma_T^2 := \sigma_T, \tag{2}$$

$$[u_N] \le 0, \quad \sigma_N \le 0, \quad \sigma_N[u_N] = 0, \tag{3}$$

where $[u_N] := u_N^1 + u_N^2$ is the jump in the normal direction across the interface Γ_c. The Tresca law of friction is given by

$$\left.\begin{aligned} &|\sigma_T| \le g, \\ &|\sigma_T| < g \Rightarrow [u_T] = 0, \\ &|\sigma_T| = g \Rightarrow \exists \kappa \ge 0 : \ [u_T] = -\kappa\sigma_T \text{ on } \Gamma_c, \end{aligned}\right\} \tag{4}$$

where $g \in \mathrm{L}^2(\Gamma_c)$, $g \ge 0$, is the given slip bound on Γ_c and $[u_T] := u_T^1 + u_T^2$.

Remark 1. In the Coulomb law of friction, g replaces $\mathcal{F}|\sigma_N|$, i.e., the product of the coefficient of friction $\mathcal{F}$ and à-priori unknown absolute value the normal contact stress σ_N.

The problem of finding the couple $\boldsymbol{u} = (\boldsymbol{u}^1, \boldsymbol{u}^2)$ satisfying (1), (2), (3), and (4) will be called $(\mathcal{P})$. Its existence and uniqueness is established in [6].

3 Algorithms and the Implementation

We start with the algebraic formulation of the non-decomposed problem. Let p_α denote the dimension of the finite element space $\mathbb{V}_{0,h}^\alpha$ defined on the triangulation

$\mathcal{T}_h^\alpha$ of Ω^α, $\alpha = 1, 2$, and $p := p_1 + p_2$. Further, let q be the number of contact nodes of Ω^1, i.e., the nodes of $\mathcal{T}_h^1$ lying on $\overline{\Gamma}_c \setminus \overline{\Gamma}_u^1$. As we consider matching grids, the contact nodes of Ω^1 and Ω^2 coincide. By $\mathbf{A} \in \mathbb{R}^{p\times p}$ and $\mathbf{b} \in \mathbb{R}^p$, we denote the stiffness matrix and the load vector, respectively, of the whole structure. Let us note that $\mathbf{A}$, $\mathbf{b}$ can be naturally decomposed into blocks corresponding to Ω^1 and Ω^2 so that $\mathbf{A} = diag(\mathbf{A}_1, \mathbf{A}_2)$, $\mathbf{b} = (\mathbf{b}_1^\top, \mathbf{b}_2^\top)^\top$, where $\mathbf{A}_\alpha \in \mathbb{R}^{p_\alpha \times p_\alpha}$ are symmetric, positive definite and $\mathbf{b}_\alpha \in \mathbb{R}^{p_\alpha}$, $\alpha = 1, 2$. We introduce the matrices $\mathbf{N}_\alpha, \mathbf{T}_\alpha \in \mathbb{R}^{q\times p_\alpha}$, $\alpha = 1, 2$, projecting contact displacements to the directions of $\boldsymbol{n}^\alpha$, $\boldsymbol{t}^\alpha$, respectively, i.e., each row of $\mathbf{N}_\alpha, \mathbf{T}_\alpha$ contains the two components of the corresponding normal $\boldsymbol{n}^\alpha$ and tangential $\boldsymbol{t}^\alpha$ vectors. For sake of simplicity we denote by $\mathbf{B}_\alpha = (\mathbf{N}_\alpha^\top, \mathbf{T}_\alpha^\top)^\top$ that are matrices with orthonormal rows. Finally, the vector $\mathbf{g} \in \mathbb{R}^q$ is determined by the nodal values of g.

The finite element approximation of $(\mathcal{P})$ leads to the following algebraic problem:

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}\mathbf{u}^\top \mathbf{A}\mathbf{u} - \mathbf{u}^\top \mathbf{b} + \sum_{i=1}^{q} g_i |\mathbf{T}_1\mathbf{u}_1 + \mathbf{T}_2\mathbf{u}_2|_i \\ &\text{subject to} \quad \mathbf{N}_1\mathbf{u}_1 + \mathbf{N}_2\mathbf{u}_2 \le \mathbf{0}, \end{aligned} \tag{5}$$

where $\mathbf{u} = (\mathbf{u}_1^\top, \mathbf{u}_2^\top)^\top$, $\mathbf{u}_\alpha \in \mathbb{R}^{p_\alpha}$, $\alpha = 1, 2$ and $|\mathbf{v}| = (|v_1|, |v_2|, ..., |v_q|)^\top$ for $v = (v_1, ..., v_q)^\top$.

The problem (5) can be solved by ALGORITHM 1 and ALGORITHM 2 which are discrete versions of our domain decomposition methods .

Algorithm 1 Let $\boldsymbol{\lambda}^{(0)} = ({\boldsymbol{\lambda}_\nu^{(0)}}^\top, {\boldsymbol{\lambda}_\tau^{(0)}}^\top)^\top \in \mathbb{R}^{2q}$ and $\theta > 0$ be given. For $k \ge 1$ compute $\mathbf{u}_\alpha^{(k)}, \mathbf{w}_\alpha^{(k)} \in \mathbb{R}^{p_\alpha}$, $\alpha = 1, 2$, and $\boldsymbol{\lambda}^{(k)} = ({\boldsymbol{\lambda}_\nu^{(k)}}^\top, {\boldsymbol{\lambda}_\tau^{(k)}}^\top)^\top \in \mathbb{R}^{2q}$ as follows:

(*Step 1*) {*Normal bilateral contact with Tresca friction for* Ω^1.}

$$\begin{aligned} \mathbf{u}_1^{(k)} := \text{argmin}\ &\frac{1}{2}\mathbf{u}_1^\top \mathbf{A}_1\mathbf{u}_1 - \mathbf{u}_1^\top \mathbf{b}_1 + \sum_{i=1}^{q} g_i |\mathbf{T}_1\mathbf{u}_1 - \boldsymbol{\lambda}_\tau^{(k-1)}|_i \\ &\text{subject to } \mathbf{N}_1\mathbf{u}_1 = \boldsymbol{\lambda}_\nu^{(k-1)}; \end{aligned}$$

(*Step 2*) {*Normal unilateral and tangential bilateral contact for* Ω^2.}

$$\begin{aligned} \mathbf{u}_2^{(k)} := \text{argmin}\ &\frac{1}{2}\mathbf{u}_2^\top \mathbf{A}_2\mathbf{u}_2 - \mathbf{u}_2^\top \mathbf{b}_2 \\ &\text{subject to } \boldsymbol{\lambda}_\nu^{(k-1)} + \mathbf{N}_2\mathbf{u}_2 \le \mathbf{0},\ \mathbf{T}_2\mathbf{u}_2 = -\boldsymbol{\lambda}_\tau^{(k-1)}; \end{aligned}$$

(*Step 3*) {*Residual deformation of* Ω^1.}

$$\mathbf{A}_1\mathbf{w}_1^{(k)} = \frac{1}{2}\mathbf{B}_1^\top(\mathbf{B}_1(\mathbf{b}_1 - \mathbf{A}_1\mathbf{u}_1^{(k)}) - \mathbf{B}_2(\mathbf{b}_2 - \mathbf{A}_2\mathbf{u}_2^{(k)}));$$

(*Step 4*) {*Residual deformation of* Ω^2.}

$$\mathbf{A}_2\mathbf{w}_2^{(k)} = \frac{1}{2}\mathbf{B}_2^\top(\mathbf{B}_1(\mathbf{b}_1 - \mathbf{A}_1\mathbf{u}_1^{(k)}) - \mathbf{B}_2(\mathbf{b}_2 - \mathbf{A}_2\mathbf{u}_2^{(k)}));$$

(*Step 5*) {*Relaxation of the contact displacements.*}

$$\boldsymbol{\lambda}^{(k)} = \boldsymbol{\lambda}^{(k-1)} + \theta(\mathbf{B}_1\mathbf{w}_1^{(k)} + \mathbf{B}_2\mathbf{w}_2^{(k)}).$$

In *Step 3* and *Step 4*, we compute deformations of the bodies induced by the non-equilibria of contact stresses on Γ_c. These deformations vanish in the solution due to the transfer condition (2). Below, we will show that the dual formulation simplifies considerably the implementation of the algorithm.

The minimization in *Step 1* is equivalent to the saddle-point problem:

Find $(\mathbf{u}_1, \mathbf{s}_1) \in \mathbb{R}^{p_1} \times \Lambda(\mathbf{g})$ *such that*

$$\mathcal{L}_1(\mathbf{u}_1, \mathbf{s}_1) = \min_{\mathbf{v}_1 \in \mathbb{R}^{p_1}} \max_{\mathbf{r}_1 \in \Lambda(\mathbf{g})} \mathcal{L}_1(\mathbf{v}_1, \mathbf{r}_1) = \max_{\mathbf{r}_1 \in \Lambda(\mathbf{g})} \min_{\mathbf{v}_1 \in \mathbb{R}^{p_1}} \mathcal{L}_1(\mathbf{v}_1, \mathbf{r}_1),$$

where $\mathcal{L}_1 : \mathbb{R}^{p_1} \times \Lambda(\mathbf{g}) \mapsto \mathbb{R}$ is the Lagrangian defined by

$$\mathcal{L}_1(\mathbf{v}_1, \mathbf{r}_1) := \frac{1}{2}\mathbf{v}_1^\top \mathbf{A}_1\mathbf{v}_1 - \mathbf{v}_1^\top \mathbf{b}_1 + \mathbf{r}_1^\top(\mathbf{B}_1\mathbf{v}_1 - \boldsymbol{\lambda}^{(k-1)})$$

with $\Lambda(\mathbf{g}) := \{\mathbf{r}_1 = (\mathbf{r}_{1\nu}^\top, \mathbf{r}_{1\tau}^\top)^\top \in \mathbb{R}^{2q} : |\mathbf{r}_{1\tau}| \leq \mathbf{g}\}$. Eliminating $\mathbf{u}_1$ from the max-min formulation we arrive at the quadratic programming problem:

$$\text{minimize } \frac{1}{2}\mathbf{s}_1^\top \mathbf{C}_1\mathbf{s}_1 - \mathbf{s}_1^\top \mathbf{h}_1 \quad \text{subject to } \mathbf{s}_1 \in \Lambda(\mathbf{g}), \tag{6}$$

where $\mathbf{C}_1 := \mathbf{B}_1\mathbf{A}_1^{-1}\mathbf{B}_1^\top$ is symmetric, positive definite and $\mathbf{h}_1 := \mathbf{B}_1\mathbf{A}_1^{-1}\mathbf{b}_1 - \boldsymbol{\lambda}^{(k-1)}$. After computing $\mathbf{s}_1$ from (6) one can obtain $\mathbf{u}_1^{(k)}$ in *Step 1* by $\mathbf{u}_1^{(k)} = \mathbf{A}_1^{-1}(\mathbf{b}_1 - \mathbf{B}_1^\top \mathbf{s}_1)$.

The minimization problem in *Step 2* can be handled analogously. It is equivalent to the saddle-point problem:

Find $(\mathbf{u}_2, \mathbf{s}_2) \in \mathbb{R}^{p_2} \times \Lambda_+$ *such that*

$$\mathcal{L}_2(\mathbf{u}_2, \mathbf{s}_2) = \min_{\mathbf{v}_2 \in \mathbb{R}^{p_2}} \max_{\mathbf{r}_2 \in \Lambda_+} \mathcal{L}_2(\mathbf{v}_2, \mathbf{r}_2) = \max_{\mathbf{r}_2 \in \Lambda_+} \min_{\mathbf{v}_2 \in \mathbb{R}^{p_2}} \mathcal{L}_2(\mathbf{v}_2, \mathbf{r}_2),$$

where $\mathcal{L}_2 : \mathbb{R}^{p_2} \times \Lambda_+ \mapsto \mathbb{R}$ is the Lagrangian defined by

$$\mathcal{L}_2(\mathbf{v}_2, \mathbf{r}_2) := \frac{1}{2}\mathbf{v}_2^\top \mathbf{A}_2\mathbf{v}_2 - \mathbf{v}_2^\top \mathbf{b}_2 + \mathbf{r}_2^\top(\boldsymbol{\lambda}^{(k-1)} + \mathbf{B}_2\mathbf{v}_2)$$

and $\Lambda_+ := \{\mathbf{r}_2 = (\mathbf{r}_{2\nu}^\top, \mathbf{r}_{2\tau}^\top)^\top \in \mathbb{R}^{2q} : \mathbf{r}_{2\nu} \geq \mathbf{0}\}$. Analogously, this max–min problem leads to the quadratic programming problem:

$$\text{minimize} \frac{1}{2}\mathbf{s}_2^\top \mathbf{C}_2\mathbf{s}_2 - \mathbf{s}_2^\top \mathbf{h}_2 \quad \text{subject to } \mathbf{s}_2 \in \Lambda_+, \tag{7}$$

where $\mathbf{C}_2 := \mathbf{B}_2\mathbf{A}_2^{-1}\mathbf{B}_2^\top$ is again symmetric, positive definite and $\mathbf{h}_2 := \mathbf{B}_2\mathbf{A}_2^{-1}\mathbf{b}_2 - \mathbf{c}+\boldsymbol{\lambda}^{(k-1)}$. After solving (7) one can obtain $\mathbf{u}_2^{(k)}$ in *Step 2* as $\mathbf{u}_2^{(k)} = \mathbf{A}_2^{-1}(\mathbf{b}_2-\mathbf{B}_2^\top\mathbf{s}_2)$.

As (6) and (7) are the minimization problems with strictly quadratic functions constrained by simple inequality bounds, it is appropriate to solve them by the conjugate gradient method combined with the projected gradient technique [4]. Since both problems are independent, one can solve them in parallel.

Step 3 and *Step 4* may be simplified. Let $\mathbf{s}_1^{(k)}$, $\mathbf{s}_2^{(k)}$ be the solutions to (6), (7), respectively, in the k-th step. Since $\mathbf{A}_\alpha\mathbf{u}_\alpha^{(k)} - \mathbf{b}_\alpha + \mathbf{B}_\alpha^\top\mathbf{s}_\alpha^{(k)} = \mathbf{0}$, we get $\mathbf{s}_\alpha^{(k)} = \mathbf{B}_\alpha(\mathbf{b}_\alpha - \mathbf{A}_\alpha\mathbf{u}_\alpha^{(k)})$. Using these results, we arrive at: $\mathbf{A}_\alpha\mathbf{w}_\alpha^{(k)} = \frac{1}{2}\mathbf{B}_\alpha^\top(\mathbf{s}_1^{(k)} - \mathbf{s}_2^{(k)})$, so that the computations of $\mathbf{u}_\alpha^{(k)}$, $\alpha = 1, 2$, can be omitted.

In the second algorithm we obtain the same structure as before, only *Step 1* and *Step 2* are different.

Algorithm 2 (different steps)

(*Step 1*) {*Linear elasticity for* Ω^1.}

$$\mathbf{u}_1^{(k)} := \operatorname{argmin} \frac{1}{2}\mathbf{u}_1^\top\mathbf{A}_1\mathbf{u}_1 - \mathbf{u}_1^\top\mathbf{b}_1 \quad \text{subject to } \mathbf{B}_1\mathbf{u}_1 = \boldsymbol{\lambda}^{(k-1)};$$

(*Step 2*) {*Unilateral contact with Tresca friction for* Ω^2.}

$$\mathbf{u}_2^{(k)} := \operatorname{argmin} \frac{1}{2}\mathbf{u}_2^\top\mathbf{A}_2\mathbf{u}_2 - \mathbf{u}_2^\top\mathbf{b}_2 + \sum_{i=1}^{q} g_i|\boldsymbol{\lambda}_\tau^{(k-1)} + \mathbf{T}_2\mathbf{u}_2|_i \quad \text{subject to } \boldsymbol{\lambda}_\nu^{(k-1)} + \mathbf{N}_2\mathbf{u}_2 \le \mathbf{0};$$

Let us denote the relative precision of the k-th iterative step of ALGORITHM 1,2 by

$$\varepsilon_\lambda^{(k)} := \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\|/\|\boldsymbol{\lambda}^{(k)}\|,$$

where $\|\cdot\|$ stands for the approximation of the $\mathbf{L}^2(\Gamma_c)$-norm. We terminate if $\varepsilon_\lambda^{(k)} \le tol$ for a prescribed tolerance $tol > 0$. In order to increase the efficiency of the algorithm, we initialize the inner iterative solvers in *Step 1* and *Step 2* by the respective results from the previous outer iterate, i.e., by $\mathbf{s}_1^{(k-1)}$ and $\mathbf{s}_2^{(k-1)}$, and we terminate them by an adaptive (inner) terminating tolerance $tol_{in}^{(k)} > 0$. The idea is to choose $tol_{in}^{(k)}$ in such a way that it respects the precision $\varepsilon_\lambda^{(k-1)}$ achieved in the outer loop: $tol_{in}^{(k)} := r_{tol} \times \varepsilon_\lambda^{(k-1)}$, where $0 < r_{tol} < 1$, $\varepsilon_\lambda^{(0)} := 1$.

4 Numerical Experiments

We consider two plane elastic bodies $\Omega^1 = (0,3) \times (1,2)$ and $\Omega^2 = (0,3) \times (0,1)$ made of an isotropic, homogeneous material characterized by Young modulus

21.19×10^{10} and Poisson ratio 0.277 (steel); see Fig.1.(a). The decompositions of Γ^1 and Γ^2 are as follows:

$$\Gamma_u^1 = \{0\} \times (1,2),\ \Gamma_c^1 = (0,3) \times \{1\},\ \Gamma_\ell^1 = \Gamma^1 \backslash \overline{\Gamma_u^1 \cup \Gamma_c^1},$$
$$\Gamma_u^2 = \{0\} \times (0,1),\ \Gamma_c^2 = (0,3) \times \{1\},\ \Gamma_\ell^2 = \Gamma^2 \backslash \overline{\Gamma_u^2 \cup \Gamma_c^2}.$$

The volume forces vanish for both bodies. The non-vanishing surface tractions $\boldsymbol{\phi}^1 = (\phi_1^1, \phi_2^1)$ act on Γ_ℓ^1 so that

$$\phi_1^1(x,2) = 0,\ \phi_2^1(x,2) = \phi_{2,L}^1 + \phi_{2,R}^1 x,\ x \in (0,3),$$
$$\phi_1^1(3,y) = \phi_{1,B}^1(2-y) + \phi_{1,U}^1(y-1),$$
$$\phi_2^1(3,y) = \phi_{2,B}^1(2-y) + \phi_{2,U}^1(y-1),\ \ y \in (1,2),$$

where $\phi_{2,L}^1 = -6e7$, $\phi_{2,R}^1 = -1e7$, $\phi_{1,B}^1 = 2e7$, $\phi_{1,U}^1 = 2e7$, $\phi_{2,B}^1 = 4e7$, and $\phi_{2,U}^1 = 2e7$. The slip bound is $g = 1.7e7$. Fig.1.(b–d) show results of the computations.

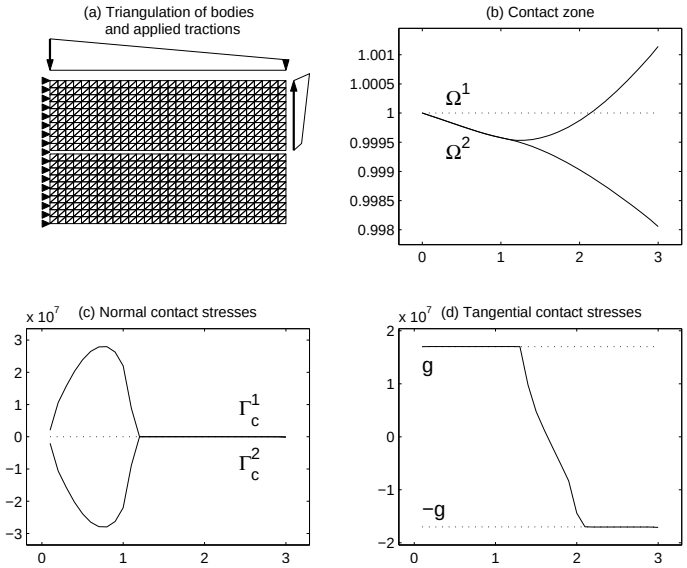


Fig. 1. Geometry and results.

In tables below we compare the performance of ALGORITHMS 1 and 2 for various values of θ and degrees of freedom p and q. We set $tol = 10^{-4}$, $r_{tol} = 0.1$ and we report the number of outer and inner iterations (*out*/*inn*). Since *inn* is proportional to computing time, it characterizes the total complexity of the algorithm. Here the symbol "-" means that the terminating tolerance is not achieved after the 100th iteration. The numerical experiments show higher efficiency of Algorithm 1 in which the non-linear conditions of non-penetration and friction are decoupled into *Step 1* and *Step 2*.

Table 1. Algorithm 1, *out/inn* for various θ.

p/q	$\theta = 0.1$	$\theta = 0.2$	$\theta = 0.3$	$\theta = 0.4$
12672/384	69/835	36/537	26/473	–
19680/480	69/845	37/574	25/445	–
23760/528	70/805	37/585	25/469	–
28224/576	69/845	37/591	25/479	–
38304/672	69/890	36/598	26/490	–
49920/768	70/881	36/610	25/497	–

Table 2. Algorithm 2, *out/inn* for various θ.

p/q	$\theta = 0.1$	$\theta = 0.2$	$\theta = 0.3$	$\theta = 0.4$
12672/384	86/959	47/571	91/921	–
19680/480	86/961	48/587	98/983	–
23760/528	86/961	47/587	99/1021	–
28224/576	87/991	47/600	–	–
38304/672	86/979	48/603	–	–
49920/768	87/1000	47/588	91/952	–

5 Conclusions and Comments

We have presented two different ways of decomposing unilateral contact problems with Tresca friction. According to the previous analysis, one can say that the variant with the decoupled non-penetration and friction conditions is more efficient. The theoretical proof of the convergence will be presented elsewhere. It is based on the Banach fixed point theorem applied to an appropriate mapping that is Lipschitzian and contractive in a suitable norm equivalent to the norm of the trace space $\mathbf{H}^{1/2}(\Gamma_c)$ (see [7] for the frictionless case).

The algorithm can be easily extend to the solution of problems with Coulomb friction as well as for 3D problems. In 3D, the inner minimization will be performed by the method of [11] that treats circular constraints arising from the friction law.

Acknowledgement. The third author acknowledges the support of the project MSM 6198910027 of the Ministry of Education of the Czech Republic and of the project GAČR 101/08/0574 of the Grant Agency of the Czech Republic.

References

1. G. Bayada, J. Sabil, and T. Sassi. Neumann-Dirichlet algorithm for unilateral contact problems: Convergence results. *C.R. Acad. Sci. Paris, Ser.I*, 335:381–386, 2002.
2. G. Bayada, J. Sabil, and T. Sassi. A Neumann–Neumann domain decomposition algorithm for the Signorini problem. *Appl. Math. Lett.*, 17:1153–1159, 2004.

3. Z. Dostál, T. Kozubek, P. Horyl, T. Brzobohatý, and A. Markopoulos. Scalable TFETI algorithm for two dimensional multibody contact problems with friction. *J. Comput. Appl. Math. (accepted)*, 2010.
4. Z. Dostál and J. Schőberl. Minimizing quadratic functions over non-negative cone with the rate of convergence and finite termination. *Comput. Optim. Appl.*, 30(1):23–44, 2005.
5. C. Eck and B.I. Wohlmuth. Convergence of a contact-Neumann iteration for the solution of two-body contact problems. *Math. Mod. Meth. Appl. Sci.*, 13(8):1103–1118, 2003.
6. J. Haslinger, I. Hlaváček, and J. Nečas. Numerical methods for unilateral problems in solid mechanics. In P.G. Ciarlet and J.L. Lions, editors, *Handbook of Numerical Analysis*, pp. 313–485. Elsevier Science, Amsterdam, 1996.
7. J. Haslinger, R. Kučera, and T. Sassi. A domain decomposition algorithm for contact problems: analysis and implementation. *Math. Model. Nat. Phenom.*, 1:123–146, 2009.
8. S. Hűeber, G. Stadler, and B.I. Wohlmuth. A primal-dual active set algorithm for three-dimensional contact problems with Coulomb friction. *SIAM J. Sci. Comput.*, 30(2):527–596, 2008.
9. M.A. Ipopa. *Domain Decomposition Algorithms for Contact Problems: Convergence and Numerical Nimulations*. PhD thesis, University of Caen, Caen 2008.
10. R. Krause. A non-smooth multiscale method for solving frictional two-body contact problems in 2D and 3D with multigrid efficiency. *SIAM J. Sci. Comput.*, 31(2):1399–1423, 2009.
11. R. Kučera. Convergence rate of an optimization algorithm for minimizing quadratic functions with separable convex constraints. *SIAM J. Optim.*, 19(2):846–862, 2008.

Numerical Solution of Linear Elliptic Problems with Robin Boundary Conditions by a Least-Squares/Fictitious Domain Method

Roland Glowinski[1] and Qiaolin He[2]

[1] Department of Mathematics, University of Houston, Houston, TX 77204, USA; Institute of Advanced Study, The Hong Kong University of Science and Technology, Kowloon, Hong Kong, angelarim@aol.com
[2] Department of Mathematics, The Hong Kong University of Science and Technology, Kowloon, Hong Kong, hqlaa@ust.hk

1 Introduction

Motivated by the numerical simulation of particulate flow with slip boundary conditions at the interface fluid/particles, our goal, in this publication, is to discuss a fictitious domain method for the solution of linear elliptic boundary value problems with Robin boundary conditions. The method is of the virtual control type and relies on a least-squares formulation making the problem solvable by a conjugate gradient algorithm operating in a well chosen control space. Numerical results are presented; they suggest optimal orders of convergence for the finite element implementation of our fictitious domain method. A (brief) history of fictitious domain methods can be found in, e.g., [[3], Chap. 8].

2 Formulation of the Boundary Value Problem

Let Ω and ω be two bounded domains of $\mathbf{R}^d$, such that $d \geq 1$ and $\overline{\omega} \subset \Omega$ (see Fig. 1). We denote by Γ and γ the boundaries of Ω and ω, respectively. The Robin–Dirichlet

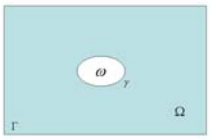


Fig. 1. Problem geometry.

problem under consideration reads as follows:

$$\alpha\psi - \mu\nabla^2\psi = f \quad \text{in} \quad \Omega \setminus \overline{\omega},$$
$$\psi = g_0 \quad \text{on} \quad \Gamma, \quad \mu\left(\frac{\partial\psi}{\partial n} + \frac{\psi}{l_s}\right) = g_1 \quad \text{on} \quad \gamma, \tag{1}$$

where: $\alpha \geq 0, \mu > 0, f \in L^2(\Omega \setminus \overline{\omega}), g_0 \in H^{3/2}(\Gamma), g_1 \in H^{1/2}(\gamma)$, $\mathbf{n}$ is the unit normal vector at γ pointing outward of $\Omega \setminus \overline{\omega}$ and l_s is a characteristic distance. We assume that Ω is convex and that γ is smooth. Problem (1) has a unique solution in $H^2(\Omega \setminus \overline{\omega})$ which is also the solution of the following linear variational problem:

$$\psi \in H^1(\Omega \setminus \overline{\omega}), \psi = g_0 \quad \text{on} \quad \Gamma,$$
$$\alpha \int_{\Omega\setminus\overline{\omega}} \psi\varphi dx + \mu \int_{\Omega\setminus\overline{\omega}} \nabla\psi \cdot \nabla\varphi dx + \frac{\mu}{l_s}\int_\gamma \psi\varphi d\gamma$$
$$= \int_{\Omega\setminus\overline{\omega}} f\varphi dx + \int_\gamma g_1\varphi d\gamma, \quad \forall\varphi \in V_0, \tag{2}$$

where $dx = dx_1 ... dx_d$ and $V_0 = \{\varphi | \varphi \in H^1(\Omega \setminus \overline{\omega}), \quad \varphi = 0 \quad \text{on} \quad \Gamma\}$.

3 A Least-Squares/Fictitious Domain Method for the Solution of Problem (1), (2)

3.1 A Fictitious Domain Formulation of Problem (1), (2)

We proceed as follows to define a fictitious domain variant of problem (1), (2):

(i) With $v \in L^2(\omega)$ we associate $\widetilde{f}(v)$ defined by

$$\widetilde{f}(v) \in L^2(\Omega), \quad \widetilde{f}(v)|_{\Omega\setminus\overline{\omega}} = f, \quad \widetilde{f}(v)|_\omega = v, \tag{3}$$

and then $\{\psi_1, \psi_2\}$ solution of the following elliptic system:

$$\alpha\psi_1 - \mu\nabla^2\psi_1 = \widetilde{f}(v) \quad \text{in} \quad \Omega, \quad \psi_1 = g_0 \quad \text{on} \quad \Gamma, \tag{4}$$
$$\alpha\psi_2 - \mu\nabla^2\psi_2 = v \quad \text{in} \quad \omega, \quad \mu\frac{\partial\psi_2}{\partial n} = \frac{\mu}{l_s}\psi_1 - g_1 \quad \text{on} \quad \gamma. \tag{5}$$

Both problems (4) and (5) have a unique solution in $H^1(\Omega)$ and $H^1(\omega)$, respectively (actually, ψ_1 and ψ_2 have both the H^2–regularity).

(ii) We define $\mathbf{A} : L^2(\omega) \to H^1(\omega)$ by

$$\mathbf{A}(v) = (\psi_2 - \psi_1)\,|_\omega\,. \tag{6}$$

Operator $\mathbf{A}$ is clearly affine and continuous.

(iii) We observe that if v verifies $\mathbf{A}(v) = 0$, we then have $\psi_2 = \psi_1$ on ω and it is easy to see that the H^2–regularity of ψ_1 and ψ_2 implies that $\psi_1|_{\Omega\setminus\overline{\omega}} = \psi$, where ψ is the solution of problem (1), (2). The problem is now to solve the functional equation

$$\mathbf{A}(u) = 0. \tag{7}$$

Indeed, the functional Eq. (7) has infinitely many solutions, but among these solutions only one is of minimal norm in $L^2(\omega)$.

Remark 1. Problem (7) can be viewed as an exact controllability problem in the sense of [2] (and as a virtual control problem in the sense of [4]). If a conjugate gradient algorithm is applied to a least-squares variant of (7) starting with 0 as initial guess, we have convergence to the unique solution of problem (7) of minimal norm in $L^2(\omega)$.

3.2 A Least-Squares Formulation of Problem (7)

A "reasonable" least-squares formulation of (7) reads as follows:

$$\begin{aligned} &\text{Find} \quad u \in L^2(\omega) \quad \text{such that} \\ &J(u) \leq J(v), \quad \forall \quad v \in L^2(\omega), \end{aligned} \tag{8}$$

with

$$J(v) = \frac{1}{2}\int_{\omega} \left[\alpha|\psi_2 - \psi_1|^2 + \mu|\nabla(\psi_2 - \psi_1)|^2\right] dx, \tag{9}$$

ψ_1 and ψ_2 in (9) being obtained from v via the solution of the elliptic boundary value problems (4) and (5), respectively. The functional J is clearly convex and C^∞ over $L^2(\omega)$. Any solution of problem (7) is a solution of the minimization problem (of the *virtual control* type) (8). Such a solution is characterized by

$$DJ(u) = 0, \tag{10}$$

where $DJ(\cdot)$ is the differential of functional J. Using classical methods from Control Theory (see, e.g., [2]), we can show that

$$\forall v \in L^2(\omega), DJ(v) = (p_1 - \psi_1)|_\omega + \psi_2, \tag{11}$$

where in (11), ψ_1 and ψ_2 are defined from v via the solution of (4) and (5), respectively, and where p_1 is the the unique solution of the following adjunct equation (written directly in variational form, here):

$$\begin{aligned} &p_1 \in H_0^1(\Omega), \\ &\int_\Omega [\alpha p_1\varphi + \mu\nabla p_1 \cdot \nabla\varphi]\, dx = \int_\omega [\alpha(\psi_1 - \psi_2)\varphi + \mu\nabla(\psi_1 - \psi_2)\cdot\nabla\varphi]\, dx \\ &\qquad + \frac{\mu}{l_s}\int_\gamma (\psi_2 - \psi_1)\varphi d\gamma, \quad \forall \quad \varphi \in H_0^1(\Omega). \end{aligned} \tag{12}$$

4 On the Conjugate Gradient Solution of the Least-Squares Problem (8)

In order to solve the (linear) least-squares problem (8), we advocate a conjugate gradient algorithm operating in the space $L^2(\omega)$; this algorithm reads as follows:

$$u^0 \quad \text{is given in} \quad L^2(\omega); \tag{13}$$

solve

$$\psi_1^0 \in H^1(\Omega),$$
$$\alpha\psi_1^0 - \mu\nabla^2\psi_1^0 = \widetilde{f}(u^0) \quad \text{in} \quad \Omega, \quad \psi_1^0 = g_0 \quad \text{on} \quad \Gamma, \tag{14}$$

$$\psi_2^0 \in H^1(\omega),$$
$$\alpha\psi_2^0 - \mu\nabla^2\psi_2^0 = u^0 \quad \text{in} \quad \omega, \quad \mu\frac{\partial\psi_2^0}{\partial n} = \frac{\mu}{l_s}\psi_1^0 - g_1 \quad \text{on} \quad \gamma, \tag{15}$$

$$p_1^0 \in H_0^1(\Omega),$$
$$\int_\Omega \left[\alpha p_1^0\varphi + \mu\nabla p_1^0\cdot\nabla\varphi\right]dx = \int_\omega \left[\alpha(\psi_1^0 - \psi_2^0)\varphi + \mu\nabla(\psi_1^0 - \psi_2^0)\cdot\nabla\varphi\right]dx$$
$$+ \frac{\mu}{l_s}\int_\gamma (\psi_2^0 - \psi_1^0)\varphi d\gamma, \quad \forall \quad \varphi \in H_0^1(\Omega), \tag{16}$$

and set

$$g^0 = (p_1^0 - \psi_1^0)|_\omega + \psi_2^0, \quad w^0 = g^0. \tag{17}$$

For $n \geq 0$, assuming that u^n, g^n and w^n are known, the last two different from 0, we compute u^{n+1}, g^{n+1} and w^{n+1} as follows (with χ_ω the characteristic function of ω):
Solve

$$\overline{\psi}_1^n \in H_0^1(\Omega),$$
$$\alpha\overline{\psi}_1^n - \mu\nabla^2\overline{\psi}_1^n = w^n\chi_\omega \quad \text{in} \quad \Omega, \quad \overline{\psi}_1^n = 0 \quad \text{on} \quad \Gamma, \tag{18}$$

$$\overline{\psi}_2^n \in H^1(\omega),$$
$$\alpha\overline{\psi}_2^n - \mu\nabla^2\overline{\psi}_2^n = w^n \quad \text{in} \quad \omega, \quad \mu\frac{\partial\overline{\psi}_2^n}{\partial n} = \frac{\mu}{l_s}\overline{\psi}_1^n \quad \text{on} \quad \gamma, \tag{19}$$

$$\overline{p}_1^n \in H_0^1(\Omega),$$
$$\int_\Omega \left[\alpha\overline{p}_1^n\varphi + \mu\nabla\overline{p}_1^n\cdot\nabla\varphi\right]dx = \int_\omega \left[\alpha(\overline{\psi}_1^n - \overline{\psi}_2^n)\varphi + \mu\nabla(\overline{\psi}_1^n - \overline{\psi}_2^n)\cdot\nabla\varphi\right]dx$$
$$+ \frac{\mu}{l_s}\int_\gamma (\overline{\psi}_2^n - \overline{\psi}_1^n)\varphi d\gamma, \quad \forall \quad \varphi \in H_0^1(\Omega), \tag{20}$$

and set

$$\overline{g}^n = (\overline{p}_1^n - \overline{\psi}_1^n)|_\omega + \overline{\psi}_2^n. \tag{21}$$

Next, compute

$$\rho_n = \frac{\int_\omega |g^n|^2 dx}{\int_\omega \overline{g}^n w^n dx} \tag{22}$$

and

$$u^{n+1} = u^n - \rho_n w^n, \tag{23}$$

$$g^{n+1} = g^n - \rho_n \overline{g}^n. \tag{24}$$

If $\frac{\int_\omega |g^{n+1}|^2 dx}{\max\left\{\int_\omega |g^0|^2 dx, \int_\omega |u^{n+1}|^2 dx\right\}} \leq tol$, take $u = u^{n+1}$ and $\psi = \psi_1^{n+1}|_{\Omega\setminus\overline{\omega}}$; else compute

$$\gamma_n = \frac{\int_\omega |g^{n+1}|^2 dx}{\int_\omega |g^n|^2 dx} \tag{25}$$

and set

$$w^{n+1} = g^{n+1} + \gamma_n w^n. \tag{26}$$

Do $n+1 \rightarrow n$ and return to (18).

5 On the Finite Element Implementation of the Least-Squares/ Fictitious Domain Methodology

5.1 Generalities

We (briefly) discuss in this section the finite element implementation of the least-squares/fictitious domain methodology described in Sects. 3 and 4. We will assume that $\overline{\omega} \subset \Omega \subset \mathbf{R}^2$ and that Ω is convex and/or has a smooth boundary. We assume also that γ is smooth. For simplicity, we still denote by ω and Ω the polygonal approximations of the above domains. From the triangulations $\mathcal{T}_{h_1}$ of Ω and $\mathcal{T}_{h_2}$ of ω, we define the following finite dimensional spaces:

$$V_{h_1} = \{\varphi | \varphi \in C^0(\overline{\Omega}), \varphi|_T \in P_1, \forall T \in \mathcal{T}_{h_1}\}, \tag{27}$$

$$V_{0h_1} = \{\varphi | \varphi \in V_{h_1}, \varphi = 0 \quad \text{on} \quad \Gamma\}, \tag{28}$$

and

$$V_{h_2} = \{\varphi | \varphi \in C^0(\overline{\omega}), \varphi|_T \in P_1, \forall T \in \mathcal{T}_{h_2}\}, \tag{29}$$

P_1 being the space of the polynomials of two variables of degree ≤ 1. We will use $\mathbf{h}$ to denote the pair $\{h_1, h_2\}$. The finite element spaces V_{h_1}, V_{0h_1} and V_{h_2} are finite dimensional approximations to $H^1(\Omega), H_0^1(\Omega)$ and $H^1(\omega)$, respectively. Similarly, we will use V_{h_2} to approximate the "control" space $L^2(\omega)$. As usual, h_1 (resp., h_2) denotes the length of the longest edge(s) of $\mathcal{T}_{h_1}$ (resp., $\mathcal{T}_{h_2}$).

5.2 Finite Element Approximation of the Least-Squares Problem (8)

We approximate the least-squares problem (8) by

$$\begin{aligned} &\text{Find} \quad u_{\mathbf{h}} \in V_{h_2} \quad \text{such that} \\ &J_{\mathbf{h}}(u_{\mathbf{h}}) \leq J_{\mathbf{h}}(v), \quad \forall \quad v \in V_{h_2}, \end{aligned} \tag{30}$$

where

$$J_{\mathbf{h}}(v) = \frac{1}{2}\int_{\omega} \left[\alpha|\psi_2 - \pi_2\psi_1|^2 + \mu|\nabla(\psi_2 - \pi_2\psi_1)|^2\right] dx. \tag{31}$$

In (31), ψ_1 is the solution of the following fully discrete Dirichlet problem:

$$\begin{aligned} &\psi_1 \in V_{h_1}, \quad \psi_1 = g_{0h_1} \quad \text{on} \quad \Gamma, \\ &\int_{\Omega} [\alpha\psi_1\varphi + \mu\nabla\psi_1 \cdot \nabla\varphi]\, dx = \int_{\Omega} f_{h_1}\varphi dx + \int_{\omega} v\pi_2\varphi dx, \quad \forall \quad \varphi \in V_{0h_1}, \end{aligned} \tag{32}$$

where: (i) g_{0h_1} is an approximation of g_0. (ii) $f_{h_1} \in V_{h_1}$; it approximates f over $\Omega \setminus \overline{\omega}$ and vanishes over ω. (iii) $\pi_2 : C^0(\overline{\Omega}) \rightarrow V_{h_2}$ is the interpolation operator defined as follows

$$\pi_2\varphi = \sum_{i=1}^{N_{h_2}} \varphi(Q_i) w_{2i}, \quad \forall \quad \varphi \in C^0(\overline{\Omega}), \tag{33}$$

$\{Q_i\}_{i=1}^{N_{h_2}}$ being the set of the vertices of $\mathcal{T}_{h_2}$ and w_{2i} the P_1- shape function associated with the vertex Q_i (we clearly have N_{h_2} = dimension of V_{h_2}). Returning to (31), the function ψ_2 there is the solution of the following discrete Neumann problem

$$\begin{aligned} &\psi_2 \in V_{h_2}, \\ &\int_{\omega} [\alpha\psi_2\varphi + \mu\nabla\psi_2 \cdot \nabla\varphi]\, dx = \int_{\omega} v\varphi dx \\ &+ \frac{\mu}{l_s}\int_{\gamma} (\pi_2\psi_1 - g_{1h_2})\varphi d\gamma, \quad \forall \quad \varphi \in V_{h_2}, \end{aligned} \tag{34}$$

g_{1h_2} being an approximation of g_1. As a discrete analogue of problem (8), the finite dimensional least-squares problem (30) is also of the virtual control type and well-suited to solution by a conjugate gradient algorithm operating in the space V_{h_2}. Due to page limitation we cannot describe this algorithm here; actually, this discrete analogue of algorithm (13)–(26) will be fully described in [1].

6 Numerical Experiments

As test problem, we consider the particular case of problem (1) associated with: (i) $\Omega = (0,4) \times (0,4), \omega = \left\{\{x_1, x_2\} \mid \left(\frac{x_1 - G_1}{a}\right)^2 + \left(\frac{x_2 - G_2}{b}\right)^2 < 1\right\}$ with $G_1 = G_2 = 2, a = 1/4$ and $b = 1/8$. (ii) $f(x_1, x_2) = \alpha(x_1^3 - x_2^3) - 6\mu(x_1 - x_2)$, $\forall \{x_1, x_2\} \in \Omega\setminus\overline{\omega}$. (iii) $g_0 = x_1^3 - x_2^3, g_1 = \mu\left[3(n_1x_1^2 - n_2x_2^2) + (x_1^3 - x_2^3)/l_s\right]$, $\{n_1, n_2\} = \mathbf{n}$ being the unit normal vector at γ pointing to ω. (iv) $\alpha = 100, \mu = 0.1$

and $l_s = 0.1$. The unique solution of problem (1), associated with the above data, in $H^1(\Omega\backslash\overline{\omega})$ is given by

$$\psi(x_1, x_2) = x_1^3 - x_2^3.$$

Concerning the finite element implementation of the least-squares/fictitious domain methodology discussed in Sects. 3 and 4, we employed for $\mathcal{T}_{h_1}$ (resp., $\mathcal{T}_{h_2}$) uniform triangulations of Ω (resp., triangulations of ω) like the one shown in Fig. 2(*left*) (resp., Fig. 2(*right*)). We used $u^0 = 0$ to initialize the discrete analogue of the conjugate gradient algorithm (13)–(26), and took $tol = 10^{-10}$ for the stopping criterion.

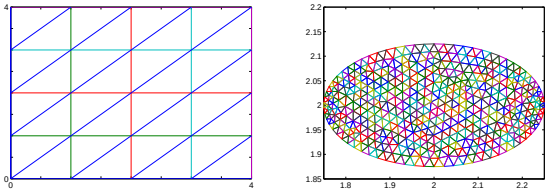


Fig. 2. A uniform triangulation of Ω(*left*); A triangulation of ω(*right*).

In Table 1, we report on the mesh sizes $h_2 = 1/40$ and $h_1 = 1/5, 1/10, 1/20$ and $1/40$: (i) The number of conjugate gradient iterations necessary to achieve convergence. (ii) Various norms of the approximation error. These results suggest: (a) For h_1 small enough, the number of iterations varies slowly with h_1. (b) $\| \psi_{\mathbf{h}} - \psi \|_{L^\infty(\Omega\backslash\overline{\omega})} \approx O(h_1^2)$, $\| \psi_{\mathbf{h}} - \psi \|_{L^2(\Omega\backslash\overline{\omega})} \approx O(h_1^2)$ and $\| \psi_{\mathbf{h}} - \psi \|_{H^1(\Omega\backslash\overline{\omega})} \approx O(h_1)$. Concerning the decay of the cost function $J_{\mathbf{h}}$, we have, if $\mathbf{h} = \{1/10, 1/40\}$ (resp., $\mathbf{h} = \{1/20, 1/40\}$), $J_{\mathbf{h}}(u^0) = 3.67$ (resp., $J_{\mathbf{h}}(u^0) = 3.35$) and $J_{\mathbf{h}}(u^{61}) = 4.20 \times 10^{-9}$ (resp., $J_{\mathbf{h}}(u^{59}) = 5.11 \times 10^{-9}$), showing clearly that the computed approximations of ψ_1 and ψ_2 match quite well over ω. In order to further investigate the convergence properties of the methodology discussed in the above sections we performed computations with $h_2 = 1/20$ and $h_1 = 1/10, 1/20, 1/40$ and $1/80$. The corresponding results have been reported in Table 2. From these results we observe that: (i) If $h_1 \geq h_2$, the number of iterations necessary to achieve convergence does not vary significantly with h_1; on the other hand this number of iterations increases sharply when h_1 decreases below h_2. (ii) The various approximation errors vary as expected (that is as in Table 1) if $h_1 \geq h_2$; on the other hand, they vary quite differently if $h_1 < h_2$, the only one behaving "nicely" being $\| \psi_{\mathbf{h}} - \psi \|_{H^1(\Omega\backslash\overline{\omega})}$, which shows a text-book $O(h_1)$ behavior as h_1 varies over the interval $[1/80, 1/10]$. From these results we suggest to take $h_1 = h_2$ to be on the safe side.

Remark 2. The results reported in [1] show a sharp decrease of the number of iteration when the methodology discussed here is applied to the solution of parabolic problems, including situations where ω is moving.

Table 1. Summary of numerical results ($h_2 = 1/40$).

h_1	Number of iterations	$\| \psi - \psi_h \|_{L^\infty(\Omega\setminus\overline{\omega})}$	$\| \psi - \psi_h \|_{L^2(\Omega\setminus\overline{\omega})}$	$\| \psi - \psi_h \|_{H^1(\Omega\setminus\overline{\omega})}$
1/5	34	0.1046	7.8370E-03	0.2855
1/10	61	2.1845E-02	1.9028E-03	0.1423
1/20	59	4.5840E-03	4.7015E-04	7.1089E-02
1/40	68	1.1385E-03	1.1708E-04	3.5518E-02

Table 2. Summary of numerical results ($h_2 = 1/20$).

h_1	Number of iterations	$\| \psi - \psi_h \|_{L^\infty(\Omega\setminus\overline{\omega})}$	$\| \psi - \psi_h \|_{L^2(\Omega\setminus\overline{\omega})}$	$\| \psi - \psi_h \|_{H^1(\Omega\setminus\overline{\omega})}$
1/10	33	2.1845E-02	1.9038E-03	0.1424
1/20	36	4.5840E-03	4.6807E-04	7.1063E-02
1/40	114	2.1385E-03	1.0163E-04	3.5434E-02
1/80	85	3.1514E-03	5.2854E-05	1.7532E-02

Acknowledgments The first author acknowledge the support of the Institute for Advanced Study (IAS) at The Hong Kong University of Science and Technology. The work is partially supported by grants from RGC CA05/06.SC01 and RGC-CERG 603107.

References

1. R. Glowinski and Q. He. A least-squares /fictitious domain method for linear elliptic problems with Robin boundary conditions. 2009. (in preparation).
2. R. Glowinski, J.L. Lions, and J .He. *Exact and Approximate Controllability for Distributed Parameter Systems: A Numerical Approach*. Cambridge University Press, Cambridge, 2008.
3. R. Glowinski. Finite element methods for incompressible viscous flow. In *Handbook of Numerical Analysis, Vol. IX*, pp. 3–1176. North-Holland, Amsterdam, 2003.
4. J.L. Lions. Virtual and effective control for distributed systems and decomposition of everything. *J. Anal. Math.*, 80:257–297, 2000.

An Uzawa Domain Decomposition Method for Stokes Problem

Jonas Koko[1] and Taoufik Sassi[2]

[1] LIMOS, Université Blaise-Pascal – CNRS UMR 6158
Campus des Cézeaux, 63173 Aubière cedex, France, koko@isima.fr
[2] LMNO, Université de Caen – CNRS UMR 6139
BP 5186, 14032 Caen Cedex, France, Taoufik.Sassi@math.unicaen.fr

1 Introduction

The Stokes problem plays an important role in computational fluid dynamics since it is encountered in the time discretization of (incompressible) Navier-Stokes equations by operator-splitting methods [2, 3]. Space discretization of the Stokes problem leads to large scale ill-conditioned systems. The Uzawa (preconditioned) conjugate gradient method is an efficient method for solving the Stokes problem. The Uzawa conjugate gradient method is a decomposition coordination method with coordination by a Lagrange multiplier.

The paper is organized as follows. In the next section we recall the Stokes problem in its strong and constrained minimization formulations. Then we introduce an additional (interface) continuity condition in the resulting constrained minimization problem and we derive a decomposition coordination method with two multiplier: the pressure (for the divergence free condition) and the interface multiplier (for the continuity condition). A domain decomposition algorithm which solves on each step an uncoupled scalar Poisson sub-problem is defined in § 3.3 and the paper concludes by several numerical realizations.

2 Model Problem

Let Ω be a bounded domain in $\mathbb{R}^d$ ($d = 2, 3$) with Lipschitz-continuous boundary Γ. We consider in Ω the Stokes problem

$$\alpha u - \nu \Delta u + \nabla p = f, \quad \text{in } \Omega, \tag{1}$$

$$\nabla \cdot u = 0, \quad \text{in } \Omega, \tag{2}$$

$$u = 0, \quad \text{on } \Gamma, \tag{3}$$

where $u = u(x)$ is the velocity vector, $p = p(x)$ is the pressure and $f = f(x)$ is the field of external forces.

In (1), $\alpha \geq 0$ is an arbitrary constant. If $\alpha = 0$, (1)–(2) is the classical Stokes problem. The constant $\nu \geq 0$ is the kinematic viscosity and if $\nu = 0$, (1)–(2) turns out to be L^2 projection encountered in time discretization of Navier-Stokes equations (see e.g. [2, 3]).

We need the following functional spaces

$$V = \left\{ v \in (H^1(\Omega))^d \ : \ v = 0 \text{ on } \Gamma \right\}, \quad L_0^2(\Omega) = \left\{ q \in L^2(\Omega) \ : \ \int_\Omega q \, \mathrm{d}x = 0 \right\},$$

and the bilinear form

$$a(u, v) = \int_\Omega (\alpha u \cdot v + \nu \nabla u : \nabla v) \, \mathrm{d}x.$$

Let us introduce the quadratic functional $J : V \to \mathbb{R}$ defined by

$$J(v) = \frac{1}{2} a(v, v) - (f, v)_\Omega,$$

where $(\cdot, \cdot)_\Omega$ denotes the standard L^2 scalar product over Ω. Assuming $\mathrm{mes}(\Gamma) > 0$, the functional J is convex, G-differentiable and coercive on V.

With the above preparations, the Stokes problem (1)–(3) can be formulated as the following constrained minimization problem

Find $u \in V_\sigma = \{ v \in V \mid \nabla \cdot v = 0 \text{ on } \Omega \}$ such that

$$J(u) \leq J(v), \quad \forall v \in V_\sigma. \tag{4}$$

Since V_σ is closed and convex subset of V and the functional J is strongly convex and coercive, the constrained minimization problem (4) has a unique solution. The pressure p is recovered as the Lagrange multiplier associated with the divergence constraint (2).

3 Uzawa Domain Decomposition for Stokes Problem

We now study the domain decomposition of (4). We first rewrite (4) in the following more useful form.

Find $u \in V$ such that:

$$J(u) \leq J(v) \quad \forall v \in V, \tag{5}$$

$$\text{subject to } \nabla \cdot u = 0 \text{ in } \Omega. \tag{6}$$

Let $\{\Omega_1, \ \Omega_2\}$ be a partition of Ω, as shown in Fig. 1, and let

$$\Gamma_{12} = \Gamma_{21} = \partial\Omega_1 \cap \partial\Omega_2, \quad \Gamma_i = \Gamma \cap \partial\Omega_i,$$
$$v_i = v|_{\Omega_i}, \quad p_i = p|_{\Omega_i}, \quad V_i = \left\{ v \in H^1(\Omega_i), \ \ v|_{\Gamma_i} = 0 \right\}.$$

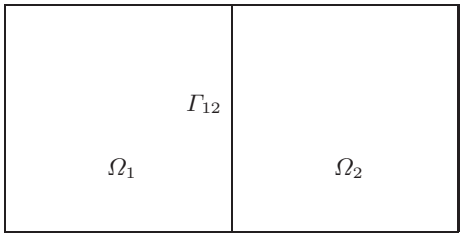


Fig. 1. Decomposition of Ω into two subdomains.

Inner products over Ω_i and Γ_{12} are defined by

$$(u,v)_{\Omega_i} = \int_{\Omega_i} uv\mathrm{d}x, \quad \text{and} \quad (u,v)_{\Gamma_{12}} = \int_{\Gamma_{12}} uv\mathrm{d}\Gamma.$$

Restrictions of the functionals a and J over Ω_i are denoted by a_i and J_i, respectively. To simplify, let us denote vector-valued functions and spaces by bold-face letters, i.e. $\boldsymbol{u} = (u_1, u_2)$, $\mathbf{V} = V_1 \times V_2$, etc.

3.1 Lagrangian Formulation and Dual Problem

Consider the following constrained minimization problem

Find $(u_1, u_2) \in V_1 \times V_2$ such that:

$$J_1(u_1) + J_2(u_2) \leq J_1(v_1) + J_2(v_2) \quad \forall (v_1, v_2) \in V \tag{7}$$

$$\nabla \cdot u_i = 0 \quad \text{in } \Omega_i \tag{8}$$

$$u_1 - u_2 = 0 \quad \text{on } \Gamma_{12}. \tag{9}$$

It is obvious that (7)–(9) is equivalent to (5)–(6). The Lagrangian functional associated with (7)–(9) is

$$\mathcal{L}(\boldsymbol{v}; \boldsymbol{q}, \mu) = \sum_{i=1}^{2} [J_i(v_i) - (q_i, \nabla \cdot v_i)_{\Omega_i}] - (\mu, [\boldsymbol{u}])_{\Gamma_{12}}. \tag{10}$$

where we have set $[\boldsymbol{u}] = (u_1 - u_2)_{|\Gamma_{12}}$. Let us introduce the set

$$\boldsymbol{P} = \left\{ (q_1, q_2) \in L^2(\Omega_1) \times L^2(\Omega_2) \ : \ \int_{\Omega_1} q_1(x)\mathrm{d}x + \int_{\Omega_2} q_2(x)\mathrm{d}x = 0 \right\}.$$

Then the solution of (7)–(9) is characterized by the following saddle-point problem

Find $(\boldsymbol{u}; \boldsymbol{p}, \lambda) \in \boldsymbol{V} \times \boldsymbol{P} \times L^2(\Gamma_{12})$ such that:

$$\mathcal{L}(\boldsymbol{u}; \boldsymbol{q}, \mu) \leq \mathcal{L}(\boldsymbol{v}; \boldsymbol{q}, \mu) \leq \mathcal{L}(\boldsymbol{v}; \boldsymbol{p}, \lambda) \quad \forall (\boldsymbol{v}; \boldsymbol{p}, \lambda) \in \boldsymbol{V} \times \boldsymbol{P} \times L^2(\Gamma_{12}). \tag{11}$$

Solving (11) is equivalent to solving the saddle point equations

$$a_i(u_i, v_i) = (f_i, v_i)_{\Omega_i} + (p_i, \nabla \cdot v_i)_{\Omega_i} + (-1)^{i+1}(\lambda, v_i) \quad \forall v_i \in V^i, \tag{12}$$

$$-(q_i, \nabla \cdot u_i)_{\Omega_i} = 0 \quad \forall q_i \in L^2(\Omega_i) \tag{13}$$

$$(\mu, [\boldsymbol{u}])_{\Gamma_{12}} = 0 \quad \forall \mu \in L^2(\Gamma_{12}). \tag{14}$$

The main advantage of the saddle point formulation is that (12) reduces to $2d$ uncoupled scalar Poisson problems if p_i and λ are known.

Suppose that $u_i = u_i(p_i, \lambda)$ is the solution of the Poisson equation (12). For convenience, in the sequel, we suppress the dependence of u_i on (p_i, λ). setting $v_i = u_i$ in (12) and substituting in (10) we get

$$J^*(\boldsymbol{p}, \lambda) := \mathcal{L}(\boldsymbol{u}(\boldsymbol{p}); \boldsymbol{p}, \lambda) = -\frac{1}{2}\sum_{i=1}^{2} a_i(u_i, u_i). \tag{15}$$

Since the mapping $(\boldsymbol{p}, \lambda) \mapsto \boldsymbol{u}(\boldsymbol{p}, \lambda)$ is linear and the bilinear forms a_i are strongly convex, we deduce that J^* is a strictly concave functional. The dual problem of (7)-(9) is the maximization problem

Find $(\boldsymbol{p}, \lambda) \in \boldsymbol{P} \times L^2(\Gamma_{12})$ such that:

$$J^*(\boldsymbol{p}, \lambda) \geq J^*(\boldsymbol{q}, \mu) \quad \forall (\boldsymbol{q}, \mu) \in \boldsymbol{P} \times L^2(\Gamma_{12}). \tag{16}$$

To derive a maximization method for (16), we need some differential informations on J^*.

3.2 Sensitivity Analysis

The sensitivity problem (which measures the variation of u_i vs. (p_i, λ)) is

$$\tilde{u}_i \in V_i;\ a_i(\tilde{u}_i, v_i) = (\tilde{p}_i, \nabla \cdot v_i)_{\Omega_i} + (-1)^{i+1}(\tilde{\lambda}, v_i)_{\Gamma_{12}} \quad \forall v_i \in V_i \tag{17}$$

so that $u_i(p_i + t\tilde{p}_i, \lambda + t\tilde{\lambda}) = u_i(p_i, \lambda) + t\tilde{u}_i$. The directional derivative of J^* is given by

$$\frac{\partial J^*(\boldsymbol{p}, \lambda)}{\partial(\boldsymbol{p}, \lambda)} \cdot (\tilde{\boldsymbol{p}}, \tilde{\lambda}) = -\sum_{i=1}^{2} a_i(u_i, \tilde{u}_i), \quad \forall (\tilde{\boldsymbol{p}}, \tilde{\lambda}), \tag{18}$$

where $\tilde{u}_i$ is the solution of the sensitivity problem (17). If we set $v_i = u_i$ in (17) and substitute the result into (18), we get

$$\frac{\partial J^*(\boldsymbol{p}, \lambda)}{\partial(\boldsymbol{p}, \lambda)} \cdot (\tilde{\boldsymbol{p}}, \tilde{\lambda}) = -\sum_{i=1}^{2} (\tilde{p}_i, \nabla \cdot u_i)_{\Omega_i} + (\tilde{\lambda}, [\boldsymbol{u}])_{\Gamma_{12}}. \tag{19}$$

Setting $g_i = \nabla_{p_i} J^*$ and $\gamma = \nabla_{\lambda} J^*$, the gradient of J^* with respect to p_i and λ, respectively, we deduce from (19) that

$$g_i = -\nabla \cdot u_i, \quad i = 1, 2, \tag{20}$$

$$\gamma = [\boldsymbol{u}], \tag{21}$$

for the standard L^2 scalar product.

Let $(\tilde{\boldsymbol{p}}, \tilde{\lambda})$ be a search direction for J^*. Since J^* is quadratic and concave, the best search direction is a conjugate gradient direction. At each iteration k, the conjugate gradient direction $(\tilde{\boldsymbol{p}}^k, \tilde{\lambda}^k)$ is given by

$$\beta_k = \left[\|\boldsymbol{g}^k\|^2_{L^2(\Omega)} + \|\gamma^k\|^2_{L^2(\Gamma_{12})}\right]\left[\|\boldsymbol{g}^{k-1}\|^2_{L^2(\Omega)} + \|\gamma^{k-1}\|^2_{L^2(\Gamma_{12})}\right]^{-1}, \quad (22)$$

$$\tilde{\boldsymbol{p}}^k = \boldsymbol{g}^k + \beta_k \tilde{\boldsymbol{p}}^{k-1}, \quad \tilde{\lambda}^k = \gamma^k + \beta_k \tilde{\lambda}^{k-1}. \quad (23)$$

We need to determine the optimal step size to complete the iteration. The optimal step size is computed as the maximizer of the real-valued function $\rho(t) = J^*(\boldsymbol{p} + t\tilde{\boldsymbol{p}}, \lambda + t\tilde{\lambda})$. Since J^* is quadratic and strictly concave, the maximizer of ρ is the unique solution of the linear equation

$$\rho'(t) = \frac{\partial J^*(\boldsymbol{p} + t\tilde{\boldsymbol{p}}, \lambda + t\lambda)}{\partial(\boldsymbol{p}, \lambda)} \cdot (\tilde{\boldsymbol{p}}, \tilde{\lambda}) = 0.$$

We deduce, from (19), that the optimal step size is

$$t = \frac{(\tilde{\lambda}, [\boldsymbol{u}])_{\Gamma_{12}} - \sum_i (\tilde{p}_i, \nabla \cdot u_i)_{\Omega_i}}{(\tilde{\lambda}, [\tilde{\boldsymbol{u}}])_{\Gamma_{12}} - \sum_i (\tilde{p}_i, \nabla \cdot \tilde{u}_i)_{\Omega_i}}. \quad (24)$$

3.3 Uzawa Conjugate Gradient Domain Decomposition Algorithm

With the preparations described in the previous subsection, we can now present our Uzawa/conjugate gradient domain decomposition algorithm for the Stokes problem.

Algorithm DDM/P

Iteration $k = 0$. $\boldsymbol{p}^0 \in \boldsymbol{P}$ and $\lambda^0 \in L^2(\Gamma_{12})$ given

Compute $u_i^0 \in V_D^i$ via

$$a_i(u_i^0, v_i) = (f_i, v_i)_{\Omega_i} + (p_i^0, \nabla \cdot v_i)_{\Omega_i} + (-1)^{i+1}(\lambda^0, v_i)_{\Gamma_{12}}, \quad \forall v_i \in V^i \quad (25)$$

Compute $\boldsymbol{g}^0 \in \boldsymbol{P}$ via

$$(g_i^0, q_i)_{\Omega_i} = -(\nabla \cdot u_i^0, q_i)_{\Omega_i} \quad \forall q_i \in L^2(\Omega_i),\ i = 1, 2.$$

Compute $\gamma^0 \in L^2(\Gamma_{12})$ via

$$(\gamma^0, \mu)_{\Gamma_{12}} = ([\boldsymbol{u}^0], \mu)_{\Gamma_{12}} \quad \forall \mu \in L^2(\Gamma_{12})$$

Initial direction: $\tilde{\boldsymbol{p}}^0 = \boldsymbol{g}^0$, $\tilde{\lambda}^0 = \gamma^0$

Iteration $k \geq 0$. Assuming $\boldsymbol{p}^k$, λ^k, $\tilde{\boldsymbol{p}}^k$, $\tilde{\lambda}^k$, $\boldsymbol{g}^k$, γ^k and $\boldsymbol{u}^k$ are known

Sensitivity problems: compute $\tilde{u}_i \in V^i$ via

$$a_i(\tilde{u}_i^k, v_i) = (\tilde{p}_i^k, \nabla \cdot v_i)_{\Omega_i} + (-1)^{i+1}(\tilde{\lambda}^k, v_i)_{\Gamma_{12}} \quad \forall v_i$$

Step size: Compute t_k using (24)

Update:

$$\lambda^{k+1} = \lambda^k + t_k\tilde{\lambda}^k,$$
$$p_i^{k+1} = p_i^k + t_k\tilde{p}_i^k, \quad u_i^{k+1} = u_i^k + t_k\tilde{u}_i^k,\ i = 1, 2.$$

Gradient: Solve the gradient systems

$$(g_i^{k+1}, q_i)_{\Omega_i} = -(\nabla \cdot u_i^{k+1}, q_i)_{\Omega_i} \quad \forall q_i \in L^2(\Omega_i),\ i = 1, 2. \tag{26}$$
$$(\gamma^{k+1}, \mu)_{\Gamma_{12}} = ([\boldsymbol{u}^{k+1}], \mu)_{\Gamma_{12}} \quad \forall \mu \in L^2(\Gamma_{12}). \tag{27}$$

Conjugate gradient direction: Compute $(\tilde{\boldsymbol{p}}^{k+1}, \tilde{\lambda}^{k+1})$ using (22)–(23)

We iterate until the gradient is "sufficiently" small, i.e.

$$\left\|\boldsymbol{g}^k\right\|^2_{L^2(\Omega)} + \left\|\gamma^k\right\|^2_{L^2(\Gamma_{12})} < \varepsilon\left(\left\|\boldsymbol{g}^0\right\|^2_{L^2(\Omega)} + \left\|\gamma^0\right\|^2_{L^2(\Gamma_{12})}\right) \tag{28}$$

where $\varepsilon > 0$ is a given tolerance. Each iteration, we solve in parallel $2d$ scalar Poisson problems. The parallelizability of the algorithm is therefore obvious.

Note that if $\boldsymbol{p}^0 = (p_1^0, p_2^0) \in \boldsymbol{P}$ and $\boldsymbol{g}^k = (g_1^k, g_2^k) \in \boldsymbol{P}$ for all $k \geq 0$, then $\boldsymbol{p}^k = (p_1^k, p_2^k) \in \boldsymbol{P}$ for all $k \geq 1$. To compute $\boldsymbol{g}^k$ in $\boldsymbol{P}$, we first solve the gradient systems (26) separately to obtain $\bar{g}_i^k$. The gradient $\boldsymbol{g}^k = (g_1^k, g_1^k)$ is then obtained by the projection of $(\bar{g}_1^k, \bar{g}_2^k)$ onto $\boldsymbol{P}$.

4 Numerical Experiments

The domain decomposition algorithms presented in the previous sections were implemented in Fortran 90, on a 172-core Linux cluster, using an MPI library. MPI subroutines are used only for solving in parallel uncoupled Poisson problems. All linear systems involved are solved by a preconditioned conjugate gradient algorithm. The preconditioning is obtained by an incomplete Choleski factorization with drop tolerance varying from 10^{-5} to 10^{-3}. For discrete velocity-pressure spaces, we use the P^1-iso-P^2/P^1 element. These spaces are well known to satisfy the discrete Babuska-Brezzi inf-sup condition.

We consider the domain $\Omega = (0,\ a) \times (0,\ 1)$ and we take $\alpha = 0$ and $\nu = 1$ in (1). The right-hand side in (1) is adjusted such that the exact solution is

$$u_1(x, y) = (1 - \cos(2\pi x/a))\sin(2\pi y),\ \ u_2(x, y) = \frac{1}{a}(\cos(2\pi y) - 1)\sin(2\pi x/a)$$
$$p(x, y) = \frac{2\pi}{a}(cos(2\pi y) - \cos(2\pi x/a))$$

In our numerical experiments, $a = 10$.

We first consider the classical Stokes problem (i.e. $\alpha = 0$ and $\nu = 1$). Table 1 shows the L^2 and H^1 errors for u_h and p_h ($h = 1/256$) obtained using Algorithm DDM/P with decomposition into 2, 4, 8, 16, 32 and 64 subdomains. We notice that

the errors are comparable with those obtained with the standard Uzawa/conjugate gradient for Stokes problem ($N_{SD} = 1$). Note that for $h = 1/256$, we have $2561 \times 257 = 658,177$ nodes in the fine (velocity) mesh. Figure 2 shows that the speed-up is significant for $N_{SD} \geq 8$.

Table 1. H^1 and L^2 errors for various number of subdomains N_{SD} for Algorithm DDM/P with $h = 1/256$.

N_{SD}	Iterations	$\|u - u_h\|_{L^2(\Omega)}$	$\|u - u_h\|_{H^1(\Omega)}$	$\|p - p_h\|_{L^2(\Omega)}$
1	20	3.7602×10^{-4}	1.2293×10^{-1}	7.1937×10^{-4}
2	110	3.7611×10^{-4}	1.2295×10^{-1}	7.2209×10^{-4}
4	120	3.7608×10^{-4}	1.2295×10^{-1}	7.5186×10^{-4}
8	131	3.7616×10^{-4}	1.2295×10^{-1}	7.8013×10^{-4}
16	158	3.7610×10^{-4}	1.2295×10^{-1}	8.2989×10^{-4}
32	205	3.7611×10^{-4}	1.2295×10^{-1}	9.4120×10^{-4}
64	258	3.7592×10^{-4}	1.2295×10^{-1}	1.1080×10^{-3}

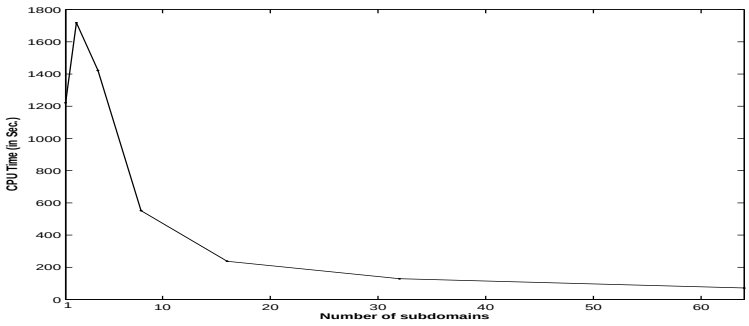


Fig. 2. CPU times vs. number of subdomains for Algorithm DDM/P with $\alpha = 0$, $\nu = 1$ and $h = 1/256$.

We now consider the case $\alpha = 1$ and $\nu = 10^{-3}$. For the one-subdomain case, we use the Cahouet-Chabard preconditioner [1], see also [2, 3]. This preconditioner is the best for the Uzawa/conjugate gradient algorithm for the generalized Stokes problem with $\alpha/\nu \gg 1$. Figure 3 shows the CPU times obtained with our domain

decomposition algorithm. We notice that the speed-up is significant when $N_{SD} \geq 16$.

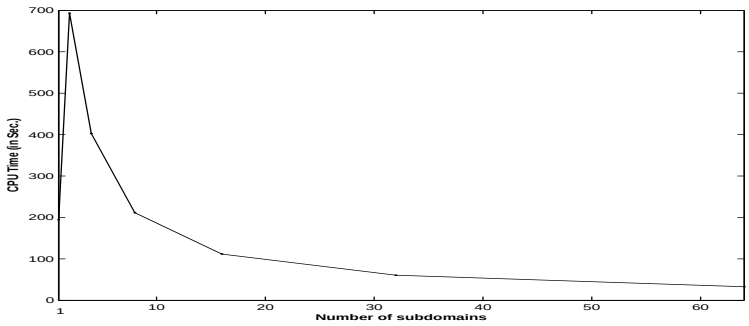


Fig. 3. CPU times vs. number of subdomains for Algorithm DDM/P with $\alpha = 1$, $\nu = 10^{-3}$ and $h = 1/256$.

5 Conclusion

A Uzawa domain decomposition algorithm for Stokes problem has been introduced. The standard L^2-scalar product is used for computing the gradient (26)–(27). This approach is easy to implement and has a significant speed-up when the number of subdomains is larger then 10. Nevertheless, it leads to a h-dependent algorithm. To improve this algorithm, different preconditioners will be investigated :

- the Steklov-Poincaré operator on the interface (see e.g. [4, 5]),
- the Cahouet-Chabard preconditioner [1] for the pressure multiplier.

A coarse component of the preconditioner will be studied. Indeed, the iteration counts for large N_{SD} could indicate the lack of a coarse component in the preconditioner.

References

1. J. Cahouet and J.P. Chabard Some fast 3-D solvers for the generalized Stokes problem. *Int. J. Numer. Methods Fluids*, 8:269–295, 1988.
2. R. Glowinski Numerical methods for fluids (part 3). In Ciarlet P.G. and Lions J.L., editors, *Numerical Methods for Fluids (Part 3)*, volume IX of *Handbook of Numerical Analysis*, pp. 3–1074. North-Holland, Amsterdam, 2003.
3. R. Glowinski and P. Le Tallec *Augmented Lagrangian and Operator-splitting Methods in Nonlinear Mechanics*. SIAM, Philadelphia, PA, 1989.
4. J. Koko Convergence analysis of optimization-based domain decomposition methods for a bonded structure. *Appl. Numer. Math.*, 58:69–87, 2008.
5. P. Le Tallec and T. Sassi Domain decomposition with nonmatching grids: augmented Lagrangian approach. *Math. Comput.*, 64:1367–1396, 1995.

A Domain Decomposition Method Combining a Boundary Element Method with a Meshless Local Petrov-Galerkin Method

Li Maojun and Zhu Jialin

College of Mathematics and Statistics, Chongqing University, Chongqing 400044, PR China, limaojun216@163.com

Summary. A non-overlapping domain decomposition algorithm combining boundary element method with meshless local Petrov-Galerkin method is presented for solving the boundary value problem with discontinuous coefficient in this paper. The static relaxation parameter is employed to speed up the convergence rate. The convergence range and the optimal value of static relaxation parameter are studied, but the numerical results show that the optimal static relaxation parameter is different for different problems. Therefore, a dynamic relaxation parameter is presented for the algorithm. The numerical results show that the number of iteration with the dynamic relaxation parameter is less than that with the static relaxation parameter.

Key words: boundary element method, meshless local Petrov–Galerkin method, domain decomposition method, relaxation parameter

1 Introduction

As we know, the meshless methods and boundary element method (BEM) are widely employed as two of the main numerical methods for the solution of a wide variety of science and engineering problems. However, they exhibit different advantages when applied to different classes of problems. The main feature of the meshless methods is the absence of an explicit mesh, and the approximate solutions are constructed entirely based on a cluster of scattered nodes. Therefore, the meshless methods are well suited to problems with extremely large deformation, dynamic fracturing, or explosion [1, 2]. On the other hand, the main advantage of BEM is that the dimensionality of the problem is reduced by one, and the BEM is very efficient for the analysis of homogenous linear problems in unbounded domains.

It is attractive to divide a computational domain into sub-domains and to use the most appropriate method for each sub-domain. The idea of coupling the meshless methods and BEM is by now well known as an efficient analysis tool, which makes use of their advantages. A great number of articles on the topic, such as combining element-free Galerkin method (EFGM) with BEM [10, 11, 12, 21],

reproducing kernel particle method (RKPM) with BEM [16], the mesh-free finite cloud method (MFCM) with BEM [17], meshless local Petrov-Galerkin (MLPG) method with BEM [9], can be found. The above coupling methods deduce an entire unified system for the whole domain, by combining the discretized equations for the BEM and different meshless methods in sub-domains. However, the algorithm for constructing a large entire system for the whole domain is complicated and time-consuming for computation when compared with that for each single equation, and may destroy the desirable features originally existing in the meshless methods matrices, namely, symmetry and sparsity.

The domain decomposition methods (DDM) combining FEM-BEM or BEM-BEM have been developed [3, 4, 5, 6, 7, 8, 13, 14, 15, 18]. The DDM is better than the above coupling methods when the domains under consideration are governed by different differential equations or constructed of different materials, especially in the case of large domain with complicated boundary manifold. Therefore, a non-overlapping DDM combining BEM with MLPG method is presented in order to make use of their advantages and preserve the nature of the both methods in this paper.

This paper is arranged as follows: Sect. 2 gives a non-overlapping domain decomposition algorithm combining BEM and MLPG method for solving the boundary value problem with discontinuous coefficient. Then, the dynamic relaxation parameter is presented for the algorithm in the next section. In Sect. 4, the convergence range and the optimal value of the static relaxation parameter are studied and the validity of the dynamic relaxation parameter is verified by numerical results. Finally, the conclusions are given in Sect. 5.

2 A DDM Combining BEM with the MLPG Method

Consider the following boundary value problem with discontinuous coefficient

$$\begin{cases} \nabla \cdot (\gamma(x) \nabla u(x)) = 0, \ x \in \Omega \\ u(x) = f(x), \ x \in \Gamma_u \\ q(x) = \gamma(x) \cdot \partial u(x)/\partial n = g(x), \ x \in \Gamma_q \end{cases} \tag{1}$$

where $\Omega \subset \mathrm{R}^2$, Γ is its boundary. $x = (x_1, x_2)$ denotes the point in R^2. $\gamma(x)$ is the conductivity coefficient. $f(x), g(x)$ are the given boundary data. The problem (1) often appears in engineering problems, e.g., heat conduction and electric conduction models with mixed materials, Darcy flow in porous media, etc. Many methods, such as FEM [19] and BEM [20], have been successfully used to solving the problem (1), however, the domain decomposition is suitable for the problems with discontinuity conductivity coefficients. In this paper, we assume that the conductivity coefficient is as follows

$$\gamma(x) = \begin{cases} 1, \ x \in \Omega_B \subset \Omega \\ \gamma_M(x), x \in \Omega_M = \Omega \setminus \Omega_B \end{cases} \tag{2}$$

Then the domain of the original problem can be decomposed into BEM sub-domain Ω_B and MLPG sub-domain Ω_M, let $\Gamma^I = \partial\Omega_B \cap \partial\Omega_M$ be the BEM/MLPG interface (Fig.1). Apparently, the continuity and equilibrium conditions should be satisfied at the interface, that is

$$u_B(x) = u_M(x), \quad \frac{\partial u_B(x)}{\partial n_B} + \gamma_M(x)\frac{\partial u_M(x)}{\partial n_M} = 0,\ x \in \Gamma^I \tag{3}$$

where $u_B(x) = u(x)|_{\Omega_B}$, $u_M(x) = u(x)|_{\Omega_M}$, n_B and n_M are the unit outward normal vectors for the BEM and MLPG sub-domains, respectively.

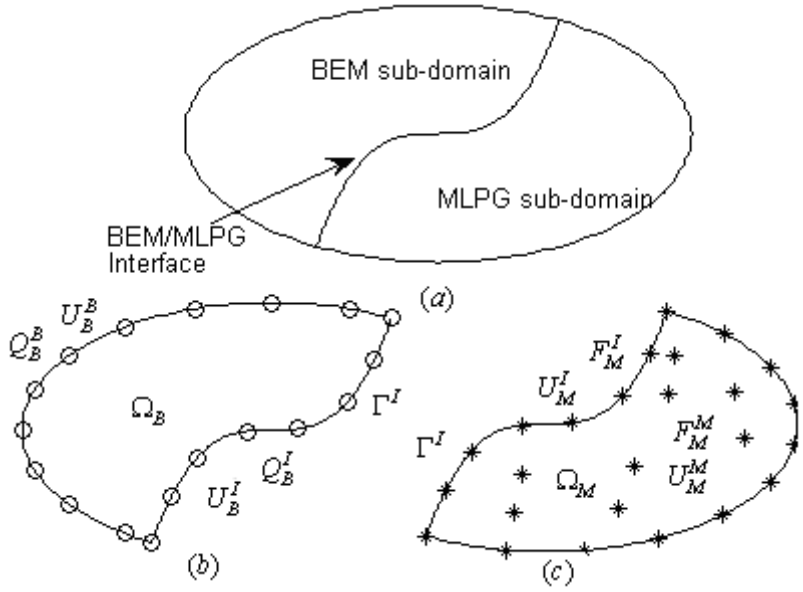


Fig. 1. Domain decomposed into BEM and MLPG sub-domains.

On the one hand, we can obtain the following boundary integral equation for the BEM sub-domain Ω_B

$$c(y)u_B(y) + \int_{\partial\Omega_B} u_B(x)\frac{\partial u^*(x,y)}{\partial n_B}\mathrm{d}\Gamma = \int_{\partial\Omega_B} u^*(x,y)\frac{\partial u_B(x)}{\partial n_B}\mathrm{d}\Gamma,\ y \in \partial\Omega_B \tag{4}$$

where $u^*(x,y) = -\frac{1}{2\pi}\ln|x-y|$ is the fundamental solution of Laplace equation, $c(y)$ depends on the geometry shape at point y. The boundary integral equation (4) can be rewritten as the following matrix form

$$\begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix} \begin{Bmatrix} U_B^B \\ U_B^I \end{Bmatrix} = \begin{pmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{pmatrix} \begin{Bmatrix} Q_B^B \\ Q_B^I \end{Bmatrix} \tag{5}$$

where U_B^B and Q_B^B are column vectors containing the non-interface nodal potentials and fluxes values, respectively, U_B^I and Q_B^I are column vectors containing the interface nodal potentials and fluxes values, respectively. H and G are the corresponding coefficient matrices.

On the other hand, we have the following local weak form for the MLPG sub-domain Ω_M by means of the MLPG method

$$\begin{aligned}&\int_{\Omega_{Mi}} (\gamma_M \cdot \nabla u \cdot \nabla v)\,\mathrm{d}\Omega + \int_{\Gamma_{ui}} \left(\alpha uv - \gamma_M \cdot \tfrac{\partial u}{\partial n_M} \cdot v\right)\mathrm{d}\Gamma \\ &=\alpha \int_{\Gamma_{ui}} fv\mathrm{d}\Gamma + \int_{\Gamma_{qi}} gv\mathrm{d}\Gamma + \int_{\Gamma_{Ii}} \left(\gamma_M \cdot \tfrac{\partial u}{\partial n_M}\right) v\mathrm{d}\Gamma\end{aligned} \tag{6}$$

where $\Omega_{Mi} \subset \Omega_M$ is a local sub-domain, v is a weight function, α is a penalty factor, $\Gamma_{Ii} = \partial\Omega_{Mi} \cap \Gamma^I$, $\Gamma_{ui} = \partial\Omega_{Mi} \cap \Gamma_u$, $\Gamma_{qi} = \partial\Omega_{Mi} \cap \Gamma_q$. Then the assembled linear equations are given by

$$\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \begin{Bmatrix} U_M^M \\ U_M^I \end{Bmatrix} = \begin{Bmatrix} F_M^M \\ F_M^I \end{Bmatrix} \tag{7}$$

where U_M^I is the interface nodal potentials vectors, U_M^M is the all nodal potentials vectors except the interface nodal potentials, K and F are the corresponding coefficient matrix and right side vector, respectively. Note that F_M^I is a vector containing $\int_{\Gamma_{Ii}} \left(\gamma_M \cdot \frac{\partial u}{\partial n_M}\right) v\mathrm{d}\Gamma$, then the conditions (3) can be rewritten as

$$U_B^I = U_M^I, \quad F_M^I = PQ_M^I = -PQ_B^I, \tag{8}$$

where P is a transition matrix, which depends on the weight function v and the shape function in the moving least square approximation, Q_M^I is a column vector containing the interface nodal fluxes values.

Therefore, a parallel Dirichlet-Neumann BEM-MLPG algorithm is as follows:

Step 1: assign the initial potential vector $U_{B,0}^I$ and flux vector $Q_{M,0}^I$, and n:=0.

Step 2: solve

$$\begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix} \begin{Bmatrix} U_{B,n}^B \\ U_{B,n}^I \end{Bmatrix} = \begin{pmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{pmatrix} \begin{Bmatrix} Q_{B,n}^B \\ Q_{B,n}^I \end{Bmatrix} \text{ for } Q_{B,n}^I \tag{9}$$

$$\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \begin{Bmatrix} U_{M,n}^M \\ U_{M,n}^I \end{Bmatrix} = \begin{Bmatrix} F_{M,n}^M \\ PQ_{M,n}^I \end{Bmatrix} \text{ for } U_{M,n}^I \tag{10}$$

Step 3: apply

$$U_{B,n+1}^I = \beta U_{M,n}^I + (1-\beta) U_{B,n}^I \tag{11}$$

$$Q_{M,n+1}^I = -Q_{B,n}^I \tag{12}$$

Step 4: check if $\left\|U_{B,n+1}^I - U_{B,n}^I\right\| + \left\|Q_{M,n+1}^I - Q_{M,n}^I\right\| \le \varepsilon \max(||U_{B,n+1}^I||, ||Q_{M,n+1}^I||)$, if yes then stop, otherwise set n:=n+1, and go to Step 2.

here β is a relaxation parameter to ensure and/or accelerate convergence, ε is the user specified error allowance.

3 A Dynamic Relaxation Parameter

If the relaxation parameter β is assigned as a constant for every iteration, an optimal $\bar{\beta}$ can be obtained by testing with different values. However, we find that the optimal static value is different for the different problems in numerical test, therefore we couldn't find a suitable optimal value for all problems. Fortunately, a dynamic relaxation parameter has been obtained for the sequential FEM-BEM algorithm [18], that is to say, the iterative procedure can be facilitated by allowing the relaxation parameter to change dynamically with each iteration. In this section, a dynamic relaxation parameter will be presented for the parallel Dirichlet-Neumann BEM-MLPG iterative algorithm.

By minimizing the square error functional

$$G(\beta) = \left\| U^I_{B,n+1}(\beta) - U^I_{B,n}(\beta) \right\|^2 + \left\| Q^I_{M,n+1}(\beta) - Q^I_{M,n}(\beta) \right\|^2 \tag{13}$$

with respect to the relaxation parameter β, one gets an optimal dynamic value for the next iteration, i.e.,

$$\beta_n = \frac{\langle e_{B,n}, e_{B,n} - e_{M,n} \rangle + \langle e_{B,n-1}, e_{B,n-1} - e_{M,n-1} \rangle}{\left\| e_{B,n} - e_{M,n} \right\|^2 + \left\| e_{B,n-1} - e_{M,n-1} \right\|^2}, \; n \geq 3 \tag{14}$$

$$e_{B,n} = U^I_{B,n} - U^I_{B,n-1}, \quad e_{M,n} = U^I_{M,n} - U^I_{M,n-1}, \; n \geq 2 \tag{15}$$

where $\langle a, b \rangle$ is a inner product, and $\|a\|^2 = \langle a, a \rangle$.

4 Numerical Examples

To illustrate the convergence results of the iterative algorithm, a numerical example is presented in this section. Moreover, the accelerated convergence of the dynamic relaxation parameter will also be shown. In this section, we choose the error bound as $\varepsilon = 10^{-4}$.

Example (Potential flow problem [6, 8]) We consider the mixed boundary value problem (1), and assume that $\Omega_B = [0,1] \times [0,1]$ and $\Omega_M = [1,2] \times [0,1]$, the conductivity coefficient $\gamma_M(x) = 2$, the boundary conditions are selected such that $u(0, x_2) = 0$, $u(2, x_2) = 200$ and zero flux elsewhere (Fig. 2a).

Using the proposed iterative algorithm in Sect. 2, the problem is solved by three different discretization types denoted as Fig. 2 (b–d).

Figure 3 shows the convergence ranges and the optimal static values with the static relaxation parameter for the different discretization types. Beyond the values shown in Fig. 3, the iterative algorithm will not converge. From Fig. 3, we know that convergence ranges and optimal static values are different for the different discretization types. Therefore it is impossible to select a suitable optimal static value for all cases.

Table 1 shows the numbers of iterations with optimal static values and dynamic values for the different discretization types. From Table 1, obviously, the number of iterations with dynamic value is less than or equal to that with the static value. Therefore, we can say that the dynamic values is the optimal relaxation parameter.

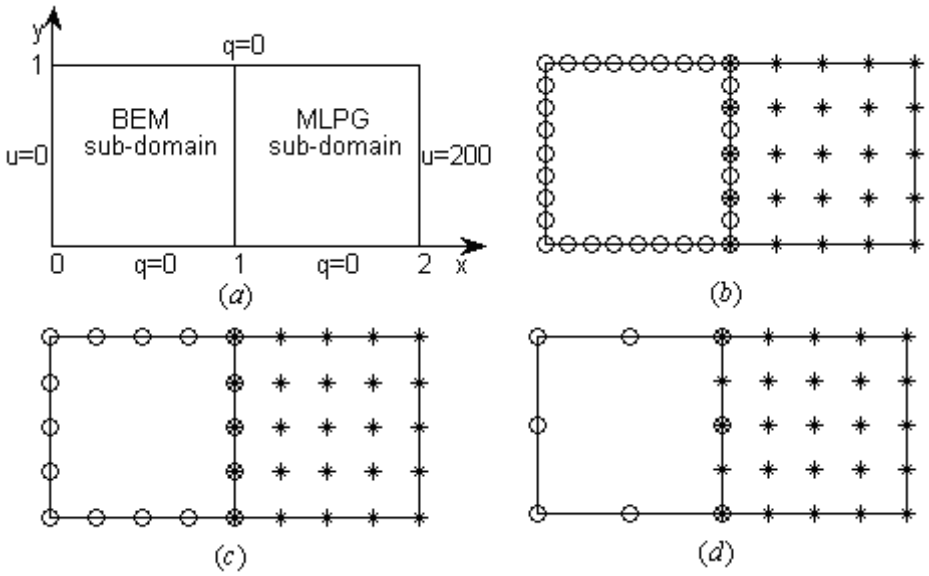


Fig. 2. Potential flow problem and discretization types.

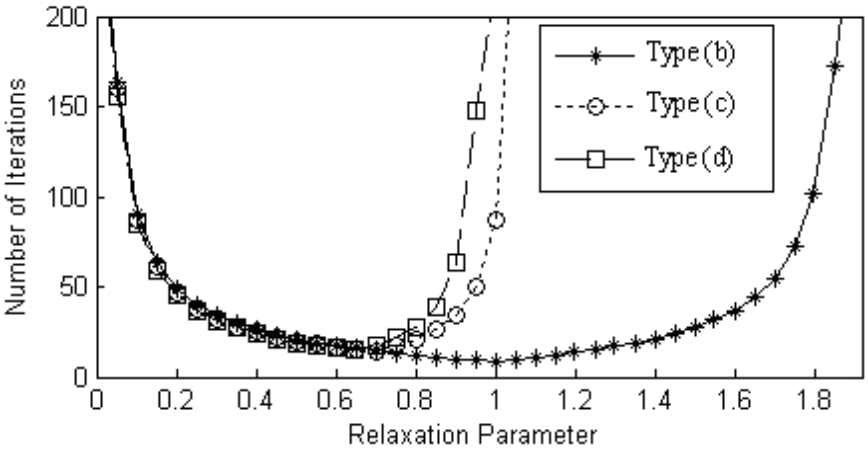


Fig. 3. Convergence ranges and optimal static values for the different discretization types.

Table 1. Number of iterations for the different discretization types.

Relaxation parameter	Type (b)	Type (c)	Type (d)
Optimal static value β	8	14	16
Dynamic values β_n	8	11	9

5 Conclusions

Generally speaking, the static relaxation parameter is always employed to speed up the convergence rate of domain decomposition methods. However the convergence ranges and optimal values of the static relaxation parameter are different for different problems. Therefore, the dynamic relaxation parameter is used in the proposed domain decomposition algorithm in this paper, the numerical results show that the dynamic relaxation parameter is the optimal relaxation parameter which is well suited to all cases.

References

1. G. Beer. An efficient numerical method for modeling initiation and propagation of cracks along material interfaces. *Int. J. Numer. Methods Eng.*, 36:3579–3595, 1993.
2. T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. Meshless method:an overview and recent developments. *Comput. Methods Appl. Mech. Engrg.*, 139:3–47, 1996.
3. C. Carstensen, M. Kuhn, and U. Langer. Fast parallel solvers for symmetric boundary element domain decomposition equations. *Numer. Math.*, 79:321–347, 1998.
4. M. Costabel. Symmetric methods for the coupling of finite elements and boundary elements. In C.A. Brebbia, W.L. Wendland, and G. Kuhn, editors, *Boundary Elements IX*, pp. 411–420. Springer, Berlin, Heidelberg, New York, NY, 1987.
5. W. M. Elleithy and H. J. Al-Gahtani. An overlapping domain decomposition approach for coupling the finite and boundary element methods. *Eng. Anal. Bound. Elem.*, 24(5): 391–398, 2000.
6. W. M. Elleithy, H. J. Al-Gahtani, and M. El-Gebeily. Convergence of the iterative coupling of bem and fem. *In twenty-first World Conference on the Boundary Element Method, BEM21,Oxford University*, pp. 281–290, 1999.
7. W. M. Elleithy, H. J. Al-Gahtani, and M. El-Gebeily. Iterative coupling of be and fe methods in elastostatics. *Eng. Anal. Bound. Elem.*, 258:685–695, 2001.
8. W. M. Elleithy and M. Tanaka. Interface relaxation algorithms for bem-bem coupling and fem-bem coupling. *Comput. Methods Appl. Mech. Eng.*, 192(26–27):2977–2992, 2003.
9. J. Euripides and D. P. Sellountos. A mlpg(lbie) approach in combination with bem. *Comput. Methods Appl. Mech. Eng.*, 194:859–875, 2005.
10. Y. T. Gu and G. R. Liu. Coupling of element free galerkin and hybrid boundary element methods using modified variational formulation. *Comput. Mech.*, 26:166–173, 2000.
11. Y. T. Gu and G. R. Liu. A coupled element-free galerkin/boundary element method for stress analysis of two-dimension so1id. *Comput. Methods Appl. Mech. Eng.*, 190:4405–4419, 2001.
12. Y.T. Gu and G.R. Liu. Meshless methods coupled with other numerical methods. *Tsinghua Sci. Technol.*, 10(1):8–15, 2005.
13. G.C. Hsiao and W. L. Wendland. Domain decomposition in boundary element methods. In R. Glowinski, Y.A. Kuznetsov, G. Meurant, J. Periaux, and O.B. Widlund, editors, *Proceedings of the Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations, Moscow, May 21–25, 1990*, pp. 41–49. SIAM, Philadelphia, PA, 1991.

14. N. Kamiya, H. Iwase, and E. Kita. Parallel implementation of boundary element method with domain decomposition. *Eng. Anal. Bound. Elem.*, 18:209–216, 1997.
15. U. Langer. Parallel iterative solution of symmetric coupled FE/BE–equation via domain decomposition. In A. Quarteroni, J. Periaux, Y.A. Kuznetsov, and O.B. Widlund, editors, *Sixth International Conference on Domain Decomposition Methods in Science and Engineering, Como, June 15–19, 1992*, volume 157 of *Contempory Mathematics*, pp. 335–344. AMS, Providence, RI, 1994.
16. C.K. Lee, S.T. Lie, and Y.Y. Shuai. On coupling of reproducing kernel particle method and boundary element method. *Comput. Mech.*, 34:282–297, 2004.
17. G. Li, G.H. Paulino, and N.R. Aluru. Coupling of the mesh-free finite cloud method with the boundary element method a collocation approach. *Comput. Methods Appl. Mech. Eng.*, 192:2355–2375, 2003.
18. C.C. Lin, E.C. Lawton, J.A. Caliendo, and L.R. Anderson. An iterative finite element-boundary element algorithm. *Comput. Struct.*, 59:899–909, 1996.
19. H. Peter, L. Carbo, P. Haria, and S. Giancarlo. A Lagrange multiplier method for the finite element solution of elliptic interface problems using non-matching meshes. *Numer. Math.*, 100:91–115, 2005.
20. H. Yang, X. Yang, and Y. Wang. A collocation method for the conductivity problem with discontinuous coefficient. *Numer. Math. J. Chin. Univ.*, 14(2):157–170, 2005.
21. Z. Zhang, K.M. Liew, and Y. Cheng. Coupling of the improved element-free galerkin and boundary element methods for two-dimensional elasticity problems. *Eng. Anal. Bound. Elem.*, 32:100–107, 2008.

A Domain Decomposition Method Based on Augmented Lagrangian with a Penalty Term in Three Dimensions

Chang-Ock Lee[1] and Eun-Hee Park[2]

[1] Department of Mathematical Sciences, KAIST, Daejeon, 305-701, South Korea, colee@kaist.edu
[2] Center for Computation and Technology, Louisiana State University, Baton Rouge, LA 70803, USA, epark2@cct.lsu.edu

Summary. In our earlier work [4], a dual iterative substructuring method for two dimensional problems was proposed, which is a variant of the FETI-DP method. The proposed method imposes continuity on the interface by not only the pointwise matching condition but also uses a penalty term which measures the jump across the interface. For a large penalization parameter, it was proven that the condition number of the resultant dual problem is bounded by a constant independent of both the subdomain size H and the mesh size h. In this paper, we extend the method to three dimensional problems. For this extension, we consider two things; one is the construction of a penalty term in 3D to give the same convergence speed as in 2D and the other is how to treat the ill-conditioning of the subdomain problems due to a large penalization parameter. To resolve these two key issues, we need to be aware of the difference between 2D and 3D in the geometric complexity of the interface. Based on the geometric observation for the difference, we suggest a modified penalty term and a preconditioner aiming at reducing couplings between functions on the interface.

1 Introduction

We consider the Poisson problem

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{1}$$

where Ω is a bounded polyhedral domain in $\mathbb{R}^3$ and $f \in L^2(\Omega)$.

In our previous work [4], for two dimensional problems, a dual iterative substructuring method was proposed using the augmented Lagrangian method, which is a variant of the FETI-DP method. To the Lagrangian functional of the standard FETI-DP, a penalty term is added, which measures the jump across the interface and includes a positive penalization parameter η. In the same way as in most dual

substructuring approaches, the saddle-point problem related to the augmented Lagrangian functional is reduced to a dual problem with Lagrange multipliers as unknowns. Then it is solved by the conjugate gradient method. Differently from the FETI-DP method, it was proven that the dual problem has a constant condition number independently of H and h even though it is not accompanied by any preconditioner.

In this paper, we extend the method to the three dimensional case. For this extension, there are two things to be considered; one is to construct a strong penalty term in 3D to guarantee the same convergence speed as in 2D and the other is how to treat the ill-conditioning of the subdomain problems due to a large penalization parameter. For both issues, we need to be aware of the difference between 2D and 3D in the geometric complexity of an interface. An interface in 3D includes not only faces but also edges which make all nodes on the interface coupled. First, it is noted that the adoption of the same penalty as for two-dimensional problems gives an algorithm which maintains the same performance with respect to the condition number of the dual problem. However, the penalty term makes an unnecessary coupling between functions on face nodes and edges nodes. Since such a coupling causes a considerable decrease on practical efficiency, we suggest a modified penalty term for the three dimensional problem aiming at reducing the coupling between functions on the interface. Next, unlike the FETI-DP method, subdomain problems containing the penalty term are solved iteratively, of which the condition number becomes large as the penalization parameter η increases. The same type of preconditioner as in 2D might be satisfactory for the ill-conditioned problem due to a large η. But, since the preconditioner suggested in [4] contains a coupling among all nodes on the interface in 3D, it is hardly practical in the implementation. Based on such an observation, a more appropriate preconditioner for three-dimension problems is constructed, which is not only optimal with respect to η but also more practical then the one used in 2D.

2 Dual Iterative Substructuring with a Penalty Term

Let $\mathcal{T}_h$ denote a quasi-uniform triangulation on Ω, where the discretization parameter h stands for the maximal mesh size of $\mathcal{T}_h$. For simplicity, we consider a triangulation of hexahedra and the standard trilinear finite element approximate solution of (1): find $u_h \in X_h$ such that

$$a(u_h, v_h) = (f, v_h) \quad \forall v_h \in X_h, \tag{2}$$

where

$$a(u_h, v_h) = \int_\Omega \nabla u_h \cdot \nabla v_h \, dx, \quad (f, v_h) = \int_\Omega f v_h \, dx,$$

and $X_h = \{v_h \in H_0^1(\Omega) \cap \mathcal{C}^0(\overline{\Omega}) \mid \forall \tau \in \mathcal{T}_h,\ v_h|_\tau \in \mathbb{Q}_1(\tau)\}$.

We decompose Ω into N non-overlapping subdomains $\{\Omega_k\}_{k=1}^N$, where a partition $\{\Omega_k\}_{k=1}^N$ of Ω is assumed to be shape-regular. On each subdomain, the triangulation $\mathcal{T}_{h_k}$ is quasi-uniform with matching grids on the boundaries of neighboring subdomains across the interface Γ. Here the interface Γ is the union of the

common interfaces among all subdomains, i.e., $\Gamma = \bigcup_{k<l} \Gamma_{kl}$, where Γ_{kl} denotes the common interface of two adjacent subdomains Ω_k and Ω_l. We define the finite-dimensional subspace X^k on each subdomain Ω_k by

$$X^k = \left\{ v_h^k \in \mathcal{C}^0(\overline{\Omega}_k) \;\middle|\; \forall \tau \in \mathcal{T}_{h_k},\ v_h^k|_\tau \in \mathbb{Q}_1(\tau),\ v_h^k|_{\partial\Omega \cap \partial\Omega_k} = 0 \right\}.$$

By enforcing the continuity at the corner points, we assemble X^k's into X_h^c:

$$X_h^c = \left\{ v = (v_h^k)_k \in \prod_{k=1}^N X^k \;\middle|\; v \text{ is continuous at each corner} \right\}.$$

The interface Γ is composed of faces which are shared by two subdomains, edges which are shared by more than two subdomains, and vertices. The geometrical objects on the interface are characterized in more details as

(i) $\mathcal{F}_{kl}$ denotes the common face of Ω_k and Ω_l, which is regarded as an open set.
(ii) $\mathcal{E}_m$, where m is an index of an edge, is an edge shared by neighboring subdomains, which does not include its end points, the vertices.

To enforce the continuity on the interface except vertices, a signed Boolean matrix B is introduced in the same way as in FETI-DP (cf. [1, 2]). Note that we do not allow any redundant continuity constraint on any edges, i.e., in the case where an edge $\mathcal{E}_m$ is shared by four subdomains, there are four different ways to choose three pairs of adjacent subdomains to impose the continuity on the edge nodes.

The finite element problem (2) is reformulated as a minimization problem with constraints imposed by the requirement of continuity across the interface Γ:

$$\min_{v \in X_h^c} \left(\frac{1}{2} \sum_{k=1}^N \int_{\Omega_k} |\nabla v|^2 \, dx - (f, v) \right) \quad \text{subject to} \quad Bv = 0.$$

As in the constrained optimization, we introduce a vector μ of Lagrange multipliers in $\mathbb{R}^M$ and define a Lagrangian functional $\mathcal{L} : X_h^c \times \mathbb{R}^M \to \mathbb{R}$ as

$$\mathcal{L}(v, \mu) = \frac{1}{2} \sum_{k=1}^N \int_{\Omega_k} |\nabla v|^2 \, dx - (f, v) + \langle Bv, \mu \rangle,$$

where M represents the number of constraints used for the pointwise matching on the interface and $\langle \cdot, \cdot \rangle$ is the Euclidean inner product in $\mathbb{R}^M$. Then, we slightly change the Lagrangian $\mathcal{L}$ by adding a penalty term. It is natural to adopt the same penalty term as suggested for the two dimensional problem in [4]:

$$J_\eta(u, v) = \sum_{k<l} \frac{\eta}{h} \int_{\Gamma_{kl}} (u^k - u^l)(v^k - v^l) \, ds, \quad \eta > 0. \tag{3}$$

To make the 3D algorithm efficient, we should minimize the coupling between the functions associated with face nodes and edge nodes. But, the penalty term in (3)

makes face nodes and edge nodes in each part Γ_{kl} of Γ coupled so that all nodes on the interface are tied. By considering the interface as a union of two separate object, faces and edges, we introduce a modified penalty term

$$J_\eta(u,v) = \eta(J_{\mathcal{F}}(u,v) + J_{\mathcal{E}}(u,v)), \quad \eta > 0, \tag{4}$$

where

$$J_{\mathcal{F}}(u,v) = \frac{1}{h}\sum_{k<l}\int_{\mathcal{F}_{kl}} (u^k_{\mathcal{F}_{kl}} - u^l_{\mathcal{F}_{kl}})(v^k_{\mathcal{F}_{kl}} - v^l_{\mathcal{F}_{kl}})\,dx$$

and

$$J_{\mathcal{E}}(u,v) = \sum_{\mathcal{E}_m}\sum_{(i,j)\in I_{\mathcal{E}_m}}\int_{\mathcal{E}_m}(u^i - u^j)(v^i - v^j)\,ds.$$

Here, $u^k_{\mathcal{F}_{kl}}$ is a part of u, which is related to the contribution to u^k on $\mathcal{F}_{kl}$ only from the face nodal basis functions excluding the edge nodal basis functions. We define an augmented Lagrangian $\mathcal{L}_\eta$ with the penalty term J_η

$$\mathcal{L}_\eta(v,\mu) = \mathcal{L}(v,\mu) + \frac{1}{2}J_\eta(v,v).$$

Given the augmented Lagrangian $\mathcal{L}_\eta$, we consider the saddle-point problem:

$$\mathcal{L}_\eta(u_h,\lambda_h) = \max_{\mu_h\in\mathbb{R}^M}\min_{v_h\in X_h^c}\mathcal{L}_\eta(v_h,\mu_h) = \min_{v_h\in X_h^c}\max_{\mu_h\in\mathbb{R}^M}\mathcal{L}_\eta(v_h,\mu_h). \tag{5}$$

It has been established that seeking the solution of (2) is equivalent to finding the saddle-point of (5) (cf. [4]). The problem (5) is represented in the algebraic form

$$\begin{bmatrix} A_\eta & B^T \\ B & 0 \end{bmatrix}\begin{bmatrix} u \\ \lambda \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix},$$

where

$$A_\eta = \begin{bmatrix} A_{\Pi\Pi} & A_{\Pi\Delta} \\ A_{\Pi\Delta}^T & A_{\Delta\Delta} + \eta J \end{bmatrix}, \quad B^T = \begin{bmatrix} 0 \\ B_\Delta^T \end{bmatrix}, \quad u = \begin{bmatrix} u_\Pi \\ u_\Delta \end{bmatrix}, \quad F = \begin{bmatrix} f_\Pi \\ f_\Delta \end{bmatrix}.$$

Here, Π indicates the degrees of freedom (dof) associated with both the interior nodes and the subdomain corners, Δ those related to the face and edge nodes on the interface, and λ the Lagrange multipliers for the continuity constraint across the interface. Eliminating u_Π and u_Δ successively, we have a dual system

$$F_\eta\lambda = d_\eta \tag{6}$$

where

$$F_\eta = B_\Delta S_\eta^{-1} B_\Delta^T, \quad d_\eta = B_\Delta S_\eta^{-1}(f_\Delta - A_{\Pi\Delta}^T A_{\Pi\Pi}^{-1} f_\Pi)$$

with

$$S_\eta = S + \eta J = (A_{\Delta\Delta} - A_{\Pi\Delta}^T A_{\Pi\Pi}^{-1} A_{\Pi\Delta}) + \eta J.$$

Note that F_η is symmetric positive definite.

3 Estimate of Condition Number

By letting the vector v_Δ be partitioned into face and edge dof $v_\Delta = \left[v_f, v_e\right]^T$, the pointwise matching operator B_Δ is represented as

$$B_\Delta = \begin{bmatrix} B_f & 0 \\ 0 & B_e \end{bmatrix}.$$

Let us denote by $D(A)$ a block diagonal matrix such that $D(A) = \text{blockdiag}(A)$. Looking at the connection between the operator B_Δ and the penalty term J_η from their definitions, it is obvious that

$$J = \begin{bmatrix} J_F & 0 \\ 0 & J_E \end{bmatrix} = \begin{bmatrix} B_f^T D(J_{B_f}) B_f & 0 \\ 0 & B_e^T D(J_{B_e}) B_e \end{bmatrix}, \tag{7}$$

where J_{B_f} and J_{B_e} stand for the 2D mass matrix on each face weighted with $1/h$ and the 1D mass matrix on each edge, respectively. We define by Λ the space of vectors of dof associated with the Lagrange multipliers. To analyze the condition number bound for F_η, based on Lemma 3.1 in [6], it is sufficient to specify a suitable norm $\|\cdot\|_\Lambda$ on Λ and to estimate constants satisfying

$$\begin{aligned} c_1\|\lambda\|_{\Lambda'}^2 \le \langle \lambda, F_\eta \lambda \rangle \le c_2 \|\lambda\|_{\Lambda'}^2 \quad \forall \lambda \in \Lambda, \\ c_3\|\mu\|_\Lambda^2 \le \langle \mu, \mu \rangle \le c_4 \|\mu\|_\Lambda^2 \quad \forall \mu \in \Lambda. \end{aligned} \tag{8}$$

Taking the structural characteristic of J into consideration, we define the norm $\|\cdot\|_\Lambda$ on Λ by

$$\|\mu\|_\Lambda^2 = \mu^T \begin{bmatrix} D(J_{B_f}) & 0 \\ 0 & D(J_{B_e}) \end{bmatrix} \mu, \quad \forall \mu \in \Lambda. \tag{9}$$

The dual norm on Λ is defined by

$$\|\lambda\|_{\Lambda'} = \max_{\substack{\mu \in \Lambda \\ \mu \neq 0}} \frac{|\langle \lambda, \mu \rangle|}{\|\mu\|_\Lambda}, \quad \forall \lambda \in \Lambda.$$

We list useful results in deriving bounds on the extreme eigenvalues of F_η.

Proposition 1. *For $S = A_{\Delta\Delta} - A_{\Pi\Delta}^T A_{\Pi\Pi}^{-1} A_{\Pi\Delta}$, there exists a constant $C = \lambda_{\max}^S / \lambda_{\min}^J$ such that*

$$v_\Delta^T S v_\Delta \le C v_\Delta^T J v_\Delta, \quad \forall v_\Delta \perp \text{Ker}(B_\Delta),$$

where $\lambda_{\max}^S$ and $\lambda_{\min}^J$ are the maximum eigenvalue of S and the minimum nonzero eigenvalue of J, respectively.

Lemma 1. *Let $\lambda_{\min}^J$ be the minimum nonzero eigenvalue of J. Then, we have*

$$\lambda_{\min}^J \ge Ch$$

where the constant C is independent of h and H.

Thanks to Lemma 3.1 in [6], we have the following estimate of the condition number $\kappa(F_\eta)$.

Theorem 1. *For any $\eta > 0$, we have*

$$\kappa(F_\eta) \le \left(1 + \frac{C}{\eta}\right) C^*$$

where

$$C = \frac{\lambda^S_{\max}}{\lambda^J_{\min}}, \quad C^* = \frac{\max\{\lambda^{J_{B_f}}_{\max}, \lambda^{J_{B_e}}_{\max}\}}{\min\{\lambda^{J_{B_f}}_{\min}, \lambda^{J_{B_e}}_{\min}\}}.$$

Furthermore, the constants C and C^* are independent of the subdomain size H and the mesh size h.

Corollary 1. *For a sufficiently large η, we have*

$$\kappa(F_\eta) \le C^*,$$

where C^* is the constant estimated in Theorem 1.

Remark 1. The condition number estimate of the augmented FETI-DP (with edge constraints) is $C(1+\log(H/h))^2$; see [3]. In our case, the vertex continuity is enough to make our method have a constant bound for the condition number of the resultant dual system.

4 Computational Issues

For the implementation of the proposed algorithm, we reorder the relevant dof in (6). By rearranging u in order $u = [u_r, u_c]^T$ where u_i, u_f, and u_e are assembled into u_r, we obtain a system in the following form

$$K^\eta_{rr} u_r + K_{rc} u_c + B_r^T \lambda = f_r \tag{10a}$$

$$K^T_{rc} u_r + K_{cc} u_c = f_c \tag{10b}$$

$$B_r u_r = 0 \tag{10c}$$

Note that K^η_{rr} is non-singular and detailed as

$$K^\eta_{rr} = K_{rr} + \eta \tilde{J} = \begin{bmatrix} A_{ii} & A_{i\Delta} \\ A^T_{i\Delta} & A_{\Delta\Delta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \eta J \end{bmatrix},$$

where

$$A_{\Delta\Delta} = \begin{bmatrix} A_{ff} & A_{fe} \\ A^T_{fe} & A_{ee} \end{bmatrix}, \quad J = \begin{bmatrix} J_F & 0 \\ 0 & J_E \end{bmatrix}.$$

By eliminating u_r from (10a), we have

$$\begin{bmatrix} F_{cc} & -F_{rc}^T \\ F_{rc} & F_{rr} \end{bmatrix} \begin{bmatrix} u_c \\ \lambda \end{bmatrix} = \begin{bmatrix} d_c \\ d_r \end{bmatrix} \tag{11}$$

where

$$F_{rr} = B_r (K_{rr}^{\eta})^{-1} B_r^T, \quad F_{rc} = B_r (K_{rr}^{\eta})^{-1} K_{rc}, \quad F_{cc} = K_{cc} - K_{rc}^T (K_{rr}^{\eta})^{-1} K_{rc}$$

and

$$d_r = B_r^T (K_{rr}^{\eta})^{-1} f_r, \quad d_c = f_c - K_{rc}^T (K_{rr}^{\eta})^{-1} f_r.$$

Since A_η is invertible, so is F_{cc}. We can therefore eliminate u_c in (11) to get

$$F_\eta \lambda = d_\eta \tag{12}$$

where

$$F_\eta = F_{rr} + F_{rc} F_{cc}^{-1} F_{rc}^T, \quad d_\eta = d_r - F_{rc} F_{cc}^{-1} d_c.$$

Note that the condition number of K_{rr}^{η} grows linearly with η. Hence we need to establish a preconditioner which reduces the effect of η. First, we introduce a preconditioner M_1 as

$$M_1 = \begin{bmatrix} A_{ii} & 0 \\ 0 & A_{\Delta\Delta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \eta J \end{bmatrix}.$$

Theorem 2. *The condition number of the preconditioned system by M_1 grows asymptotically as*

$$\kappa(M_1^{-1} K_{rr}^{\eta}) := \frac{\lambda_{\max}(M_1^{-1} K_{rr}^{\eta})}{\lambda_{\min}(M_1^{-1} K_{rr}^{\eta})} \lesssim \left(\frac{H}{h}\right)^2.$$

Next, we suggest a preconditioner M_2 as

$$M_2 = \begin{bmatrix} A_{ii} & 0 \\ 0 & \tilde{A}_{\Delta\Delta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \eta J \end{bmatrix} \quad \text{with} \quad \tilde{A}_{\Delta\Delta} = \begin{bmatrix} A_{ff} & 0 \\ 0 & A_{ee} \end{bmatrix}.$$

Theorem 3. *Two preconditioners M_1 and M_2 are spectrally equivalent, i.e., there are constants c and C independent of h and H such that*

$$c v_r^T M_2 v_r \le v_r^T M_1 v_r \le C v_r^T M_2 v_r, \quad \forall v_r.$$

Therefore, the condition number of the preconditioned system by M_2 grows asymptotically as

$$\kappa(M_2^{-1} K_{rr}^{\eta}) \lesssim \left(\frac{H}{h}\right)^2.$$

Finally, by eliminating the coupling between all pairs of faces and edges, we establish a preconditioner M_3 as

$$M_3 = \begin{bmatrix} A_{ii} & 0 \\ 0 & \overline{A}_{\Delta\Delta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \eta J \end{bmatrix} \quad \text{with} \quad \overline{A}_{\Delta\Delta} = \begin{bmatrix} \overline{A}_{ff} & 0 \\ 0 & \overline{A}_{ee} \end{bmatrix}.$$

Here, the matrices $\overline{A}_{ff}$ and $\overline{A}_{ee}$ are block diagonal with a block for each face and for each edge, respectively. Also we rewrite A_{ff} and A_{ee} as block matrices of the same structure as $\overline{A}_{ff}$ and $\overline{A}_{ee}$.

Theorem 4. *Assume that on each subdomain Ω_k, a triangulation $\mathcal{T}_{h_k}$ satisfies*

$$Volume(T_c) \leq \min\{Volume(T_c^a)\},$$

where $T_c \in \mathcal{T}_{h_k}$ is a hexahedron containing a subdomain corner as one of its vertices and T_c^a is an adjacent hexahedron to T_c. Then, the condition number of the preconditioned system by M_3 grows asymptotically as

$$\kappa(M_3^{-1}K_{rr}^{\eta}) \lesssim \left(\frac{H}{h}\right)^2.$$

Remark 2. The condition number of the subdomain problem in FETI-DP also depends on the ratio H/h, more presisely, $\kappa(K_{rr}) \leq C(H/h)^3$. In FETI-DP, the subdomain problems are usually solved by direct methods. However, in the case of subdomain problems of relatively large size, iterative solvers are used (cf. [5]).

Acknowledgments This work was supported by NRF-2007-313-C00080.

References

1. C. Farhat, M. Lesoinne, and K. Pierson. A scalable dual-primal domain decomposition method. *Numer. Lin. Alg. Appl.*, 7:687–714, 2000.
2. A. Klawonn and O.B. Widlund. FETI and Neumann–Neumann iterative substructuring methods: Connections and new results. *Commun. Pure Appl. Math.*, 54:57–90, 2001.
3. A. Klawonn, O.B. Widlund, and M. Dryja. Dual-Primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients. *SIAM J. Numer. Anal.*, 40: 159–179, 2002.
4. C.-O. Lee and E.-H. Park. A dual iterative substructuring method with a penalty term. *Numer. Math.*, 112:89–113, 2009.
5. J. Li and O.B. Widlund. On the use of inexact subdomain solvers for BDDC algorithms. *Comput. Methods Appl. Mech. Eng.*, 196:1415–1428, 2007.
6. J. Mandel and R. Tezaur. Convergence of a substructuring method with Lagrange multipliers. *Numer. Math.*, 73:473–487, 1996.

Spectral Element Agglomerate Algebraic Multigrid Methods for Elliptic Problems with High-Contrast Coefficients

Yalchin Efendiev[1], Juan Galvis[1], and Panayot S. Vassilevski[2]

[1] Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA, efendiev@math.tamu.edu[‡]
[2] Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, L-560, Livermore, CA 94550, USA[§]

1 Summary

We apply a recently proposed [5] robust overlapping Schwarz method with a certain spectral construction of the coarse space in the setting of element agglomeration algebraic multigrid methods (or agglomeration AMGe) for elliptic problems with high-contrast coefficients. Our goal is to design multilevel iterative methods that converge independent of the contrast in the coefficients. We present simplified bounds for the condition number of the preconditioned operators. These bounds imply convergence that is independent of the contrast. In the presented preliminary numerical tests, we use geometric agglomerates; however, the algorithm is general and offers some simplifications over the previously proposed spectral agglomerate AMGe methods (cf., [2, 3]).

2 Introduction

The purpose of this paper is to present some preliminary results on the performance of recently proposed overlapping Schwarz methods [5] for elliptic equations with high-contrast coefficients. These methods converge independent of the contrast and use a spectral construction of the coarse space. In this paper, we extend the methods and results of [5] to the multilevel case. The resulting multilevel methods are optimal in terms of the contrast. As it turns out, the resulting multilevel methods can be viewed as a version of previously proposed spectral agglomerate algebraic multigrid methods (or agglomerate ϱAMGe), proposed originally in [3] (see also [1]) and then

[‡] The work of Y.E. is partially supported by NSF and DOE.
[§] The work of this author was performed under the auspices of the U.S. DOE by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

extended in [2] to allow for multilevel recursion without visiting the fine level. A computational survey of various AMGe methods is found in [6], see also [7].

The approach proposed in [5] for the two-level case, which we extend in the present contribution to the multilevel case, needs only to identify the "vertices" of the agglomerates; no additional topological relations are required (assuming that we have somehow come up with an agglomeration algorithm or when geometric meshes are simply used as in our present experiments). The methods introduced here simplify the previously proposed similar spectral agglomerate AMGe methods in [2, 3]. The simplification occurs due to the fact that the present method uses overlapping subdomains (unions of fine grid agglomerates (elements) that share a common vertex) as domains where the local eigenproblems are posed. To define the resulting coarse basis, a partition of unity is applied at the end (as commonly used). To implement the method, we need only an agglomeration algorithm and an algorithm to generate the vertices of the resulting agglomerates. No other topological relations need to be constructed or reduced Schur complements of local matrices need to be computed (as in [2] or [6]) and still the method allows for recursion without visiting the finest grid; see Sect. 3 below for details.

We apply the method to difficult elliptic finite element problems with discontinuous high-contrast coefficients (discontinuity, generally not aligned with the coarse elements). The two-level version of the method was proven in [5] to be robust with respect to the contrast. Our experiments that we present in this contribution show that the same result holds for a more practical multilevel extension of the method that we describe in the Sect. 3. The actual multiplicative and additive versions of the multilevel method are summarized after that in the following two sections, Sects. 4–5. Finally, numerical results are presented in Sect. 7.

3 Notation and Building Tools

Let $D \subset \mathbb{R}^2$ (or $\mathbb{R}^3$) be a polygonal domain. We would like to find $u^* \in H_0^1(D)$ such that

$$a(u^*, v) = f(v) \quad \text{for all } v \in H_0^1(D), \tag{1}$$

where the bilinear form a and the functional f are defined by

$$a(u, v) = \int_D \kappa(x) \nabla u(x) \cdot \nabla v(x) dx \tag{2}$$

and $f(v) = \int_D f(x)v(x)dx$ for all $u, v \in H_0^1(D)$. We allow discontinuous and high-contrast coefficient κ.

Let $\mathcal{T}_h = \mathcal{T}^{(0)}$ be a fine triangulation with mesh parameter $h = h^{(0)}$ and $V_h = V^{(0)}$ be the finite element space of piecewise linear functions on $\mathcal{T}^{(0)}$. The Galerkin formulation of (1) is to find $u^* \in V^{(0)}$ with $a(u^*, v) = f(v)$ for all $v \in V^{(0)}$, or in matrix form

$$Au^* = b, \tag{3}$$

where for all $u, v \in V^h(D)$ we have $u^T A v = \int_D \kappa \nabla u \cdot \nabla v$, and $v^T b = \int_D f v$. We denote $A^{(0)} = A$. For each fine element $\tau \in \mathcal{T}^{(0)}$, let $A_\tau^{(0)}$ be the local finite element matrix. The fine-grid stiffness matrix $A^{(0)}$ can be obtained by the standard assembling procedure based on the local (element) matrices $A_\tau^{(0)}$, $\tau \in \mathcal{T}^{(0)}$.

It is sufficient to consider the case of piecewise constant coefficient κ. From now on we assume that κ is a piecewise constant coefficient in $\mathcal{T}^h$ with value $\kappa = \kappa_e$ on each fine triangulation element $e \in \mathcal{T}^h$.

Introduce a coarser mesh $\mathcal{T}^{(1)} \supset \mathcal{T}^{(0)}$ with parameter $h^{(1)}$. We assume that each coarse element $T_c \in \mathcal{T}^{(1)}$ is the union of fine elements τ with $\tau \in \mathcal{T}^{(0)}$. Define also the subdomains $\{T\}$ as coarse vertex neighborhoods. For each subdomain T, there is coarse vertex x_T such that $T = \cup\{T_c : x_T \in T_c\}$. Each subdomain T contains only one coarse vertex x_T. Now, we define the subdomain matrices $A_T^{(0)}$. For interior floating subdomains, let $A_T^{(0)}$ be finite element Neumann matrix corresponding to that subdomain. If T is a boundary subdomain let $A_T^{(0)}$ be the finite element matrix with homogeneous Neumann boundary conditions in the inner boundary $\partial T \cap D$ and homogeneous Dirichlet boundary conditions in the exterior boundary $\partial T \cap \partial D$. For any subdomain T, the matrix $A_T^{(0)}$ can be obtained by local assembling of finite element matrices as follows

$$A_T^{(0)} = \sum_{\tau \in T} I_\tau^{(0)} A_\tau^{(0)} I_\tau^{(0)T},$$

where $I_\tau^{(0)}$ is the extension by zero operator. Let $M_T^{(0)}$ the weighted mass matrix with coefficient κ and of the same size of $A_T^{(0)}$. We solve the high-contrast eigenvalue problem

$$A_T^{(0)} \phi_k = \lambda_k M_T^{(0)} \phi_k, \quad \phi_k = \phi_{k,T}, \quad \lambda_1 \le \lambda_2 \ldots \lambda_k \le \lambda_{k+1} \le \ldots. \tag{4}$$

In practice, the mass matrix can be replaced with the diagonal of the respective stiffness matrix. This eigenvalue problem reveals the "small" part of the spectrum of the local subdomain matrix $A_T^{(0)}$. It can be shown that only a few small eigenvalues depend on the contrast, i.e., that they vanish asymptotically as the contrast increases. In particular, the number of these eigenvalues is the same as the number of isolated high-conductivity inclusions when homogeneous Neumann boundary conditions in (4) are considered. The idea is to include the corresponding eigenvector information into the coarse space. Let $\{D_T\}$ be a partition unity represented by nonnegative diagonal matrices D_T, that is, $\sum_T I_T D_T I_T^T = Id$, where $Id : V^{(0)} \to V^{(0)}$ is the identity operator and I_T is the extension by zero operator. Let $L_T^{(0)}$ be an integer and define the coarse basis functions associated to the vertex x_T by

$$\Phi_k^{(1)} = D_T P^{(0)} \phi_k^{(0)}, \qquad k = 1, \ldots, L_T, \quad \Phi_k^{(1)} = \Phi_{k,T}^{(1)}, \tag{5}$$

where for convenience we define $P^{(0)}$ as the identity operator. These are the coarse degrees of freedom. That is, we define the coarse space

$$V^{(1)} = \text{Span}\{\Phi_k^{(1)} = \Phi_{k,T}^{(1)} \ \ T \text{ subdomain}, \ 1 \le k \le L_T^{(0)}\}.$$

We mention that one can modify the mass matrix $M_T^{(0)}$ to get coarse spaces with a smaller dimension, see [4].

Let $N_{T_c}^{(1)}$ be the number of coarse degrees of freedom in a coarse element, or coarse basis functions with support containing a coarse element, $T_c \in \mathcal{T}^{(1)}$. With the (new) coarse basis functions we construct local matrices $A_{T_c}^{(1)}$.

Denote by $P^{(1)}$ the matrix whose columns are the coarse basis functions just defined including all subdomains T, that is

$$P^{(1)} = [\Phi_{T,k}^{(1)}]_{T,1\le k\le L_T^{(0)}}.$$

The matrix $P^{(1)} : V^{(1)} \to V^{(0)}$ is the interpolation from the coarse space $V^{(1)}$. We use the Galerkin relation to define the coarse-level "1" matrix

$$A^{(1)} = P^{(1)T} A^{(0)} P^{(1)}.$$

We now consider additional nested coarse meshes $\mathcal{T}^{(2)} \supset \cdots \supset \mathcal{T}^{(L)}$ with parameters $h^{(2)}, \ldots, h^{(\mathcal{L})}$ respectively. The procedure described above can be called recursively to construct coarse spaces $V^{(\ell)}$ and interpolations $P^{(\ell)} : V^{(\ell)} \to V^{(0)}$. At level ℓ, we consider the coarser triangulation $\mathcal{T}^{(\ell+1)}$ and we construct as before: local element matrices, $A_\tau^{(\ell)}, \tau \in \mathcal{T}^{(\ell)}$; subdomain local matrices, $A_T^{(\ell)}, M_T^{(\ell)}$, T subdomain; coarse basis functions,

$$\Phi_k^{(\ell+1)} = D_T P^{(\ell)} \phi_k^{(\ell)}, \quad k = 1, \ldots, L_T^{(\ell)}; \tag{6}$$

interpolation, $P^{(\ell+1)} = [\Phi_{T,k}^{(\ell)}]_{T,k}$; and coarse-level $\ell+1$ matrix defined by $A^{(\ell+1)} = P^{(\ell+1)T} A^{(0)} P^{(\ell+1)}$.

Note that $P^{(\ell+1)} : V^{(\ell+1)} \to V^{(0)}$ where we have defined the coarser space

$$V^{(\ell+1)} = \text{Span}\{\Phi_k^{(\ell)} = \Phi_{k,T}^{(\ell+1)} \ \ T \text{ subdomain}, \ 1 \le k \le L_T^{(0)}\}.$$

Each eigenvalue problem is defined at a current level: $A_T^{(\ell)} \phi_k = \lambda_k M_T^{(\ell)} \phi_k$. These are sparse small size eigenvalue problems. Observe that, in order to construct new coarse basis functions we interpolate the solution of the local weighted eigenvalue problems into the finest space $V^{(0)}$ and then apply the partition of unity. In order to obtain a genuine multilevel construction only a minor modification needs to be done. Definition (6) of the coarse basis functions changes to

$$\widetilde{\Phi}_k^{(\ell+1)} = \widetilde{D}_T^{(\ell)} \phi_k^{(\ell)}, \quad k = 1, \ldots, L_T^{(\ell)}. \tag{7}$$

Where $\widetilde{D}_T^{(\ell)}$ is now a partition of unity of the ℓth (coarse) degrees of freedom. In this case the corresponding spaces $\widetilde{V}^{(\ell)}$ are nested, $\widetilde{V}^{(0)} \supset \widetilde{V}^{(1)} \supset \cdots \supset \widetilde{V}^{(\mathcal{L})}$, with interpolation $\widetilde{P}^{(\ell)} : \widetilde{V}^{(\ell+1)} \to \widetilde{V}^{(\ell)}$ and matrices $\widetilde{A}^{(\ell+1)} = \widetilde{P}^{(\ell+1)T} \widetilde{A}^{(\ell)} \widetilde{P}^{(\ell+1)}$. Usual recursively defined multigrid (or MG) (V-cycle) algorithms can be used based on this construction. The MG algorithm based on the non-nested spaces is summarized in the next section. It falls into the category of "subspace correction" methods.

4 Multigrid Method

Now we describe the multigrid method. We recall that in Sect. 3 we constructed coarse (non-nested) subspaces $V^{(1)}, \ldots, V^{(\mathcal{L})}$ associated to the coarse triangulations $\mathcal{T}^{(1)} \subset \mathcal{T}^{(2)} \cdots \subset \mathcal{T}^{(\mathcal{L})}$. Given $x, b \in V^{(0)}$, we define $y = MG(x, b)$ as corresponding multigrid (V-cycle) operator with (multiplicative) Schwarz smoother with initial guess x and right hand side b. A detailed description of the computations is presented in Fig. 1. The operator $r \to MG(0, r)$, $r \in V^{(0)}$, can be used as a preconditioner. We also consider the genuine multilevel (V-cycle) multigrid method with in the construction using the coarse basis functions defined in (7).

Input: $x, b \in V^{(0)}$. Output: y=MG(x,b).

(i) Initialize $y = x$
(ii) For $\ell = 0, \ldots, \mathcal{L} - 1$, smooth:
  a) Set $r = b - Ay$, $r_\ell = P^{(\ell)T} r$ and $c_\ell = 0$
  b) For $s = 1, \ldots, N_S$, $\quad c_\ell \leftarrow c_\ell + I_{T_s}^{(\ell)} (A_{T_s}^{(\ell)})^{-1} I_{T_s}^{(\ell)T} (r_\ell - A^{(\ell)} c_\ell)$
  c) $y \leftarrow y + P^{(\ell)} c_\ell$
(iii) Coarse correction:
  a) $r = b - Ay$ and $r_{\mathcal{L}} = P^{(\mathcal{L})T} r$
  b) $c_{\mathcal{L}} = (A^{(\mathcal{L})})^{-1} r_{\mathcal{L}}$
  c) $y \leftarrow y + P^{(\mathcal{L})} c_{\mathcal{L}}$
(iv) For $\ell = \mathcal{L} - 1, \ldots, 0$, smooth
  a) $r = b - Ay$, $r_\ell = P^{(\ell)^T} r$ and $c_\ell = 0$
  b) For $s = N_S, \ldots, 1$, $\quad c_\ell \leftarrow c_\ell + I_{T_s}^{(\ell)} (A_{T_s}^{(\ell)})^{-1} I_{T_s}^{(\ell)T} (r_\ell - A^{(\ell)} c_\ell)$
  c) $y \leftarrow y + P^{(\ell)} c_\ell$

Fig. 1. Multigrid operator.

5 Multilevel Additive Preconditioner (BPX)

Now we define a BPX–like additive multilevel method with overlapping Schwarz method as smoother. Given $r \in V^{(0)}$ we define

$$M_{add}^{-1} r = \sum_{\ell=1}^{\mathcal{L}} \sum_{T} P^{(\ell)} I_T^{(\ell)T} (A_T^{(\ell)})^{-1} I_T^{(\ell)} P^{(\ell)T} r,$$

where the second sum runs over all subdomains at level ℓ, $\ell = 1, \ldots, \mathcal{L}$. See [5] for a two-level version of this method. In [5], it is proved that $\mathrm{Cond}(M_{add}^{-1} A^{(0)}) \leq C(1 + \frac{1}{h^{(1)} \lambda_{L+1}})$, where C is a constant independent of the contrast and $\lambda_{L+1} = \min_T \lambda_{L_T^{(1)}+1}$. If, in each subdomain, the right number of basis functions is chosen, then, the previous estimate becomes *independent of the contrast*. We note that the multilevel extension of [5] would require the solution of fine triangulation eigenvalue problems in each subdomain at every level. In our construction in Sect. 3, we solve eigenvalue problems at the actual level.

6 Condition Number Bounds

Now we present a simplified version of the condition number bounds for the methods described above. We recall that in this paper we are interested in the performance of the methods in terms of the contrast. We have the following results which proof uses tools developed in [2, 3, 5], see also [4]. A complete analysis of the proposed algorithms and more detailed numerical experiments are in progress.

Theorem 1. *We have the condition number bounds for the preconditioned operators: Cond*$(MG(0,\cdot)A^{(0)}) \leq C$ *and Cond*$(M_{add}^{-1}A^{(0)}) \leq D$*, where the constants* C *and* D *depend on the number of levels and on the contrast-independent eigenvalues* $\lambda_k = \lambda_{k,T}^{\ell}$*,* $k \geq L_T$*,* T *is a subdomain,* $1 \leq \ell \leq \mathcal{L}$*.*

We stress upon the fact that our experiments indicate that the constants C and D above seem to be mesh-independent (at least in the case of geometric agglomerates). The independence (or exact dependence) of these estimates on the number of levels, as well as the analysis of the genuine multilevel algorithm, are subject of ongoing research.

7 Numerical Experiments

In this section we present representative numerical experiments that show that the proposed methods have an optimal convergence in terms of contrast.

We consider $D = [0,1] \times [0,1]$ and solve problem (1) with two different distributions of high-contrast coefficients κ. In the first experiment, we consider the coefficients depicted on the left of Fig. 2. We use Preconditioned Conjugate Gradient (PCG) with the preconditioners described above, the multigrid, the genuine multilevel and the multilevel additive methods, see Sects. 4 and 5. The mesh and degrees of freedom information for the basis functions constructions in Sect. 3 is displayed in Table 1. The degrees of freedom information corresponding to the genuine multilevel method is similar and it is omitted. In Table 2 we present the (estimated) condition number of the preconditioned operator for different values of the contrast η. We iterate until the initial residual is reduced by a factor of $tol = 10^{-10}$. In our experiments, the selection of the number of small eigenvalues in each subdomain follows the criteria $L_T^{(\ell)} = \min_k \lambda_k > \rho$; for $\ell = 0, \ldots, \mathcal{L}$, where ρ is chosen based on the first large (order one) jump of the eigenvalues (c.f., [2, 5, 6]).

In a second experiment we consider the coefficient depicted on the right of Fig. 2. These coefficients have several inclusions and two long channels. The mesh information is contained in Table 1 and the condition number estimates are given Table 3. As we observe from these numerical results that the proposed methods have optimal convergence in terms of the contrast.

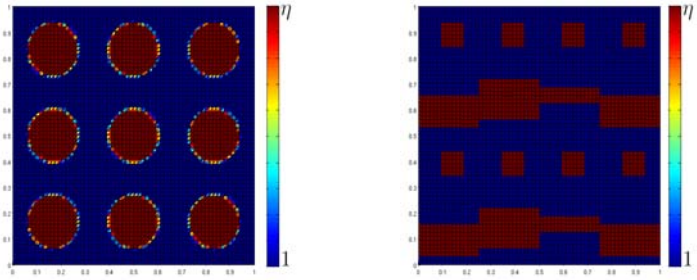


Fig. 2. *Left*: Coefficient corresponding to nine high-contrast inclusions. *Light grey* indicates the high-contrast part, $\kappa(x) = \eta$. *Dark grey* indicates value $\kappa(x) = 1$. The results are displayed in Table 2 and the degrees of freedom information in Table 1. *Right*: Coefficient with inclusions and long channels. See Tables 1 and 3.

Table 1. Mesh information and interior degrees of freedoms in each level for the coefficients depicted in Fig. 2 ($5th$ and $6th$ column, respectively). See Sect. 3.

Level	h	Subdomains	Nodes (int)	dof MG-Ex. 1	dof- MG Ex. 2
0	1/64	32×32	4225(3969)	4225(3969)	4225(3969)
1	1/32	16×16	1089(961)	1089(961)	1089(961)
2	1/16	8×8	289(225)	289(225)	289(225)
3	1/8	4×4	81(49)	117(77)	113(78)
4	1/4	2×2	25(9)	77(45)	38(19)
5	1/2	1×1	9(1)	45(9)	27(6)

Table 2. Condition number estimates for the preconditioned CG. The information on the degrees of freedom is in Table 1 ($5th$ column).

η	Multigrid	Genuine ML	Additive
10^3	2.1389	2.1087	31.9844
10^4	2.3288	2.3201	36.8847
10^5	2.3612	2.4559	37.7580
10^6	2.3647	2.4961	37.8532

Table 3. Condition number estimates for the preconditioned CG. The information on the degrees of freedom is in Table 1 ($6th$ column).

η	Multigrid	Genuine ML	Additive
10^3	1.7780	2.0010	27.0319
10^4	1.7834	2.0751	27.4616
10^5	1.7822	2.0837	27.5052
10^6	1.7829	2.0846	27.5096

References

1. M. Brezina, C. Heberton, J. Mandel, and P. Vanek. An iterative method with convergence rate chosen a priori. UCD/CCM Report 140, 1999.
2. T. Chartier, R. Falgout, V.E. Henson, J.E. Jones, T.A. Manteuffel, S.F. McCormick, J.W. Ruge, and P.S. Vassilevski. Spectral element agglomerate AMGe. In *Domain Decomposition Methods in Science and Engineering XVI*, volume 55 of *Lecture Notes in Computational Science and Engineering*, pp. 513–521. Springer, Berlin, 2007.
3. T. Chartier, R.D. Falgout, V.E. Henson, J. Jones, T. Manteuffel, S. McCormick, J. Ruge, and P. S. Vassilevski. Spectral AMGe (ρAMGe). *SIAM J. Sci. Comput.*, 25(1):1–26, 2003.
4. H. Yanping, R. Kornhuber, O. Widlund, and J. Xu (eds.). Domain Decomposition in Science and Engineering XIX, Springer Verlag, Berlin Heidelberg, p. 189–196, 2011.
5. J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media. to appear in SIAM MMS.
6. I. Lashuk and P.S. Vassilevski. On some versions of the element agglomeration AMGe method. *Numer. Linear Algebra Appl.*, 15(7):595–620, 2008.
7. P.S. Vassilevski. *Multilevel Block Factorization Preconditioners: Matrix-based Analysis and Algorithms for Solving Finite Element Equations*. Springer, New York, NY, 2008.

A FETI-DP Formation for the Stokes Problem Without Primal Pressure Components

Hyea Hyun Kim[1] and Chang-Ock Lee[2]

[1] Department of Mathematics, Chonnam National University, Gwangju, Korea, hyeahyun@gmail.com; hkim@jnu.ac.kr
[2] Department of Mathematical Sciences, KAIST, Daejeon, Korea, colee@kaist.edu

Summary. A scalable FETI–DP (Dual-Primal Finite Element Tearing and Interconnecting) algorithm for the Stokes problem is developed and analyzed. Advantages of this approach are a coarse problem without primal pressure unknowns and the use of a relatively cheap lumped preconditioner. Especially in three dimensions, these advantages provide a more robust and faster FETI-DP algorithm. In three dimensions, the velocity unknowns at subdomain corners and the averages of velocity unknowns over common faces are selected as the primal unknowns in the FETI-DP formulation. A condition number bound of the form $C(H/h)$ is established, where C is a positive constant which is independent of any mesh parameters and H/h is the number of elements across individual subdomains.

1 Introduction

FETI–DP (Dual-Primal Finite Element Tearing and Interconnecting) algorithms are known to be among the most scalable domain decomposition methods, which are iterative substructuring methods based on Lagrange multipliers, see [1, 2]. This family of algorithms was developed for the Stokes problem by [3, 5, 6, 7]. In all these works for the Stokes problem, a compatibility condition on the dual velocity unknowns is required for each subdomain. As a consequence of this requirement, the velocity averages over edges in addition to the velocity unknowns at the subdomain corners are selected as primal unknowns in two dimensions. In three dimensions, the introduction of face averages and more complicated primal unknowns related to edges are unavoidable. By enforcing a compatibility condition on the dual velocity unknowns in each subdomain, additional primal unknown pressure components, constant in each subdomain, are used to enforce these compatibility conditions in these algorithms. This gives an indefinite coarse problem with both primal velocity and primal pressure unknowns.

In our previous work [4], we developed a new FETI-DP algorithm for the Stokes problem in two dimensions. In this algorithm, only velocity unknowns at the subdomain corners are selected as primal variables to reduce complication of the

implementation. The primal pressure components are not used in contrast to other approaches for the Stokes problem. In this formulation, we can eliminate all the pressure unknowns by solving local Stokes problems, since such a selection of the primal velocity unknowns results in dual velocity unknowns which guarantee the solvability of the local Stokes problems without eliminating spurious pressure components. The Dirichlet-type preconditioners are no longer relevant to the FETI-DP formulation and a lumped preconditioner is naturally employed. Its condition number bound $C(H/h)(1+\log(H/h))$ was proved with the constant C depending on the inf-sup constant of a certain pair of velocity and pressure spaces. Furthermore it was shown that the inf-sup constant is independent of any mesh parameters for rectangular subdomain partitions. This method can be considered as an extension of the work in [8] to the Stokes problem.

In this paper, we extend the FETI-DP algorithm without primal pressure unknowns to the three-dimensional Stokes problem. By relaxing the compatibility condition on the dual velocity unknowns, we can select a relatively small set of primal unknowns, which are the primal velocity unknowns at the subdomain corners. To ensure scalability of a method in three dimensions, our method involves only primal velocity unknowns, which are velocity averages over common faces.

The resulting coarse problem of our method consists of only the primal velocity unknowns and becomes symmetric and positive definite. This allows the use of a more practical Cholesky solver for the coarse problem in contrast to indefinite coarse problems that appear in [5, 6, 7, 9]. With a lumped preconditioner, a scalable condition number bound $C(H/h)$ is obtained for the FETI-DP algorithm.

2 FETI-DP Formulation

2.1 Model Problem

We consider the three-dimensional Stokes problem,

$$\begin{aligned} -\triangle \mathbf{u} + \nabla p &= \mathbf{f} \ \text{ in } \Omega, \\ \nabla \cdot \mathbf{u} &= 0 \ \text{ in } \Omega, \\ \mathbf{u} &= 0 \ \text{ on } \partial\Omega, \end{aligned} \tag{1}$$

where Ω is a bounded polyhedral domain in $\mathbf{R}^3$ and $\mathbf{f} \in [L^2(\Omega)]^3$. We introduce a pair of inf-sup stable finite element spaces $(\widehat{X}, \overline{P})$ where $\widehat{X} \subset [H_0^1(\Omega)]^3$ and $\overline{P} \subset L_0^2(\Omega)$. Here $H_0^1(\Omega)$ is the Sobolev space with functions that have square integrable weak derivatives up to the first order and with zero traces on $\partial\Omega$, and $L_0^2(\Omega)$ consists of square integrable functions with average value zero over Ω. In addition, the velocity functions are continuous and the pressure functions can be discontinuous across element boundaries, and $\overline{P} = P \bigcap L_0^2(\Omega)$, where P is the space of pressure functions before enforcing the zero average condition.

2.2 FETI-DP Formulation Without Primal Pressure Components

We now decompose Ω into a non-overlapping subdomain partition $\{\Omega_i\}_{i=1}^N$ in such a way that each subdomain aligns to the finite element triangulation of Ω. The subdomain finite element spaces are then obtained as

$$X^{(i)} = \widehat{X}|_{\Omega_i},\ P^{(i)} = P|_{\Omega_i},$$

that are the restrictions of $\widehat{X}$ and P to the individual subdomains. Among the subdomain velocity unknowns, we select some unknowns at each subdomain boundary as primal unknowns and we denote each part of the subdomain velocity unknowns by $\mathbf{u}_I^{(i)}$, $\mathbf{u}_\Pi^{(i)}$, and $\mathbf{u}_\Delta^{(i)}$, where I, Π, and Δ denote unknowns of the subdomain interior, the primal unknowns, and the remaining dual unknowns at the subdomain boundary, respectively. In the present work, the velocity unknowns at the subdomain corners and the averages of the velocity unknowns over common faces are selected as the primal unknowns.

We introduce the velocity spaces, $X_I^{(i)}$, $X_\Pi^{(i)}$, and $X_\Delta^{(i)}$ corresponding to the unknowns $\mathbf{u}_I^{(i)}$, $\mathbf{u}_\Pi^{(i)}$, and $\mathbf{u}_\Delta^{(i)}$, respectively. We also introduce a space $X_r^{(i)}$ of both the interior and the dual velocity unknowns,

$$X_r^{(i)} = X_I^{(i)} \times X_\Delta^{(i)},$$

and use the notation $\mathbf{u}_r^{(i)}$ for the velocity unknowns in the space $X_r^{(i)}$.

Throughout the paper, for given spaces $W^{(i)}$ associated with Ω_i we denote by W the product space of $W^{(i)}$ and by $\widetilde{W}$ the subspace of W, where the strong continuity at the primal unknowns is enforced. The subspace of W, where continuity at all interface unknowns is enforced, will be denoted by $\widehat{W}$. The unknowns at these spaces W, $\widetilde{W}$, and $\widehat{W}$ are then decoupled, partially coupled, and fully coupled across the subdomain interface, respectively. We also use the same notational convention for the velocity unknowns; $\mathbf{u}_r$ denotes $(\mathbf{u}_r^{(1)}, \ldots, \mathbf{u}_r^{(N)})$ and $\widetilde{\mathbf{u}}$ denotes velocity unknowns in the space $\widetilde{X}$. We will use the same notation $\mathbf{u}$ to denote velocity unknowns and the corresponding finite element function.

We now obtain a discrete form of the Stokes problem (1) in the finite element space $(\widetilde{X}, \overline{P})$ by enforcing the pointwise continuity on the remaining part of the interface unknowns using Lagrange multipliers $\boldsymbol{\lambda} \in M$:

find $((\mathbf{u}_I, \mathbf{u}_\Delta, \widehat{\mathbf{u}}_\Pi), \overline{p}, \boldsymbol{\lambda}) \in \widetilde{X} \times \overline{P} \times M$ such that

$$\begin{pmatrix} K_{II} & K_{I\Delta} & K_{I\Pi} & \overline{B}_I^T & 0 \\ K_{I\Delta}^T & K_{\Delta\Delta} & K_{\Delta\Pi} & \overline{B}_\Delta^T & J_\Delta^T \\ K_{I\Pi}^T & K_{\Delta\Pi}^T & K_{\Pi\Pi} & \overline{B}_\Pi^T & 0 \\ \overline{B}_I & \overline{B}_\Delta & \overline{B}_\Pi & 0 & 0 \\ 0 & J_\Delta & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_I \\ \mathbf{u}_\Delta \\ \widehat{\mathbf{u}}_\Pi \\ \overline{p} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_I \\ \mathbf{f}_\Delta \\ \mathbf{f}_\Pi \\ 0 \\ 0 \end{pmatrix}, \tag{2}$$

where $\overline{B}_I$, $\overline{B}_\Delta$, and $\overline{B}_\Pi$ are defined by

$$-\sum_i \int_{\Omega_i} \nabla \cdot \widetilde{\mathbf{u}}\, q\, dx, \quad \forall q \in \overline{P},$$

J_Δ is a boolean matrix that computes jump of the dual unknowns across the subdomain interface Γ_{ij},

$$J_\Delta \mathbf{u}_\Delta|_{\Gamma_{ij}} = \mathbf{u}_\Delta^{(i)} - \mathbf{u}_\Delta^{(j)},$$

and the other operators are defined by

$$\sum_i \int_{\Omega_i} \nabla \widetilde{\mathbf{u}} \cdot \nabla \widetilde{\mathbf{v}}\, dx.$$

The common interface Γ_{ij} can be an edge or a face of subdomains Ω_i and Ω_j. We introduce fully redundant Lagrange multipliers in our FETI-DP formulation. For $\boldsymbol{\lambda} \in M$, $\boldsymbol{\lambda}|_{F_{ij}}$ denotes the Lagrange multipliers which are related to the continuity constraints $\mathbf{u}_\Delta^{(i)} - \mathbf{u}_\Delta^{(j)} = 0$ on the common face $\overline{F}_{ij}$. Similarly, $\boldsymbol{\lambda}|_{E_{ik}}$ denotes the Lagrange multipliers related to the continuity constraints $\mathbf{u}_\Delta^{(i)} - \mathbf{u}_\Delta^{(k)} = 0$ on the common edge E_{ik}, which is the only common part of the two subdomains Ω_i and Ω_k. We call $\boldsymbol{\lambda}|_{F_{ij}}$ face-based Lagrange multipliers and $\boldsymbol{\lambda}|_{E_{ik}}$ edge-based Lagrange multipliers, respectively.

We recall that the pressure finite element space,

$$\overline{P} = P \bigcap L_0^2(\Omega),$$

where $P = \prod_{i=1}^N P^{(i)}$. These local pressure spaces $P^{(i)}$ do not satisfy the zero average condition. In order to eliminate all the pressure unknowns by solving independent local Stokes problems, we will use the pressure space P instead of $\overline{P}$ in our FETI-DP formulation. By adding a constant pressure component, we extend the pressure space $\overline{P}$ to the space P. The added constant component will give us an additional condition on $\widetilde{\mathbf{u}}$,

$$\sum_i \int_{\Omega_i} \nabla \cdot \widetilde{\mathbf{u}}\, q\, dx = 0, \quad q = c, \tag{3}$$

which is equivalent to

$$\sum_i \int_{\Omega_i} \nabla \cdot \widetilde{\mathbf{u}}\, c\, dx = c \sum_{ij} \int_{F_{ij}} (\mathbf{u}_\Delta^{(i)} - \mathbf{u}_\Delta^{(j)}) \cdot \mathbf{n}_{ij}\, ds = 0.$$

Here F_{ij} denotes the common face of two subdomains Ω_i and Ω_j. The above equation can be obtained as a linear combination of the continuity constraints on $\mathbf{u}_\Delta$,

$$J_\Delta \mathbf{u}_\Delta = 0.$$

Since $J_\Delta \mathbf{u}_\Delta = 0$ has been already enforced in (2), by adding (3) to the algebraic system (2), we obtain an extended algebraic system which is equivalent to (2).

We write the extended algebraic system with the pressure space P as follows: find $((\mathbf{u}_I, \mathbf{u}_\Delta, \widehat{\mathbf{u}}_\Pi), p, \boldsymbol{\lambda}) \in (\widetilde{X}, P, M)$ such that

$$\begin{pmatrix} K_{II} & K_{I\Delta} & K_{I\Pi} & B_I^T & 0 \\ K_{I\Delta}^T & K_{\Delta\Delta} & K_{\Delta\Pi} & B_\Delta^T & J_\Delta^T \\ K_{I\Pi}^T & K_{\Delta\Pi}^T & K_{\Pi\Pi} & B_\Pi^T & 0 \\ B_I & B_\Delta & B_\Pi & 0 & 0 \\ 0 & J_\Delta & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_I \\ \mathbf{u}_\Delta \\ \widehat{\mathbf{u}}_\Pi \\ p \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_I \\ \mathbf{f}_\Delta \\ \mathbf{f}_\Pi \\ 0 \\ 0 \end{pmatrix}. \tag{4}$$

Here B_I, B_Δ, and B_Π are defined by

$$-\sum_i \int_{\Omega_i} \nabla \cdot \widetilde{\mathbf{u}}\, q\, dx, \quad \forall q \in P,$$

and the other terms are the same as those in (2).

In the new algebraic form, the unknowns $(\mathbf{u}_I, \mathbf{u}_\Delta, p)$ can be eliminated by solving independent local problems. The advantage of the extended algebraic system is that no pressure unknowns are left and only the primal velocity unknowns remain after solving the local problems. The primal velocity unknowns can be eliminated by solving a global coarse problem, which is smaller and more practical than those of other domain decomposition algorithms for the Stokes problem [5, 6, 7, 9]. As a result a linear system on $\boldsymbol{\lambda}$ will be obtained,

$$F_{DP}\boldsymbol{\lambda} = d.$$

The introduction of fully redundant Lagrange multipliers and the extension of the pressure space make the resulting system singular.

The extension of the pressure space introduces one more null space component which is given by

$$\boldsymbol{\mu}_0|_{F_{ij}} = \zeta_{ij}\mathbf{n}_{ij}, \ \forall F_{ij} \ \text{ and } \ \boldsymbol{\mu}_0|_{E_{lk}} = 0, \ \forall E_{lk}.$$

Here $\boldsymbol{\mu}_0|_{F_{ij}}$ and $\boldsymbol{\mu}_0|_{E_{lk}}$ are face-based and edge-based Lagrange multipliers, respectively, $\mathbf{n}_{ij}$ is the unit normal to the face F_{ij}, and at each nodal point $x_l \in \overline{F}_{ij}$, $\zeta_{ij}(x_l)$ is given by

$$\zeta_{ij}(x_l) = \int_{F_{ij}} \phi_l(x(s), y(s), z(s))\, ds,$$

where ϕ_l is the velocity basis function related to the node x_l. This can be shown by observing that $(\mathbf{u}_I, \mathbf{u}_\Delta, \widehat{\mathbf{u}}_\Pi) = 0$, $p = c$, and $\boldsymbol{\lambda} = c\boldsymbol{\mu}_0$ are solutions of (4) for the zero force terms $(\mathbf{f}_I, \mathbf{f}_\Delta, \mathbf{f}_\Pi) = 0$ with c an arbitrary constant.

Let Range(J_Δ) be the range space of J_Δ. We then have

$$M = \text{Null}(J_\Delta^T) \bigoplus \text{Range}(J_\Delta).$$

We now introduce a subspace of M, which is orthogonal to the null space components of F_{DP},

$$M_c = \left\{ \boldsymbol{\mu} \in \text{Range}(J_\Delta) \,:\, \boldsymbol{\mu}^t \boldsymbol{\mu}_0 = 0 \right\}.$$

We then perform the preconditioned conjugate gradient iteration in the subspace M_c by employing a lumped preconditioner $\widehat{M}^{-1}$,

$$\widehat{M}^{-1} = \begin{pmatrix} 0 \\ J_\Delta^T \\ 0 \end{pmatrix}^T \begin{pmatrix} K_{II} & K_{I\Delta} & B_I^T \\ K_{I\Delta}^T & K_{\Delta\Delta} & B_\Delta^T \\ B_I & B_\Delta & 0 \end{pmatrix} \begin{pmatrix} 0 \\ J_\Delta^T \\ 0 \end{pmatrix} = J_\Delta K_{\Delta\Delta} J_\Delta^T.$$

3 Analysis of a Bound of Condition Number

In this section, we will provide a condition number bound of the FETI-DP operator with the lumped preconditioner by proving the following inequalities:

$$C_1 \beta^2 \langle \widehat{M}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle \le \langle F_{DP}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle \le C_2 \left(\frac{H}{h} \right) \langle \widehat{M}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle, \quad \forall \boldsymbol{\lambda} \in M_c, \tag{5}$$

where β is the inf-sup constant of a certain pair of velocity and pressure finite element spaces, $(\widehat{E}_{I,\Pi}, \overline{P})$. In a more detail, $\widehat{E}_{I,\Pi} = X_I + \widehat{E}_\Pi$ and $\widehat{E}_\Pi$ consists of functions $\mathbf{v}$ in $\widehat{X}$, which minimize the H^1-discrete seminorm and satisfy

$$\begin{aligned} \mathbf{v}(V) &= \mathbf{a}_V, \\ \textstyle\int_F I^h(\theta_F \mathbf{v})(x)\, dx(s) &= \mathbf{a}_F, \\ \textstyle\int_F I^h(\theta_E \mathbf{v})(x) \cdot \mathbf{n}_F\, dx(s) &= a_F^E, \quad \forall F \in \mathcal{F}(E), \end{aligned}$$

with given values of $\mathbf{a}_V$, $\mathbf{a}_F$, and a_F^E, which are provided for all vertices $V \in \mathcal{V}$, all faces $F \in \mathcal{F}$, and all edges $E \in \mathcal{E}$. Here $\mathcal{F}(E)$ is the set of faces with the edge E in common, and θ_F and θ_E are face and edge cut-off functions, which are one at the interior nodes of the face F, and those of the edge E, respectively, and zero at the other unknowns. In addition, $I^h(\mathbf{v})$ is the nodal interpolant of $\mathbf{v}$ to the velocity finite element space $\widehat{X}$.

These inequalities in (5) yield the following condition number bound,

$$\kappa(\widehat{M}^{-1} F_{DP}) \le C \frac{1}{\beta^2} \left(\frac{H}{h} \right).$$

3.1 Lower Bound

Lemma 1. *For any $\boldsymbol{\mu} \in M_c$, there exists $\mathbf{u} \in \widetilde{X}$ such that*
1. $J_\Delta \mathbf{u}_\Delta = \boldsymbol{\mu}$,
2. $\sum_i \int_{\Omega_i} \nabla \cdot \mathbf{u}\, q\, dx = 0, \quad \forall q \in P$,
3. $\langle K\mathbf{u}, \mathbf{u} \rangle \le C \frac{1}{\beta^2} \langle K_{\Delta\Delta} J_\Delta^T \boldsymbol{\mu}, J_\Delta^T \boldsymbol{\mu} \rangle$, where β is the inf-sup constant of the pair $(\widehat{E}_{I,\Pi}, \overline{P})$.

We introduce

$$\widetilde{X}(\text{div}) = \left\{ \mathbf{v} \in \widetilde{X} \ : \ \int_{\Omega_i} \nabla \cdot \mathbf{v} q \, dx = 0, \quad \forall q \in P \right\}.$$

We then have the identity,

$$\langle F_{DP}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle = \max_{\mathbf{v} \in \widetilde{X}(\text{div})} \frac{\langle J_\Delta \mathbf{v}_\Delta, \boldsymbol{\lambda} \rangle^2}{\langle K\mathbf{v}, \mathbf{v} \rangle}, \tag{6}$$

where K is the stiffness matrix given by

$$K = \begin{pmatrix} K_{II} & K_{I\Delta} & K_{I\Pi} \\ K_{I\Delta}^T & K_{\Delta\Delta} & K_{\Delta\Pi} \\ K_{I\Pi}^T & K_{\Delta\Pi}^T & K_{\Pi\Pi} \end{pmatrix}.$$

The following lower bound can be obtained from Lemma 1 and (6):

Theorem 1. *For any $\boldsymbol{\lambda} \in M_c$, we have*

$$C_1 \beta^2 \langle \widehat{M}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle \leq \langle F_{DP}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle,$$

where β is the inf-sup constant of the pair $(\widehat{E}_{I,\Pi}, \overline{P})$ and C_1 is a positive constant that does not depend on any mesh parameters.

3.2 Upper Bound

The following result is obtained from a Poincaré inequality, see [8, Lemma 4]:

Lemma 2. *Let Ω_i be a three-dimensional subdomain. For any function $v \in H^1(\Omega_i)$,*

$$\|v - c_F\|^2_{L^2(F)} \leq CH|v|^2_{H^1(\Omega_i)},$$

where F is a face of the subdomain Ω_i and c_F is given by

$$c_F = \frac{\int_F I^h(\theta_F v) dx(s)}{\int_F dx(s)}.$$

By using the above lemma, we obtain:

Lemma 3. *There exists a constant C such that*

$$\langle K_{\Delta\Delta} J_\Delta^T J_\Delta \mathbf{u}_\Delta, J_\Delta^T J_\Delta \mathbf{u}_\Delta \rangle \leq C \frac{H}{h} \langle K\mathbf{u}, \mathbf{u} \rangle, \quad \text{for any } \mathbf{u} \in \widetilde{X}.$$

The identity in (6) combined with Lemma 3 gives the upper bound:

Theorem 2. *For any $\boldsymbol{\lambda} \in M_c$, we have*

$$\langle F_{DP}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle \leq C_2 \frac{H}{h} \langle \widehat{M}\boldsymbol{\lambda}, \boldsymbol{\lambda} \rangle,$$

where C_2 is a positive constant that does not depend on any mesh parameters.

References

1. C. Farhat, M. Lesoinne, and K. Pierson. A scalable dual-primal domain decomposition method. *Numer. Linear Algebra Appl.*, 7(7–8):687–714, 2000.
2. C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, and D. Rixen. FETI-DP: a dual-primal unified FETI method. I. A faster alternative to the two-level FETI method. *Int. J. Numer. Methods Eng.*, 50(7):1523–1544, 2001.
3. H.H. Kim and C.-O. Lee. A Neumann-Dirichlet preconditioner for a FETI-DP formulation of the two-dimensional Stokes problem with mortar methods. *SIAM J. Sci. Comput.*, 28 (3):1133–1152, 2006.
4. H.H. Kim, C.-O. Lee, and Eun-Hee Park. A FETI-DP formulation for the Stokes problem without primal pressure components. *To appear in SIAM J. Numer. Anal.*
5. J. Li. Dual-primal FETI methods for stationary Stokes and Navier-Stokes equations. Ph. D. Thesis, Department of Mathematics, Courant Institute, New York University, New York, NY, 2002.
6. J. Li. A dual-primal FETI method for incompressible Stokes equations. *Numer. Math.*, 102 (2):257–275, 2005.
7. J. Li and O. Widlund. BDDC algorithms for incompressible Stokes equations. *SIAM J. Numer. Anal.*, 44(6):2432–2455, 2006.
8. J. Li and O.B. Widlund. On the use of inexact subdomain solvers for BDDC algorithms. *Comput. Methods Appl. Mech. Eng.*, 196:1415–1428, 2007.
9. L.F. Pavarino and O.B. Widlund. Balancing Neumann-Neumann methods for incompressible Stokes equations. *Commun. Pure Appl. Math.*, 55(3):302–335, 2002.

Schwarz Waveform Relaxation Methods for Systems of Semi-Linear Reaction-Diffusion Equations

Stéphane Descombes[1], Victorita Dolean[1], and Martin J. Gander[2]

[1] Université de Nice Sophia-Antipolis, Laboratoire J.-A. Dieudonné, UMR CNRS 6621, 06018 Nice 02, France, Stephane.Descombes@unice.fr; victoria.Dolean@unice.fr

[2] Université de Genève, Section de Mathématiques, CP 64, 1211 Genève, Geneva CH-1211, Switzerland, martin.gander@unige.ch

Summary. Schwarz waveform relaxation methods have been studied for a wide range of scalar linear partial differential equations (PDEs) of parabolic and hyperbolic type. They are based on a space-time decomposition of the computational domain and the subdomain iteration uses an overlapping decomposition in space. There are only few convergence studies for non-linear PDEs.

We analyze in this paper the convergence of Schwarz waveform relaxation applied to systems of semi-linear reaction-diffusion equations. We show that the algorithm converges linearly under certain conditions over long time intervals. We illustrate our results, and further possible convergence behavior, with numerical experiments.

1 Introduction

Schwarz waveform relaxation methods are domain decomposition methods for evolution problems, which were invented independently in [1, 6, 7], where the latter paper only appeared several years later in print. These methods use a domain decomposition in space, and a subdomain iteration in space-time to converge to the underlying time dependent solution, see Fig. 1 for an illustration. Schwarz waveform relaxation methods exhibit different convergence behaviors, depending on the underlying PDE and the time interval of the simulation: for the heat equation, convergence is linear over long times, see [6], and superlinear over short times, see [7]. For the wave equation, convergence is obtained in a finite number of steps for bounded time intervals, see [4], where also an optimized variant is described, which was first proposed in [3], both for hyperbolic and parabolic problems.

The analysis of Schwarz waveform relaxation methods for nonlinear problems is significantly more difficult: for scalar semilinear reaction diffusion problems, see [2], and for scalar convection dominated nonlinear conservation laws, see [5]. The purpose of our paper is to present a first convergence analysis for systems of nonlinear PDEs, for the model problem of semilinear reaction diffusion equations.

2 Systems of Semi-linear Reaction Diffusion Equations

To simplify the presentation, we show our results for a system of two equations in one spatial dimension, but the techniques used in the analysis can be generalized to systems with more unknowns, and also to higher dimensions. We consider on a bounded domain $\Omega \subset \mathbb{R}$ the system of semi-linear reaction diffusion equations

$$\begin{aligned} \partial_t \boldsymbol{u} - \Delta \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) &= 0 \qquad \text{in } \Omega \times (0,T), \\ \boldsymbol{u}(x,t) &= \boldsymbol{g}(x,t) \text{ on } \partial\Omega \times (0,T), \\ \boldsymbol{u}(x,0) &= \boldsymbol{u}_0(x) \text{ in } \Omega, \end{aligned} \tag{1}$$

where $\boldsymbol{u} = (u_1, u_2)$ represents the vector of two unknown concentrations to be determined, and $\boldsymbol{f}(\boldsymbol{u}) = (f_1(u_1,u_2), f_2(u_1,u_2))$. A well posedness result for such systems of semi-linear reaction diffusion equations can be found in [8], see Corollary 3.3.5, p. 56.

Our analysis of the Schwarz waveform relaxation algorithm is based on comparison principles. Such principles have been studied in various contexts for system (1), see for example [10, 11], and they often require quite elaborate proofs for the generality employed. We state here precisely the results we need.

Lemma 1. *Let $\boldsymbol{u} = (u_j)_{1\le j\le 2} \in C^{2,1}(\Omega \times [0,\infty))^2$ be a function for which each component satisfies the inequality*

$$\begin{aligned} \partial_t u_i - \Delta u_i + a_{i1}(x,t)u_1 + a_{i2}(x,t)u_2 &> 0 \text{ in } \Omega \times (0,\infty), \\ u_i(x,t) &> 0 \text{ on } \partial\Omega \times (0,\infty), \\ u_i(x,0) &> 0 \text{ in } \Omega. \end{aligned} \tag{2}$$

If $a_{ij}(x,t) \le 0$ for $i \ne j$ and all $(x,t) \in \Omega \times (0,\infty)$, then $u_i(x,t) > 0$ for all $(x,t) \in \Omega \times (0,\infty)$.

The proof of this theorem by contradiction is a straightforward extension of the result in the scalar case, see [2]. The strict inequalities in Lemma 1 can however be relaxed, as we show next.

Lemma 2. *Under the same assumptions as in Lemma 1, if*

$$\begin{aligned} \partial_t u_i - \Delta u_i + a_{i1}(x,t)u_1 + a_{i2}(x,t)u_2 &\ge 0 \text{ in } \Omega \times (0,\infty), \\ u_i(x,t) &\ge 0 \text{ on } \partial\Omega \times (0,\infty), \\ u_i(x,0) &\ge 0 \text{ in } \Omega, \end{aligned} \tag{3}$$

then $u_i(x,t) \ge 0$ for all $(x,t) \in \Omega \times (0,\infty)$.

Proof. By performing the change of variables $\tilde{u}_i(x,t) := e^{Ct}u_i(x,t)$, where C is a constant to be chosen, the first inequality of (3) can be rewritten as

$$\partial_t \tilde{u}_i - \Delta \tilde{u}_i - C\tilde{u}_i + a_{i1}\tilde{u}_1 + a_{i2}\tilde{u}_2 \ge 0. \tag{4}$$

Now let $\hat{u} = \tilde{u} + \varepsilon$. If we rewrite (4) in terms of $\hat{u}$, and choose the constant C such that $C < a_{i1} + a_{i2}$, we can apply Lemma 1, and taking the limit for $\varepsilon \to 0$ shows that $u_i(x,t) \ge 0$.

3 Schwarz Waveform Relaxation Algorithm

We consider the semi-linear reaction diffusion system (1) in the domain $\Omega = (0, L)$. We decompose the domain into two overlapping subdomains $\Omega_1 = (0, \beta L)$ and $\Omega_2 = (\alpha L, L)$, $\alpha < \beta$, as shown in Fig. 1. We denote by $\boldsymbol{g_1}(t) := \boldsymbol{g}(0, t)$ and by

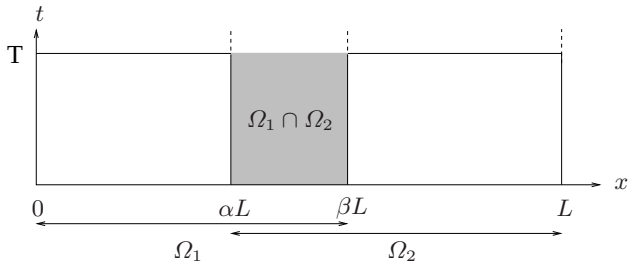


Fig. 1. Space-time domain decomposition.

$\boldsymbol{g_2}(t) := \boldsymbol{g}(L, t)$. The classical Schwarz waveform relaxation algorithm constructs at each iteration n approximate solutions $\boldsymbol{v}^n$, $\boldsymbol{w}^n$ on subdomains Ω_i, $i = 1, 2$, by solving the equations

$$\begin{aligned}
\partial_t \boldsymbol{v}^{n+1} - \Delta \boldsymbol{v}^{n+1} + \boldsymbol{f}(\boldsymbol{v}^{n+1}) &= 0 && \text{in } \Omega_1 \times (0,T),\\
\boldsymbol{v}^{n+1}(0,t) &= \boldsymbol{g}_1(t) && \text{on } (0,T),\\
\boldsymbol{v}^{n+1}(\beta L,t) &= \boldsymbol{w}^{n}(\beta L,t) && \text{on } (0,T),\\
\boldsymbol{v}^{n+1}(x,0) &= \boldsymbol{u}_0(x) && \text{in } \Omega_1\\
\partial_t \boldsymbol{w}^{n+1} - \Delta \boldsymbol{w}^{n+1} + \boldsymbol{f}(\boldsymbol{w}^{n+1}) &= 0 && \text{in } \Omega_2 \times (0,T),\\
\boldsymbol{w}^{n+1}(\alpha L,t) &= \boldsymbol{v}^{n}(\alpha L,t) && \text{on } (0,T),\\
\boldsymbol{w}^{n+1}(L,t) &= \boldsymbol{g}_2(t) && \text{on } (0,T),\\
\boldsymbol{w}^{n+1}(x,0) &= \boldsymbol{u}_0(x) && \text{in } \Omega_2.
\end{aligned} \tag{5}$$

In order to analyze the convergence of the Schwarz waveform relaxation algorithm (5) to the solution $\boldsymbol{u}$ of (1), we denote the errors in subdomain Ω_1 by $\boldsymbol{d}^n := \boldsymbol{u} - \boldsymbol{v}^n$ and in Ω_2 by $\boldsymbol{e}^n := \boldsymbol{u} - \boldsymbol{w}^n$. The error equations are

$$\begin{aligned}
\partial_t \boldsymbol{d}^{n+1} - \Delta \boldsymbol{d}^{n+1} + \boldsymbol{f}(\boldsymbol{u}) - \boldsymbol{f}(\boldsymbol{v}^{n+1}) &= 0 \quad \text{in } \Omega_1 \times (0,T),\\
\boldsymbol{d^{n+1}}(\beta L,t) &= \boldsymbol{e}^{n}(\beta L,t) \quad \text{on } (0,T),\\
\partial_t \boldsymbol{e}^{n+1} - \Delta \boldsymbol{e}^{n+1} + \boldsymbol{f}(\boldsymbol{u}) - \boldsymbol{f}(\boldsymbol{w}^{n+1}) &= 0 \quad \text{in } \Omega_2 \times (0,T),\\
\boldsymbol{e^{n+1}}(\alpha L,t) &= \boldsymbol{d}^{n}(\alpha L,t) \quad \text{on } (0,T),
\end{aligned} \tag{6}$$

where the initial and boundary conditions on the exterior boundaries are zero, since the error vanishes there. Using a Taylor expansion with remainder term in Lagrange form, we obtain for $i = 1, 2$

$$\partial_t d_i^{n+1} - \Delta d_i^{n+1} + \partial_1 f_i(\xi_{i,1}, u_2) d_1^{n+1} + \partial_2 f_i(v_1^{n+1}, \xi_{i,2}) d_2^{n+1} = 0 \text{ in } \Omega_1 \times (0,T),$$
$$\partial_t e_i^{n+1} - \Delta e_i^{n+1} + \partial_1 f_i(\xi'_{i,1}, u_2) e_1^{n+1} + \partial_2 f_i(w_1^{n+1}, \xi'_{i,2}) e_2^{n+1} = 0 \text{ in } \Omega_2 \times (0,T),$$

a linear system with variable coefficients, depending on the Jacobian of the non-linearity.

Our convergence analysis is based on upper solutions for the errors, which are constant in time.

Theorem 1 (Linear Convergence Estimate). *Assume that $\partial_1 f_1 \geq 0$ and $\partial_2 f_2 \geq 0$, and that there exists a constant a satisfying $0 < a < (\pi/L)^2$, such that $-a \leq \partial_1 f_2 \leq 0$ and $-a \leq \partial_2 f_1 \leq 0$. Then the errors in the Schwarz waveform relaxation algorithm (5) satisfy*

$$\sup_{x\in\Omega_1} \|\boldsymbol{d}^{2n+1}(x,\cdot)\|_\infty \leq C_1\gamma^k \|\boldsymbol{e}^0(\beta L,\cdot)\|_\infty, \tag{7}$$

$$\sup_{x\in\Omega_2} \|\boldsymbol{e}^{2n+1}(x,\cdot)\|_\infty \leq C_2\gamma^k \|\boldsymbol{d}^0(\alpha L,\cdot)\|_\infty, \tag{8}$$

where γ in $(0,1)$ is

$$\gamma = \left(\frac{\sin(\sqrt{a}\alpha L)}{\sin(\sqrt{a}\beta L)}\right)\left(\frac{\sin(\sqrt{a}(1-\beta)L)}{\sin(\sqrt{a}(1-\alpha)L)}\right),$$

and the constants $C_{1,2}$ are given by

$$C_1 = \sup_{0<x<\beta L} \frac{\sin(\sqrt{a}x)}{\sin(\sqrt{a}\beta L)}, \quad C_2 = \sup_{\alpha L<x<L} \frac{\sin(\sqrt{a}(L-x))}{\sin(\sqrt{a}(1-\alpha)L)}.$$

Proof. Let $M := \max\{||e_1^n(\beta L,\cdot)||_\infty, ||e_2^n(\beta L,\cdot)||_\infty\}$ and define the function $\tilde{d}^{n+1} := M\sin(\sqrt{a}x)/\sin(\sqrt{a}\beta L)$, which is the unique solution of the steady state problem

$$-\Delta\tilde{d}^{n+1} - a\tilde{d}^{n+1} = 0, \quad \text{with} \quad \tilde{d}^{n+1}(0) = 0, \quad \tilde{d}^{n+1}(\beta L) = M.$$

In order to show that $\tilde{d}^{n+1}$ is a supersolution of the two errors d_1^{n+1} and d_2^{n+1}, we consider the differences $\hat{d}_1^{n+1} := \tilde{d}^{n+1} - d_1^{n+1}$ and $\hat{d}_2^{n+1} := \tilde{d}^{n+1} - d_2^{n+1}$, which satisfy in Ω_1 the system of equations

$$\begin{aligned}
&\partial_t\hat{d}_1^{n+1} - \Delta\hat{d}_1^{n+1} - a\tilde{d}^{n+1} - \partial_1 f_1(\xi_{1,1},u_2)d_1^{n+1} - \partial_2 f_1(v_1^{n+1},\xi_{1,2})d_2^{n+1} = 0,\\
&\partial_t\hat{d}_2^{n+1} - \Delta\hat{d}_2^{n+1} - a\tilde{d}^{n+1} - \partial_1 f_2(\xi_{2,1},u_2)d_1^{n+1} - \partial_2 f_2(v_1^{n+1},\xi_{2,2})d_2^{n+1} = 0.
\end{aligned}$$

Adding and subtracting $\partial_1 f_1(\xi_{1,1},u_2)\tilde{d}^{n+1}$ and $\partial_2 f_1(v_1^{n+1},\xi_{1,2})\tilde{d}^{n+1}$ in the first equation, and $\partial_1 f_2(\xi_{2,1},u_2)\tilde{d}^{n+1}$ and $\partial_2 f_2(v_1^{n+1},\xi_{2,2})\tilde{d}^{n+1}$ in the second equation, we obtain

$$\begin{aligned}
\partial_t\hat{d}_1^{n+1} - \Delta\hat{d}_1^{n+1} &= -\partial_1 f_1(\xi_{1,1},u_2)\hat{d}_1^{n+1} - \partial_2 f_1(v_1^{n+1},\xi_{1,2})\hat{d}_2^{n+1}\\
&\quad +\partial_1 f_1(\xi_{1,1},u_2)\tilde{d}^{n+1} + (a + \partial_2 f_1(v_1^{n+1},\xi_{1,2}))\tilde{d}^{n+1},\\
\partial_t\hat{d}_2^{n+1} - \Delta\hat{d}_2^{n+1} &= -\partial_1 f_2(\xi_{2,1},u_2)\hat{d}_1^{n+1} - \partial_2 f_2(v_1^{n+1},\xi_{2,2})\hat{d}_2^{n+1}\\
&\quad +(a + \partial_1 f_2(\xi_{2,1},u_2))\tilde{d}^{n+1} + \partial_2 f_2(v_1^{n+1},\xi_{2,2})\tilde{d}^{n+1}.
\end{aligned}$$

Under the assumptions of the theorem, and using the fact that $\tilde{d}^{n+1}$ is strictly positive in the interior of subdomain Ω_1, we obtain the system of inequalities

$$\partial_t \hat{d}_1^{n+1} - \Delta \hat{d}_1^{n+1} + \partial_1 f_1(\xi_{1,1}, u_2)\hat{d}_1^{n+1} + \partial_2 f_1(v_1^{n+1}, \xi_{1,2})\hat{d}_2^{n+1} \geq 0,$$
$$\partial_t \hat{d}_2^{n+1} - \Delta \hat{d}_2^{n+1} + \partial_1 f_2(\xi_{2,1}, u_2)\hat{d}_1^{n+1} + \partial_2 f_2(v_1^{n+1}, \xi_{2,2})\hat{d}_2^{n+1} \geq 0.$$

Since $\partial_2 f_1(v_1^{n+1}, \xi_{1,2}) \leq 0$ and $\partial_1 f_2(\xi_{2,1}, u_2) \leq 0$, we can now apply Lemma 2 to conclude that $\hat{d}_1^{n+1} = \tilde{d}^{n+1} - d_1^{n+1} \geq 0$ and $\hat{d}_2^{n+1} = \tilde{d}^{n+1} - d_2^{n+1} \geq 0$ in Ω_1. Using a similar argument for the sums, we obtain that also $\tilde{d}^{n+1} + d_1^{n+1} \geq 0$ and $\tilde{d}^{n+1} + d_2^{n+1} \geq 0$, which implies by the positivity of $\tilde{d}^{n+1}$ that their modulus is bounded, and we have for $i = 1, 2$

$$|d_i^{n+1}(x,t)| \leq \tilde{d}^{n+1} = \max\left\{||e_1^n(\beta L, \cdot)||_\infty, ||e_2^n(\beta L, \cdot)||_\infty\right\} \frac{\sin(\sqrt{a}x)}{\sin(\sqrt{a}\beta L)}.$$

Using a similar argument on subdomain Ω_2, we obtain for $i = 1, 2$

$$|e_i^{n+1}(x,t)| \leq \max\left\{||d_1^n(\alpha L, \cdot)||_\infty, ||d_2^n(\alpha L, \cdot)||_\infty\right\} \frac{\sin(\sqrt{a}(L-x))}{\sin(\sqrt{a}(1-\alpha)L)}.$$

Since the bounds are uniform in t, we obtain over a double step

$$\|\boldsymbol{d}^{n+1}(\alpha L, \cdot)\|_\infty \leq \gamma \|\boldsymbol{d}^{n-1}(\alpha L, \cdot)\|_\infty,$$
$$\|\boldsymbol{e}^{n+1}(\beta L, \cdot)\|_\infty \leq \gamma \|\boldsymbol{e}^{n-1}(\beta L, \cdot)\|_\infty,$$

and by induction (7) and (8). The fact that $\gamma < 1$ has been shown already in [2].

4 Numerical Results

We present numerical results for three different semilinear systems: the Belousov-Zhabotinsky equations, the FitzHugh-Nagumo equations, and the Lotka-Volterra system with migration. All numerical experiments are performed in the domain $\Omega = (0, 1)$ and on the time interval $(0, T)$, with $T = 12\pi$. We discretize the equations with a standard three point finite difference method in space (with mesh size $\Delta x = \frac{1}{20}$), and a semi-implicit Euler time-discretization scheme (with time step $\Delta t = \frac{\pi}{10}$), where implicit integration is used for the diffusive term and explicit integration for the reaction term. The space-time domain Ω is decomposed into two overlapping domains (with overlap size $\delta = 2\Delta x$) and we use Dirichlet conditions at the interfaces.

4.1 Belousov-Zhabotinsky Equations

The Belousov-Zhabotinsky equations model non-equilibrium thermodynamics, resulting in the establishment of a nonlinear chemical oscillator, see [9], p. 322 for details. They are given by

$$\begin{aligned} &\partial_t u_1 - \tfrac{1}{2}\partial_{xx} u_1 - u_1(1 - u_1 - r u_2) = 0, \\ &\partial_t u_2 - \tfrac{1}{2}\partial_{xx} v + b u_1 u_2 = 0. \end{aligned} \tag{9}$$

The hypotheses of Theorem 1 are satisfied for this system after performing the change of variables $\tilde{u}_1 = 1 - u_1, \tilde{u}_2 = u_2$, under the condition that the components u_1 and u_2 remain positive. This condition holds, provided that the initial conditions satisfy $u_1(x, 0) = u_{1,0}(x) \geq 0$ and $u_2(x, 0) = u_{2,0}(x) \geq 0$. Figure 2 shows the linear convergence predicted by the convergence bound of Theorem 1.

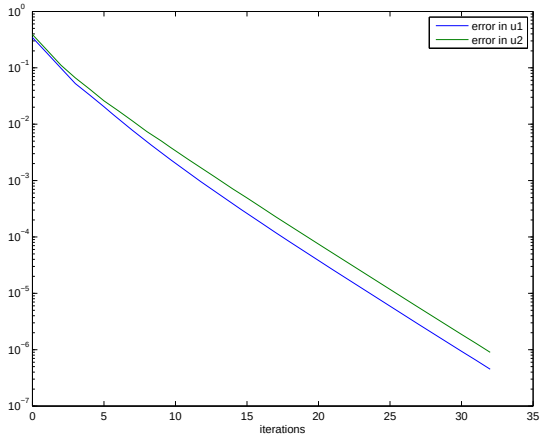


Fig. 2. Convergence history for the Belousov-Zhabotinsky equations.

4.2 FitzHugh-Nagumo Equations

The system of reaction diffusion equations

$$\begin{aligned} &\partial_t u_1 - \tfrac{1}{2}\partial_{xx} u_1 - f(u_1) + u_2 = 0, \\ &\partial_t u_2 - \tfrac{1}{2}\partial_{xx} u_2 - u_1 + u_2 = 0, \end{aligned}$$

with $f(u_1) = u_1 - u_1^3$ is called the FitzHugh-Nagumo equations, and describes how an action potential travels through a nerve. It is the prototype of an excitable system (e.g., a neuron) or an activator-inhibitor system: close to the ground state, one component stimulates the production of both components, while the other one inhibits their growth, see [9], p. 161 for details. This system does not satisfy the hypotheses of Theorem 1, but nevertheless we observe linear convergence, as shown in Fig. 3.

4.3 Lotka-Volterra Equations

The Lotka-Volterra equations with migration term are

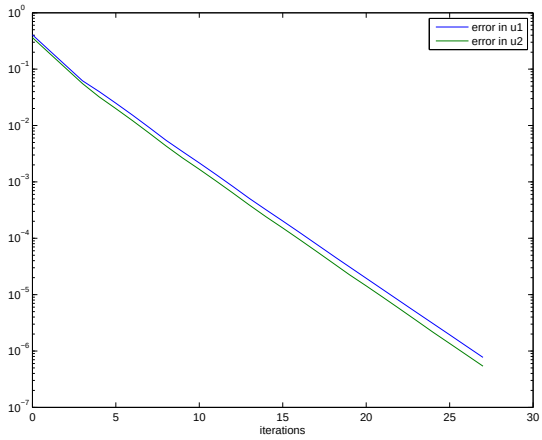


Fig. 3. Convergence history for the FitzHugh-Nagumo equations.

$$\partial_t u_1 - \tfrac{2}{25}\partial_{xx} u_1 - u_1(1-u_2) = 0,$$
$$\partial_t u_2 - \tfrac{2}{25}\partial_{xx} u_2 + u_2(1-u_1) = 0,$$

and they describe a biological predator-prey system, where both predator and prey are migrating randomly. This system does not satisfy the hypotheses of Theorem 1, and now we observe quite a different convergence behavior, as shown in Fig. 4.

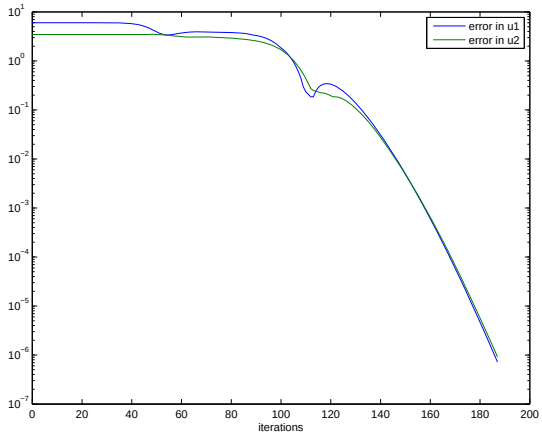


Fig. 4. Convergence history for the Lotka-Volterra equations with migration.

5 Conclusions

Schwarz waveform relaxation algorithms often exhibit superlinear convergence, as observed in the last example, see for example [2]. A corresponding convergence analysis requires however quite different techniques from the ones we have presented here, and will appear in an upcoming paper.

References

1. M.J. Gander. Overlapping Schwarz waveform relaxation for parabolic problems. In J. Mandel, C. Farhat, and X.-C. Cai, editors, *Tenth International Conference on Domain Decomposition Methods*. Contemporary Mathematics 218, AMS, Boulder, CO, 1998.
2. M.J. Gander. A waveform relaxation algorithm with overlapping splitting for reaction diffusion equations. *Numer. Linear Algebra Appl.*, 6:125–145, 1998.
3. M.J. Gander, L. Halpern, and F. Nataf. Optimal convergence for overlapping and non-overlapping Schwarz waveform relaxation. In C-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, editors, *Eleventh International Conference of Domain Decomposition Methods*. ddm.org, Bergen, 1999.
4. M.J. Gander, L. Halpern, and F. Nataf. Optimal Schwarz waveform relaxation for the one dimensional wave equation. *SIAM J. Numer. Anal.*, 41(5):1643–1681, 2003.
5. M.J. Gander and C. Rohde. Overlapping Schwarz waveform relaxation for convection dominated nonlinear conservation laws. *SIAM J. Sci. Comput.*, 27(2):415–439, 2005.
6. M.J. Gander and A.M. Stuart. Space time continuous analysis of waveform relaxation for the heat equation. *SIAM J. Sci. Comput.*, 19:2014–2031, 1998.
7. E. Giladi and H.B. Keller. Space time domain decomposition for parabolic problems. *Numer. Math.*, 93(2):279–313, 2002.
8. D. Henry. *Geometric Theory of Semilinear Parabolic Equations*, volume 840 of *Lecture Notes in Mathematics*. Springer, Berlin, 1981. ISBN 3-540-10557-3.
9. J.D. Murray. *Mathematical Biology*, volume 19 of *Biomathematics*. Springer, Berlin, 1989. ISBN 3-540-19460-6.
10. C.V. Pao. *Nonlinear Parabolic and Elliptic Equations*. Plenum Press, New York, NY, 1992. ISBN 0-306-44343-0.
11. A.I. Volpert, V.A. Volpert, and V.A. Volpert. *Traveling Wave Solutions of Parabolic Systems*, volume 140 of *Translations of Mathematical Monographs*. American Mathematical Society, Providence, RI, 1994. ISBN 0-8218-4609-4. Translated from the Russian manuscript by James F. Heyda.

A Sparse QS-Decomposition for Large Sparse Linear System of Equations

Wujian Peng[1] and Biswa N. Datta[2]

[1] Department of Math, Zhaoqing University, Zhaoqing, China,
douglas_peng@yahoo.com
[2] Department of Math, Northern Illinois University, Dekalb, IL, USA,
dattab@math.niu.edu

Summary. A direct solver for large scale sparse linear system of equations is presented in this paper. As a direct solver, this method is among the most efficient direct solvers available so far with flop count as $O(n\,logn)$ in one-dimensional situations and $O(n^{3/2})$ in second dimensional situation. This method has advantages over the existing fast solvers in which it can be used to handle more general situations, both well-conditioned or ill-conditioned systems; more importantly, it is a very stable solver and a naturally parallel procedure! Numerical experiments are presented to demonstrate the efficiency and stability of this algorithm.

1 Introduction

One of the core tasks in mathematical computations is to solve algebraic linear system of equations, which often arise from solving physical and engineering problems modeled in PDE and discretized by FEM or FDM. Frequently one may encounter systems with up to hundreds of thousands of unknowns and coefficient matrix is usually very sparse, i.e., most of the entries in the coefficient matrix are zeros.

Currently direct methods for solving large scale linear systems are variations of Gaussian elimination method (GE), such as TDMA, the multifrontal methods [1, 4], Pardiso [2] and superLU [5],etc. It is well-known that GE method is not stable due to the accumulated rounding errors, even if partial pivoting is applied; the excessive flops needed in GE (which is a part reason for the instability) is also an big issue. Though parallel can be partly implemented in reordering of unknowns [3, 6], a forward and a backward substituting procedure are always needed.

In this paper we are to discuss a complete parallel efficient algorithm for solving linear system of equations. This algorithm is based on the recognizing of certain special sequence of vectors, which is then orthogonalized by a special algorithm called Layered Group Orthogonalization (LGO). The new direct solver is based on LGO and is applied to solve linear systems with banded matrices arising from some one-dimensional problems. Numerical examples show some astonishing stability behavior of this new method. More applications of the idea presented here are to be followed by later papers.

2 A Quasi-Orthogonal Vector Sequence

We start from recognizing a special sequence of vectors, which are closely related to banded matrices.

Definition 1. *A sequence of vectors $v_1, v_2, \cdots, v_n$ is called quasi-orthogonal if there exists an integer m such that after grouping the above sequence into groups of m vectors*

$$\{v_1, \cdots v_m\}, \{v_{m+1} \cdots, v_{2m}\}, \cdots, \{v_{lm+1}, \cdots, v_n\}$$

there holds

$$v_i^T v_j = 0, \forall v_i \in G_{i'}, v_j \in G_{j'} \qquad (|i' - j'| > 1) \tag{1}$$

where G_i is the ith group containing vectors $\{v_{(i-1)m+1} \cdots, v_{im}\}$. Note the last group does not necessarily have m vectors. Furthermore, since any integer greater than m can be used to group the vector sequence while (1) is satisfied, we call this sequence as k-orthogonal sequence with k as the smallest integer m available.

Obviously a regular orthogonal sequence can be regarded as a quasi-orthogonal sequence, but a k-orthogonal sequence ($k > 0$) is not orthogonal. We are to present an efficient algorithm to orthogonalize a k-orthogonal sequence a moment later, but first let's give some examples of quasi-orthogonal sequences.

Example 1. Let $A \in \mathcal{R}$ be a bidiagonal matrix, $A = (a_1, a_2, \cdots a_n)$ be a partition with $a_i (i = 1, \cdots n)$ as its column vectors, then $a_1, a_2, \cdots a_n$ form a 1-orthogonal sequence. If one partitions A into a sequence of row vectors, one also gets a 1-orthogonal sequence.

Similarly one can see that the row/column vectors of a tridiagonal matrix form a 2-orthogonal sequence. As a matter of fact, given any banded matrix A with at most $k+1$ nonzero entries at each row/column, the associated row/column vectors always form a k-orthogonal sequence.

Note here we define k-sequence from vectors with only a few nonzero entries. Does there exist indeed vector sequence with all entries as nonzero numbers to form a k-orthogonal sequence? The following remark provides a positive answer to this question.

Remark 1. Let $Q \in \mathcal{R}^{n \times n}$ be an orthogonal matrix, $v_1, v_2, \cdots, v_n$ be a k-orthogonal sequence with each $v_i \in \mathcal{R}^n$, then sequence $Qv_1, Qv_2, \cdots, Qv_n$ also form a k-orthogonal sequence.

Outline of the proof Note that we have

$$(Qv_i)^T Qv_j = v_i^T Q^T Q v_j = v_i^T v_j = 0, \text{ for } |i' - j'| > 1,$$

where $v_i \in G_{i'}, v_j \in G_{j'}$, the conclusion thus comes directly from the above observation.

It is easy to see that any vector combinations inside each group will not change the k-orthogonal property of the sequence. As a direct consequence, the following statement is true and will be used in later section.

Remark 2. Let $G_1, G_2, \cdots G_l$ be a sequence of groups associated with a k-orthogonal sequence. If we orthogonalize vectors in each group, the newly formed groups of vector, still denoted by $G_1, G_2, \cdots G_l$, keep the k-orthogonal property.

3 Layered Group Orthogonalization

As we mentioned previously, the k-orthogonal sequence is not a regular orthogonal sequence, thus a process is needed to orthogonalize this sequence. However classical Gram-Schmidt process or Hessenberg is not suitable for this purpose since the sparsity of the vector (matrix) structure will be destroyed. Although Hessenberg QR factorization can be used to orthogonalize the sequence, its parallel implementations is not an easy task [7] since in essence QR factorization is a sequential algorithm

In this section we are to provide a new type of orthogonalization process especially designed for this type of quasi-orthogonal sequence, which is called Layered Group Orthogonalization process (LGO). The detail comes as follows.

3.1 Algorithm (LGO)

Let $v_1, v_2, \cdots, v_n$ be a k-orthogonal vector sequence and is grouped into groups $G_1, G_2, \cdots G_l$. Assuming each group has $m(\geq k)$ vectors.

Step 1: Orthogonalize all odd groups (these can be done independently), i.e., $G_1, G_3, \cdots G_l$ are orthogonalized at this step (assuming l is odd).

Step 2: If there are no even groups left, stop. Otherwise update each even group by making it orthogonal to its neighboring odd groups.

$$\bar{G}_{2i} = \bar{G}_{2i} - \bar{G}_{2i+1}\bar{G}_{2i+1}^T\bar{G}_{2i} - \bar{G}_{2i-1}\bar{G}_{2i-1}^T\bar{G}_{2i}$$

where $\bar{G}_j$ represents the matrix formed by vectors in vector group G_j.

Step 3: Take out all odd groups and renumber the groups left in the same order: $G_2, G_4, \cdots G_{2s} \longrightarrow G_1, G_2, \cdots, G_s$

Step 4: Go back to Step 1.

To justify the above algorithm, we need first the following conclusions.

Proposition 1 *The vector sequence formed by vectors in all odd groups after step 1 are orthogonal.*

Proof By the definition of k-orthogonality, vectors from different odd groups are orthogonal already. By Remark 2 and the fact that vectors in each odd group are orthogonalized in step 1, vectors from the same group are orthogonal to each other. This completes the proof.

Proposition 2 *The vectors in every even group are orthogonal to vectors in any odd groups after step 2. i.e., given any vector $v_i \in G_{2i'+1}$, $v_j \in G_{2j'}$, there holds $v_i^T v_j = 0$.*

The proof of this proposition is omitted .

Proposition 1 and 2 actually indicate this fact: "half" vectors in the original k-orthogonal sequence are orthogonalized in the first run of the loop in the above algorithm. To make the second run keeps similar orthogonalization function, we need the conclusions that follows immediately.

Proposition 3 *Vector sequence formed by vectors from even groups after second step in the above algorithm still form a k-orthogonal sequence.*

Proof It is omitted here.

3.2 Matrix representation of LGO

For the sake of simplicity, we assume k-orthogonal sequence $v_1, v_2, \cdots, v_n$ is grouped into $l+1$ groups $G_0, G_1, G_2, \cdots, G_l$ with each containing k vectors, we denote the first step operation in the algorithm by matrix $S_1^{(i)}$ and $S_2^{(i)}$ for the second step in each run of the loop ($i = 1, 2, \ldots$). Then one can see that

$$S_1^{(1)} = \begin{pmatrix} R_1^{-1} & & & & \\ & I_k & & & \\ & & R_3^{-1} & & \\ & & & I_k & \\ & & & & \ddots \end{pmatrix} \quad \text{and } S_2^{(1)} = \begin{pmatrix} I_k & -\hat{G}_1^T G_2 & & & \\ 0 & I_k & 0 & & \\ & -\hat{G}_3^T G_2 & I_k & -\hat{G}_3^T G_4 & \\ & & 0 & I_k & 0 \\ & & & & \ddots \end{pmatrix}$$

assuming $l+1$ is odd. Thus one can see that the first step in each run of the loop always corresponding to a block diagonal matrix with only a small portion of blocks are non-identity blocks, while the second step matrix has a small portion of columns having at most three non-zero blocks.

The whole process, denoted by matrix S^{-1}, has the following form: $S^{-1} = S_1^{(1)} S_2^{(1)} \cdots S_1^{(r)} S_2^{(r)}$, where $r = [\log(n/k)]$. Let Q be the resulted orthogonal matrix by LGO process, then one has $A = QS$, where A is the matrix formed by the k-orthogonal sequence as its column vectors. Note that both Q and S are sparse matrices with $O(k^2 n \log(n/k))$ nonzero entries. Actually the following graphs show their sparsity pattern.

4 LGO Solver and Numerical Experiments

An immediate application of the LGO factorization is of course to solve linear system $Ax = b$ with A as a banded matrix. Formerly one would have by multiplying both sides the inverse of matrix S,

$$S^{-1}Ax = S^{-1}b.$$

Note that $Q = S^{-1}A$ is an orthogonal matrix, thus the solution x can be written as

$$x = (S^{-1}A)^T S^{-1}b = Q^T S^{-1}b.$$

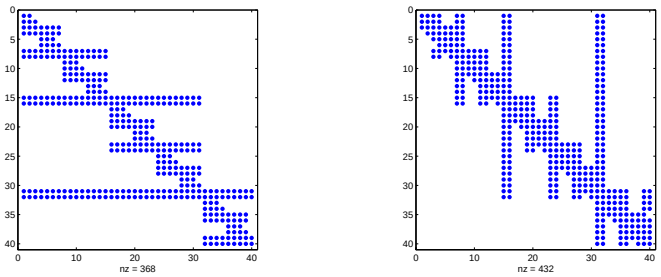


Fig. 1. Distribution of nonzeros of sparse matrix Q and S.

In actual calculations, the inverse of matrix S is not explicitly formed, and as we noted previously, the operation $S^{-1}A$ and $S^{-1}b$ only cost $O(k^2 n \log(n/k))$ flops.

Next we present results of some numerical experiments. We take the simplest banded matrix–tridiagonal matrix. As we know currently the most frequently used method is the so-called TDMA. Our comparison of LGO method and TDMA in case A is diagonally dominant shows that these two methods reach almost the same precision and the result are thus not listed here.

In case of A not diagonally dominant, we tried to use TDMA, UMFPACK and LGO to solve a system with A slightly away from diagonally dominant as follows:

$$A = tridiag\{-1.05, 2, -1\}$$

and the condition number of this matrix A grows from hundreds when n is less than 100 to $O(10^{13})$ when n is about 1,000.

In Fig. 2 one can see that both TDMA and UMFPACK work well in case of system with 1,000 unknowns, however when the size of system is greater than 1,500, the relative error becomes unacceptable. In order to test the stability of LGO, we

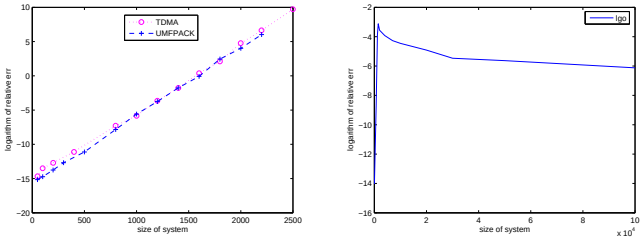


Fig. 2. Comparison of relative error by using TDMA, UMFPACK and LGO.

construct an extremely ill-conditioned linear system with A as a tridiagonal matrix with diagonal entries as 2, upper diagonal entries as -1 and lower diagonal entries run from 1 to $n-1$ with n as the size of the matrix. The following table is the calculated condition numbers of A when n varies.

n	10	15	20	25
cond (A)	9.3066e+004	5.0624e+008	4.7161e+013	3.9742e+017

Both TDMA and UMFPACK fail when $n = 30$. But by using LGO solver we successfully obtained pretty good solutions for n up to $100,000$ with the exact solution as $f(x) = x(1-x)e^{x+6}$ (see Fig. 3).

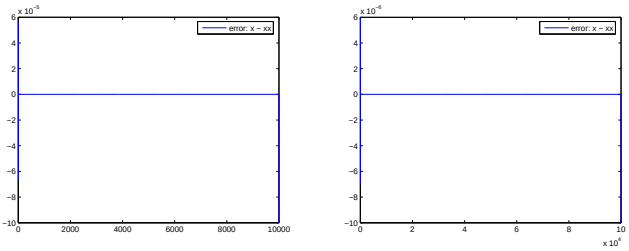


Fig. 3. Error between the exact solution and calculated solution using LGO.

5 A Nested Direct Domain Decomposition Idea

In this section we are going to briefly talk about another important application of the aforementioned direct solver, which is actually a generalization of LGO.

Based on our discovery of the quasi-orthogonal sequence introduced in previous section, we find that in FEM or FDM, each mesh node corresponding to a row in resulted coefficient matrix can be naturally grouped into different groups by its geometric locations; and if two groups are separated by at least a few grid lines (Fig. 4), then row vectors corresponding to these two groups can be orthogonalized independently. A simple implementation is described in the following.

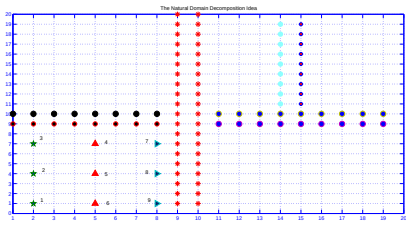


Fig. 4. Partitioning of direct domain decomposition.

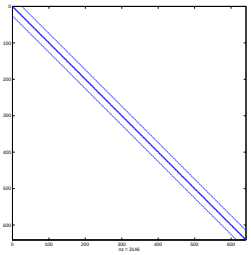


Fig. 5. Zero pattern of stiffness matrix.

A generalized LGO process is used to handle a second dimensional problem on $[0,1]^2$. Here we assume the domain is discretized with a rectangular mesh. In each layer of the LGO process we use small rectangular areas as the subdomains and the

subdomains in the upper layer are formed by grouping the neighboring rectangular subdomains in previous layer which share a common vertex. By using lexicography order of nodes the resulted coefficient matrix A is a banded matrix with zeros inside the band, as shown in Fig. 6.

The generalized LGO factors of A in this case have however some different zero patterns, as we can see from the following figures (Figs. 6, 7) that the counts for nonzeros are of order $O(n^{1.5})$, which can also be verified by estimating each step of the generalized LGO process.

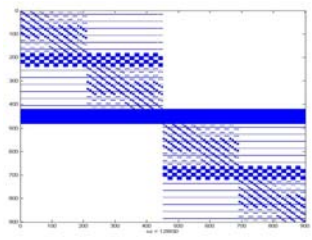


Fig. 6. Zero pattern of factor Q.

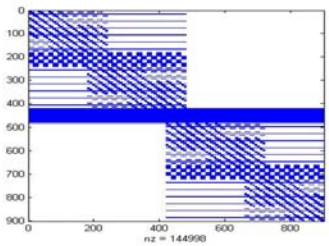


Fig. 7. Zero pattern of factor S.

Comparison between UMFPACK and LGO have been made on both well-conditioned and ill-conditioned systems from some simple two dimensional problems. Numerical experiments show that for standard Poisson problem defined on rectangular domain $[0,1]^2$, both methods work well (Fig. 8). However, for an artificially constructed discrete operator which leads to the following Toeplitz-like matrix $A = (a_{ij})_{n\times n}$ (very ill-conditioned) where

$$a_{ij} = \begin{cases} 4, & (i = j) \\ -6, & (i = j-1 \text{ and } (i \bmod l) \neq 1) \\ -2, & (i = j+1 \text{ and } (i \bmod l) \neq 0) \\ -1, & (|i-j| = l) \\ 0, & (\text{ otherwise}) \end{cases}$$

and l is the positive square root of system size n, UMFPACK fails when the size of system n is greater than a few thousands while LGO solver still shows satisfying relative error when n is more than $20,000$ (Fig. 9). More meaningful tests and applications of LGO are currently under the way and the results will be presented in later papers.

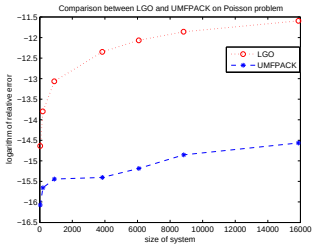


Fig. 8. Comparison on Poisson problem.

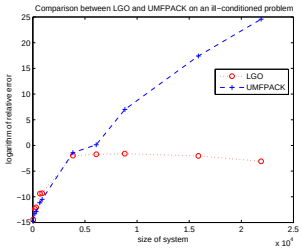


Fig. 9. Comparison on ill-conditioned problem.

Acknowledgement We would like to thank Professor Qun Lin for his great support while the first author was visiting LSEC. We also benefit from our discussion with Dr. Qiya Hu , Dr. Linbo Zhang and Dr. Zhongzhi Bai in LSEC as well as Dr. Jun Zou and Dr. Shuhua Zhang. We are encouraged to finish this part of research and gave a report at the conference.

References

1. I.S. Duff. A survey of sparse matrix research. In *Proc. IEEE*, 65(4): 500–535, 1977.
2. K. Gartner, W. Fichtner, and A. Stricker. Pardiso: A high-performance serial and parallel sparse linear solver in semiconductor device simulation. *J. Future Generation Comp. Syst.*, 18:69–78, 2001.
3. A. George. Nested dissection of a regular finite element mesh. *SIAM J. Numer. Anal.*, 10: 345–363, 1973.
4. A.M. Erisman, I.S. Duff and J.K. Reid. *Direct Methods for Sparse Matrices*. Oxford University Press, New York, NY, 1986.
5. J. Gilbert, J. Demmel and X. Li. An asynchronous parallel supernodal algorithm for sparse Gaussian elimination. *SIAM J. Matrix Anal. Appl.*, 20(4):915–952, 1999.
6. X.S. Li and J.W. Demmel. Making sparse Gaussian elimination scalable by static pivoting. In *Proceedings of the 1998 ACM/IEEE Conference on Supercomputing*, Orlando, FL, 1998.
7. J. Touriño, R. Doallo and E.L. Zapata. Sparse Householder qr factorization on a mesh. In *Proceedings of the Fourth Euromicro Workshop on Parallel and Distributed Processing*, Braga, 1998.

Is Additive Schwarz with Harmonic Extension Just Lions' Method in Disguise?

Felix Kwok

Section de mathématiques, Université de Genève, Geneva, Switzerland,
Felix.Kwok@unige.ch

Summary. The Additive Schwarz Method with Harmonic Extension (ASH) was introduced by Cai and Sarkis (1999) as an efficient variant of the additive Schwarz method that converges faster and requires less communication. We show how ASH, which is defined at the matrix level, can be reformulated as an iteration that bears a close resemblance to the parallel Schwarz method at the continuous level, provided that the decomposition of subdomains contains no cross points. In fact, the iterates of ASH are identical to the iterates of the discretized parallel Schwarz method outside the overlap, whereas inside the overlap they are linear combinations of previous Schwarz iterates. Thus, the two methods converge with the same asymptotic rate, unlike additive Schwarz, which fails to converge inside the overlap (Efstathiou & Gander 2007).

1 The Methods of Lions, AS, RAS and ASH

Let $\Omega \subset \mathbb{R}^n$ be a bounded open set. Suppose we want to solve the elliptic PDE

$$\mathcal{L}u = f \quad \text{on } \Omega, \qquad u = g \quad \text{on } \partial\Omega. \tag{1}$$

Based on the theoretical work of [8, 9] introduced the first domain decomposition methods for solving (1). In the two-subdomain case, let $\Omega_1, \Omega_2 \subset \Omega$ such that $\Omega_1 \cup \Omega_2 = \Omega$ and $\Omega_1 \cap \Omega_2 \neq \emptyset$. We also define $\Gamma_i = \partial\Omega \cap \bar{\Omega}_i$ and $\Gamma_{ij} = \partial\Omega_i \cap \bar{\Omega}_j$ for $i, j = 1, 2$. Then Lions' *parallel Schwarz method* calculates the subdomain iterates $u_i^k : \Omega_i \to \mathbb{R}$, $i = 1, 2$ via

$$\mathcal{L}u_i^{k+1} = f \quad \text{on } \Omega_i, \qquad u_i^{k+1} = g \quad \text{on } \Gamma_i, \qquad u_i^{k+1} = u_{3-i}^k \quad \text{on } \Gamma_{i,3-i}. \tag{2}$$

If we discretize the parallel Schwarz method (2), we obtain for $k = 0, 1, \ldots,$

$$A_1\mathbf{u}_1^{k+1} = f_1 - A_{12}\mathbf{u}_2^k, \qquad A_2\mathbf{u}_2^{k+1} = f_2 - A_{21}\mathbf{u}_1^k, \tag{3}$$

where $A_i = R_i A R_i^T$, $A_{ij} = (R_i A - A_i R_i) R_j^T$, and R_i restricts the set $V = \{1, \ldots, n\}$ of all nodes onto the subset V_i of nodes that lie in Ω_i. The above method

trivially generalizes to the case of many subdomains if there are no cross points, i.e., $\Omega_i \cap \Omega_j \cap \Omega_l = \emptyset$ for distinct i, j and l:

$$A_i \mathbf{u}_i^{k+1} = f_i - \sum_{j \neq i} A_{ij} \mathbf{u}_j^k, \qquad \text{for all } i. \tag{4}$$

Note that (4) does not define a global approximate solution $\mathbf{U}^k$ that is valid over the entire domain Ω. In fact, if the subdomains overlap, there is no unique way of defining $\mathbf{U}^k$ in terms of the $\mathbf{u}_j^k$ until the method has converged. Thus, one cannot directly consider parallel Schwarz as a preconditioner for the global system and use it in combination with Krylov subspace methods.

In order to turn parallel Schwarz into a preconditioner, [3] introduced the *additive Schwarz* (AS) method, which is equivalent to a block Jacobi iteration when the subsets V_j are disjoint. However, when the subdomains overlap, the method no longer converges inside the overlapping regions ([4, 6]). To obtain a convergent method, [2] introduced the methods of *Restricted Additive Schwarz* (RAS) and *Additive Schwarz with Harmonic Extension* (ASH), which are defined as follows: let $\tilde{\Omega}_j$ be a partition of Ω such that $\tilde{\Omega}_j \subset \Omega_j$. Let $\tilde{V}_j$ be the nodes that lie in $\tilde{\Omega}_j$, and $\tilde{R}_l$ be a matrix of the same size as R_l, such that

$$[\tilde{R}_l]_{ij} = \begin{cases} 1 & \text{if } [R_l]_{ij} = 1 \text{ and } j \in \tilde{V}_l, \\ 0 & \text{otherwise.} \end{cases}$$

Then, starting from an initial guess of the *global solution* $\mathbf{U}^0$, RAS calculates

$$\mathbf{U}^{k+1} = \mathbf{U}^k + \sum_j \tilde{R}_j^T A_j^{-1} R_j (f - A\mathbf{U}^k), \tag{5}$$

whereas ASH computes

$$\mathbf{U}^{k+1} = \mathbf{U}^k + \sum_j R_j^T A_j^{-1} \tilde{R}_j (f - A\mathbf{U}^k). \tag{6}$$

By restricting either the residual or the update to $\tilde{V}_j$, RAS and ASH avoids the redundant updates that occur within the overlap when Additive Schwarz is used. There exist other methods capable of eliminating the non-converging modes in AS, such as the method of Restricted Additive Schwarz with Harmonic Overlap (RASHO), which was proposed by [1].

It is clear that the RAS and ASH preconditioners are transposes of each other when A is symmetric; one thus expects the two methods to converge at a similar rate. In the case where A is an M-matrix, [5] proved that RAS and ASH both converge as an iterative method. For the RAS method, [6] showed that the iterates produced are equivalent to those of the discretized parallel Schwarz method, regardless of the number of subdomains and whether cross points are present. On the other hand, to our knowledge no such interpretation exists for the ASH method. Our goal is to offer such an interpretation in the case where cross points are absent.

2 Assumptions and the Main Result

Before stating the main result, we make some assumptions that are algebraic manifestations of the fact that there are no cross points. The first one is self-evident based on the definition of the restriction operators R_k.

Assumption 1 (No cross points) For distinct i, j and l, we have

$$R_i R_j^T R_j R_l^T = 0. \tag{7}$$

The next pair of assumptions ensures that $\partial\Omega_j \setminus \partial\Omega$ are partitioned into r connected components, each of which must be a subset of only one $\tilde{\Omega}_i$ for some i (see Fig. 1).

Assumption 2 (Partition of internal boundaries) For all $i \neq j$, we must have

$$(R_i - \tilde{R}_i)(AR_j^T - R_j^T A_j) = 0, \tag{8a}$$

$$(R_i A - A_i R_i)(R_j^T - \tilde{R}_j^T) = 0. \tag{8b}$$

The two conditions are simply transposes of each other; hence, they will be satisfied simultaneously if A has a symmetric nonzero pattern. Also note that when $i = j$, the two relations are trivially satisfied: since $\tilde{R}_i = \tilde{R}_i R_i^T R_i$, we have

$$(R_i - \tilde{R}_i)(AR_i^T - R_i^T A_i) = \underbrace{R_i A R_i^T}_{A_i} - \underbrace{R_i R_i^T}_{I} A_i - \tilde{R}_i R_i^T \underbrace{R_i A R_i^T}_{A_i} + \tilde{R}_i R_i^T A_i = 0. \tag{9}$$

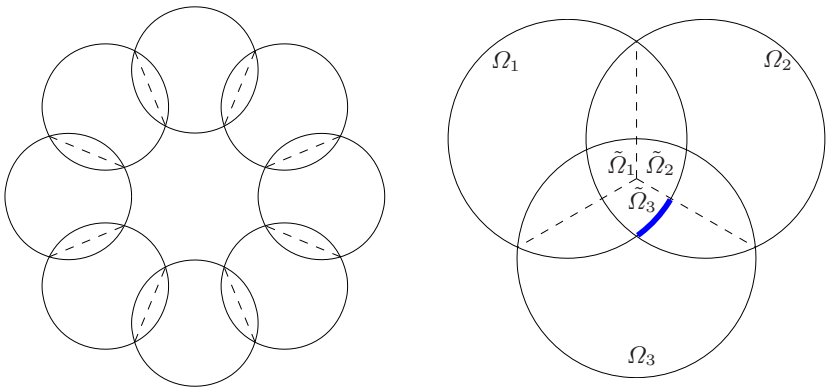


Fig. 1. Some examples of decompositions into subdomains, with solid lines delimiting Ω_i and dashed lines delimiting the $\tilde{\Omega}_i$. In (**a**), Assumptions 1–2 are satisfied, whereas in (**b**) they are not, because the stencil of $AR_1^T - R_1^T A_1$ along the thick portion of $\partial\Omega_1$ would extend into $\Omega_2 \setminus \tilde{\Omega}_2$, violating (8a).

We can interpret (8a) as follows. For any vector w over V_j, the vectors $AR_j^T w$ and $R_j^T A_j w$ agree inside V_j, but $AR_j^T w$ may have nonzero entries outside V_j (which $R_j^T A_j w$ cannot have). For a PDE, these entries are located along the boundary $\partial\Omega_j$. The assumption then says that these entries must correspond to nodes that are either inside $\tilde{\Omega}_i$ or completely outside Ω_i, as in Fig. 1(a). In Fig. 1(b), the thick portion of $\partial\Omega_1$ is inside $\Omega_2 \setminus \tilde{\Omega}_2$, violating (8a).

We are now ready to state our main result.

Theorem 1. *Suppose $\mathbf{U}^0 = 0$ and Assumptions 1 and 2 are satisfied. Then the iterates $\mathbf{U}^k$ of the ASH method are related to the iterates $\mathbf{v}_i^k$ of the discretized parallel Schwarz method with $\mathbf{v}_i^0 = 0$ and*

$$A_i \mathbf{v}_i^1 = \tilde{R}_i f, \tag{10}$$

$$A_i \mathbf{v}_i^k = R_i f - \sum_{j \neq i} A_{ij} \mathbf{v}_j^{k-1} \qquad (k \geq 2) \tag{11}$$

via the relation

$$\sum_{j=1}^{N} R_j^T \mathbf{v}_j^k = \mathbf{U}^k + \Big(\sum_j R_j^T R_j - I\Big)\mathbf{U}^{k-1}. \tag{12}$$

Remark Since $\sum_j R_j R_j^T - I$ is zero outside the overlap, it is clear that the iterates of ASH and Parallel Schwarz are identical outside the overlap, whereas inside they are linear combinations of the current and previous iterates.

3 Proof of the Main Result

We assume throughout this section that Assumptions 1 and 2 hold. We let

$$\mathbf{r}^k := f - A\mathbf{U}^k, \qquad \delta\mathbf{u}_j^k := A_j^{-1}\tilde{R}_j(f - A\mathbf{U}^k), \qquad \mathbf{u}_j^k := \sum_{l=0}^{k-1} \delta\mathbf{u}_j^l.$$

From (6) it is clear that $\mathbf{U}^k = \sum_j R_j^T \mathbf{u}_j^k$. We also define $\mathbf{v}_j^k$ such that $\mathbf{v}_j^0 = \mathbf{u}_j^0 = 0$, $\mathbf{v}_j^1 = \mathbf{u}_j^1$, and

$$\mathbf{v}_j^k = \mathbf{u}_j^k + \sum_{i \neq j} R_j R_i^T \mathbf{u}_i^{k-1} \qquad (k \geq 2). \tag{13}$$

The following properties are elementary and will be used repeatedly:

(i) $R_i R_i^T = I$ for all i,
(ii) $\tilde{R}_i^T R_i = R_i^T \tilde{R}_i = \tilde{R}_i^T \tilde{R}_i$ for all i,
(iii) $\tilde{R}_i R_i^T = R_i \tilde{R}_i^T = \tilde{R}_i \tilde{R}_i^T$ for all i,
(iv) $\sum_j \tilde{R}_j^T \tilde{R}_j = I$.

Lemma 1. *For all $k \geq 1$ and for all i, we have $(R_i - \tilde{R}_i)\mathbf{r}^k = 0$.*

Proof. Fix i and let $k \geq 0$. We calculate

$$(R_i - \tilde{R}_i)\mathbf{r}^k = R_i R_i^T (R_i - \tilde{R}_i)\mathbf{r}^k = R_i \Big[\sum_j \tilde{R}_j^T \tilde{R}_j R_i^T (R_i - \tilde{R}_i)\mathbf{r}^k\Big].$$

Since $R_j^T R_j$ and $R_i^T(R_i - \tilde{R}_i)$ are diagonal matrices, they commute, and hence

$$(R_i - \tilde{R}_i)\mathbf{r}^k = R_i \Big[\sum_j R_i^T (R_i - \tilde{R}_i)\tilde{R}_j^T \tilde{R}_j \mathbf{r}^k\Big] = R_i R_i^T (R_i - \tilde{R}_i) \sum_j \tilde{R}_j^T \tilde{R}_j \mathbf{r}^k.$$

Noting that $R_i R_i^T = I$ and $\tilde{R}_j^T \tilde{R}_j = R_j^T \tilde{R}_j$, we can rewrite $\tilde{R}_j \mathbf{r}^k$ as $A_j \delta \mathbf{u}_j^k$. Then (8a) and (9) together give

$$\begin{aligned}(R_i - \tilde{R}_i)\mathbf{r}^k &= (R_i - \tilde{R}_i) \sum_j A R_j^T \delta \mathbf{u}_j^k \\ &= (R_i - \tilde{R}_i) A(\mathbf{U}^{k+1} - \mathbf{U}^k) = (R_i - \tilde{R}_i)(\mathbf{r}^k - \mathbf{r}^{k+1}).\end{aligned}$$

Cancelling $(R_i - \tilde{R}_i)\mathbf{r}^k$ from both sides gives the required result.

Proof of Theorem 1 We first prove the relation (12). We have

$$\begin{aligned}\sum_j R_j^T \mathbf{v}_j^k &= \sum_j R_j^T \mathbf{u}_j^k + \sum_j R_j^T \sum_{l \neq j} R_j R_l^T \mathbf{u}_l^{k-1} \\ &= \mathbf{U}^k + \sum_j R_j^T R_j \Big(\sum_l R_l^T \mathbf{u}_l^{k-1} - R_j^T \mathbf{u}_j^{k-1}\Big) \\ &= \mathbf{U}^k + \Big(\sum_j R_j^T R_j\Big)\mathbf{U}^{k-1} - \sum_j R_j^T \mathbf{u}_j^{k-1} \\ &= \mathbf{U}^k + \Big(\sum_j R_j^T R_j - I\Big)\mathbf{U}^{k-1},\end{aligned}$$

as required. Now let $A_{i\Gamma} := R_i A - A_i R_i$ be the boundary operator. Multiplying both sides of (12) by $A_{i\Gamma}$ on the left gives

$$\begin{aligned}A_{i\Gamma} \sum_j R_j^T \mathbf{v}_j^k &= A_{i\Gamma}\mathbf{U}^k + A_{i\Gamma}\Big(\sum_j R_j^T R_j - I\Big)\mathbf{U}^{k-1} \\ &= A_{i\Gamma}\mathbf{U}^k + A_{i\Gamma}\Big(\sum_j (R_j^T R_j - \tilde{R}_j^T R_j)\Big)\mathbf{U}^{k-1} \\ &= A_{i\Gamma}\mathbf{U}^k + A_{i\Gamma}\Big(\sum_j (R_j - \tilde{R}_j)^T R_j\Big)\mathbf{U}^{k-1} = A_{i\Gamma}\mathbf{U}^k,\end{aligned}$$

since $A_{i\Gamma}(R_j - \tilde{R}_j)^T = 0$ for all i and j by (8b) and (9). When $i \neq j$, $A_{i\Gamma} R_j^T = A_{ij}$ by definition, and when $i = j$, we have

$$A_{i\Gamma} R_i^T = (R_i A - A_i R_i) R_i^T = R_i A R_i^T - A_i R_i R_i^T = A_i - A_i \cdot I = 0.$$

So in fact we have

$$A_{i\Gamma}\mathbf{U}^k = A_{i\Gamma}\sum_j R_j^T\mathbf{v}_j^k = \sum_{j\neq i} A_{ij}\mathbf{v}_j^k. \tag{14}$$

We now prove the main statement of the theorem. For $k \geq 0$, we have

$$\begin{aligned}
A_i\mathbf{v}_i^{k+1} &= A_i\mathbf{u}_i^{k+1} + A_i\sum_{j\neq i} R_iR_j^T\mathbf{u}_j^k \\
&= A_i\delta\mathbf{u}_i^k + A_i\mathbf{u}_i^k + A_i\sum_{j\neq i} R_iR_j^T\mathbf{u}_j^k \\
&= \tilde{R}_i(f - A\mathbf{U}^k) + A_i\sum_{j} R_iR_j^T\mathbf{u}_j^k \\
&= \tilde{R}_if - \tilde{R}_iA\mathbf{U}^k + A_iR_i\mathbf{U}^k.
\end{aligned}$$

If $k = 0$, then all terms other than $\tilde{R}_if$ vanish because $\mathbf{U}^0 = 0$; we have thus proved (10). We continue by assuming $k \geq 1$:

$$\begin{aligned}
A_i\mathbf{v}_i^{k+1} &= \tilde{R}_if - \tilde{R}_iA\mathbf{U}^k + (R_iA - A_{i\Gamma})\mathbf{U}^k \\
&= \tilde{R}_if + (R_i - \tilde{R}_i)A\mathbf{U}^k - A_{i\Gamma}\mathbf{U}^k \\
&= \tilde{R}_if + (R_i - \tilde{R}_i)(f - \mathbf{r}^k) - A_{i\Gamma}\mathbf{U}^k \\
&= R_if - \underbrace{(R_i - \tilde{R}_i)\mathbf{r}^k}_{=0} - \sum_{j\neq i} A_{ij}\mathbf{v}_j^k.
\end{aligned}$$

by Lemma 1 and (14), and (11) follows.

4 Convergence Rate

Given the close relationship between ASH and Parallel Schwarz, one would expect that the two methods converge at the same speed. This is true if the overlap subproblem is well posed.

Theorem 2 (cf. [7]). *Let R_o be the restriction operator onto the union of all overlaps, i.e., R_o is a full row-rank matrix such that $R_o^TR_o = \sum_j R_j^TR_j - I$. If $R_oAR_o^T$ is non-singular, then the ASH method* (6) *converges if and only if the parallel Schwarz method* (10), (11) *converges. In addition, when both methods converge, they do so at the same asymptotic rate.*

To illustrate this theorem, we solve Poisson's equation on the unit square with homogeneous Dirichlet boundary condition. We use a 20×20 grid, which is divided into two subdomains with a two-row overlap (Fig. 2(a)). We then solve this problem using (i) the discrete Parallel Schwarz method with Dirichlet boundary conditions, as defined in Eq. (3), (ii) the ASH method, and (iii) overlapping Additive Schwarz.

The convergence history for all three methods is shown in Fig. 2(b). We see that ASH converges linearly, unlike AS, which does not converge because of the overlap. In addition, the curves for Parallel Schwarz and ASH are very close to one another, and their slopes are asymptotically equal. We also see from Table 1 that the ratio of successive errors for parallel Schwarz alternates between 0.5737 and 0.5895 (which is typical for a two-subdomain problem), whereas ASH converges at a rate of $0.5815 = \sqrt{0.5737 \cdot 0.5895}$, the geometric mean. Thus, as iterative methods, ASH and parallel Schwarz converge at the same rate, as stated in Theorem 2.

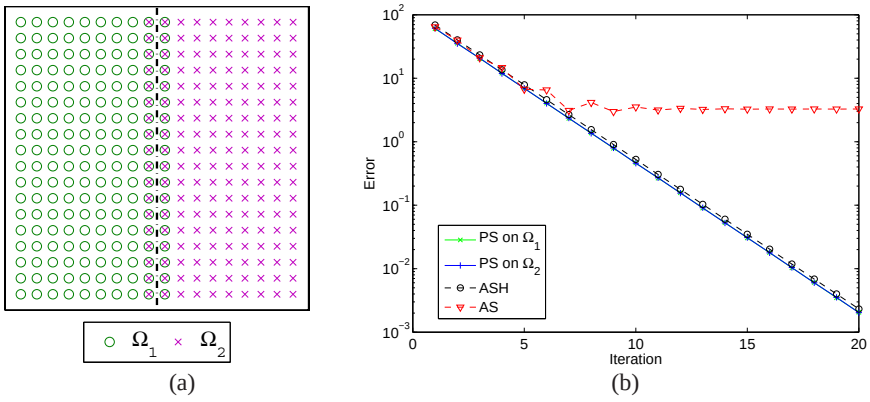


Fig. 2. (**a**) A two-subdomain decomposition. (**b**) Convergence behaviour for the parallel Schwarz, additive Schwarz and ASH methods.

Table 1. Error norms for Parallel Schwarz and ASH iterates.

	Parallel Schwarz				ASH	
	Error on Ω_1		Error on Ω_2		Error on Ω	
Iters	L_2-Norm	Ratio	L_2-Norm	Ratio	L_2-Norm	Ratio
1	60.4103		61.2159		69.0920	
2	35.5724	0.5888	35.0937	0.5733	40.1592	0.5812
3	20.4046	0.5736	20.6840	0.5894	23.3519	0.5815
4	12.0281	0.5895	11.8656	0.5737	13.5796	0.5815
5	6.9001	0.5737	6.9947	0.5895	7.8969	0.5815
6	4.0676	0.5895	4.0126	0.5737	4.5923	0.5815
7	2.3335	0.5737	2.3654	0.5895	2.6706	0.5815
8	1.3756	0.5895	1.3570	0.5737	1.5530	0.5815
9	0.7891	0.5737	0.7999	0.5895	0.9031	0.5815
10	0.4652	0.5895	0.4589	0.5737	0.5252	0.5815

5 Conclusions

We have shown that when the domain decomposition contains no cross points, the ASH method and parallel Schwarz have identical iterates outside the overlap. When both methods converge, they do so at the same asymptotic rate, provided the overlap subproblem is well posed. Thus, ASH is simply Lions' parallel method disguised as a preconditioner. The same conclusions hold when optimized transmission conditions are used; the proof is given in a separate article [7]. Such an insight can be used to estimate convergence rates of the optimized ASH method in cases where known results (e.g. [5]) do not apply. It would be interesting to see whether similar ideas can be used to relate RASHO to the parallel Schwarz method. Finally, a crucial assumption throughout this paper is that there must be no cross points. Since no such assumption is required in the interpretation of RAS [6], it would be instructive to study a problem with cross points to see whether similar results hold for arbitrary domain decompositions.

References

1. X.-C. Cai, M. Dryja, and M. Sarkis. Restricted additive Schwarz preconditioners with harmonic overlap for symmetric positive definite linear systems. *SIAM J. Numer. Anal.*, 41 (4):1209–1231 (electronic), 2003.
2. X.-C. Cai and M. Sarkis. A restricted additive Schwarz preconditioner for general sparse linear systems. *SIAM J. Sci. Comput.*, 21(2):792–797 (electronic), 1999.
3. M. Dryja and O.B. Widlund. Some domain decomposition algorithms for elliptic problems. In *Iterative Methods for Large Linear Systems (Austin, TX, 1988)*, pp. 273–291. Academic Press, Boston, MA, 1990.
4. E. Efstathiou and M.J. Gander. Why restricted additive Schwarz converges faster than additive Schwarz. *BIT*, 43(suppl):945–959, 2003.
5. A. Frommer and D.B. Szyld. An algebraic convergence theory for restricted additive Schwarz methods using weighted max norms. *SIAM J. Numer. Anal.*, 39(2):463–479 (electronic), 2001.
6. M.J. Gander. Schwarz methods in the course of time. *ETNA*, 31:228–255, 2008.
7. F. Kwok. Optimized additive Schwarz with harmonic extension as a discretization of continous parallel Schwarz methods, SINUM, 2010.
8. P.-L. Lions. On the Schwarz alternating method. I. In *First International Symposium on Domain Decomposition Methods for Partial Differential Equations (Paris, 1987)*, pp. 1–42. SIAM, Philadelphia, PA, 1988.
9. H.A. Schwarz. Über einen Grenzübergang durch alternierendes Verfahren. *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich*, 15:272–286, 1870.

Domain Decomposition Methods for a Complementarity Problem*

Haijian Yang[1] and Xiao-Chuan Cai[2]

[1] College of Mathematics and Econometrics, Hunan University, Changsha, Hunan, 410082, P. R. China, haijian.yang@colorado.edu
[2] Department of Computer Science, University of Colorado, Boulder, CO 80309, USA, cai@cs.colorado.edu

Summary. We introduce a family of parallel Newton-Krylov-Schwarz methods for solving complementarity problems. The methods are based on a smoothed grid sequencing method, a semismooth inexact Newton method, and a two-grid restricted overlapping Schwarz preconditioner. We show numerically that such an approach is highly scalable in the sense that the number of Newton iterations and the number of linear iterations are both nearly independent of the grid size and the number of processors. In addition, the method is not sensitive to the sharp discontinuity that is often associated with obstacle problems. We present numerical results for some large scale calculations obtained on machines with hundreds of processors.

1 Introduction

Complementarity problems have many important applications [3, 4, 6], and there are growing interests in developing efficient parallel algorithms for solving these semismooth problems on large scale supercomputers. One popular approach is the class of semismooth methods which solves the complementarity problem by first reformulating it as a semismooth system of equations and then applying a generalized Newton method to solve this system. There are extensive theoretical and numerical results associated with this approach, see, e.g., [5, 7, 8]. However, all existing approaches seem to have scalability problems in the sense that when the degree of freedoms in the problem increases the number of nonlinear or linear iterations increases drastically.

In this paper, we introduce a class of general purpose two-grid Newton-Krylov-Schwarz (NKS) algorithms for complementarity problems associated with partial differential equations. The methods are based on an inexact semismooth Newton method, a smoothed grid sequencing method and a two-level cascade restricted

* The research was supported in part by DOE under DE-FC-02-06ER25784, and in part by NSF under grants CCF-0634894 and CNS-0722023.

overlapping Schwarz preconditioning technique. It turns out, with an appropriate grid sequencing, the convergence rate of the semismooth Newton method can be made nearly independent of the number of unknowns of the system using either the Fischer-Burmeister function or the minimum function. Using the two-level restricted Schwarz preconditioner with sufficient overlap, the number of linear iterations also becomes nearly independent of the number of unknowns of the system. More importantly, both the linear and nonlinear iterations are nearly independent of the number of processors in our numerical experiments on machines with hundreds of processors.

2 Semismooth Function Approaches for Complementarity Problems

Let $\Omega \in R^2$ be a bounded open domain on which a linear or nonlinear differential operator $L(u)$ is defined. Many problems can be described as finding a function $u(x)$ defined in certain space such that

$$\begin{cases} L\mathrm{u}(x) \geq 0, & x \in \Omega \\ \mathrm{u}(x) \geq \Phi, & x \in \Omega \\ (\mathrm{u}(x) - \Phi) L\mathrm{u}(x) = 0, & x \in \Omega \end{cases} \tag{1}$$

with some boundary conditions assumed for $u(x), x \in \partial\Omega$. Here Φ is given and often called an obstacle. In this paper, we consider the following complementarity problem:

$$\begin{array}{ll} \text{find} & u_h \in R^n, \\ \text{such that} & u_h \geq \phi, \quad F(u_h) \geq 0, \quad (u_h - \phi)^T F(u_h) = 0, \end{array} \tag{2}$$

where $F(u_h) = (F_1(u_h), \cdots, F_n(u_h))^T : R^n \to R^n$ denotes a continuously differentiable function from the discretized version of $L(u)$, and $\phi \in R^n$ denotes the obstacle from the discretization of Φ.

2.1 Semismooth Newton Methods

Let $a_i = (u_h - \phi)_i$ and $b_i = F_i(u_h)$, the reformulations of the complementarity problem based on the Fischer-Burmeister function [5] and the minimum function [7] are as follows:

$$\mathcal{F}_{FB}(a, b) := a + b - \sqrt{a^2 + b^2} = 0, \tag{3}$$

$$\mathcal{F}_{MIN}(a, b) := \min\{a, b\} = 0. \tag{4}$$

In fact, the Fischer-Burmeister function is differentiable everywhere except at the point $(a, b) = (0, 0)$, while the minimum function is piecewise smooth with its nondifferentiable points forming the line $\{(a, b)^T \in R^2 : a = b\}$.

If we apply a Newton-type method to (3) and (4), respectively, then it leads to the class of inexact semismooth Newton methods, in which we need to solve a right-preconditioned Jacobian system

$$\|\mathcal{F}(u_h^k) + J_k M_k^{-1}(M_k s_k)\| \leq \max\{\eta_r \|\mathcal{F}(u_h^k)\|, \eta_a\},$$

where J_k is a generalized Jacobian of $\mathcal{F}(u_h^k)$ to be introduced below, $\eta_r \in [0,1)$ is a relative tolerance, $\eta_a \in [0,1)$ is an absolute tolerance, and M_k^{-1} is an overlapping Schwarz preconditioner [9, 10].

For both the Fischer-Burmeister function and the minimum function, the generalized Jacobian matrix J_k is of the form

$$J_k = D_a^k + D_b^k F'(u_h^k) \tag{5}$$

with diagonal matrices (depending on the iteration index k)

$$D_a^k = \text{diag}(d_{a_1}, \ldots, d_{a_n}), \quad D_b^k = \text{diag}(d_{b_1}, \ldots, d_{b_n}). \tag{6}$$

The values of D_a^k and D_b^k in (6) corresponding to the Fischer-Burmeister function take the form

$$d_{a_i} := \begin{cases} 1 - a_i/\sqrt{a_i^2 + b_i^2}, & \text{if } a_i^2 + b_i^2 \neq 0, \\ 1, & \text{if } a_i^2 + b_i^2 = 0, \end{cases}$$

and

$$d_{b_i} := \begin{cases} 1 - b_i/\sqrt{a_i^2 + b_i^2}, & \text{if } a_i^2 + b_i^2 \neq 0, \\ 1, & \text{if } a_i^2 + b_i^2 = 0. \end{cases}$$

Similarly, when using the minimum function, the values of D_a^k and D_b^k in (6) assume the form

$$d_{a_i} := \begin{cases} 1, & a_i < b_i, \\ 0, & a_i \geq b_i, \end{cases}$$

and

$$d_{b_i} := \begin{cases} 0, & a_i < b_i, \\ 1, & a_i \geq b_i. \end{cases}$$

When using a Newton type method to solve complementarity problems, one of the major problems is the deterioration of the convergence rate when the mesh is refined. We here propose a smoothed grid sequencing technique: First, compute the solution u_H^* of the nonlinear system $\mathcal{F}_H(u_H) = 0$ on a coarse grid. Second, interpolate the coarse solution to obtain $\tilde{u}_h^0 = I_H^h u_H^*$, which is then smoothed by replacing its value at each grid point with the following weighed average of its neighboring values:

$$\begin{array}{ccccc} \frac{1}{16} & & \frac{1}{8} & & \frac{1}{16} \\ & \searrow & \downarrow & \swarrow & \\ \frac{1}{8} & \rightarrow & \frac{1}{4} & \leftarrow & \frac{1}{8} \\ & \nearrow & \uparrow & \nwarrow & \\ \frac{1}{16} & & \frac{1}{8} & & \frac{1}{16}. \end{array}$$

The smoothed vector is then used as the initial guess for the fine grid Newton iteration. In Fig. 1, we show the surface plots of the nonlinear system $\mathcal{F}_h(I_H^h u_H^*)$ on a fine grid without smoothing (*left figure*), and $\mathcal{F}_h(u_H^0)$ with one sweep of smoothing (*right figure*) for an obstacle problem. More details of this problem will be discussed in the numerical experiments section.

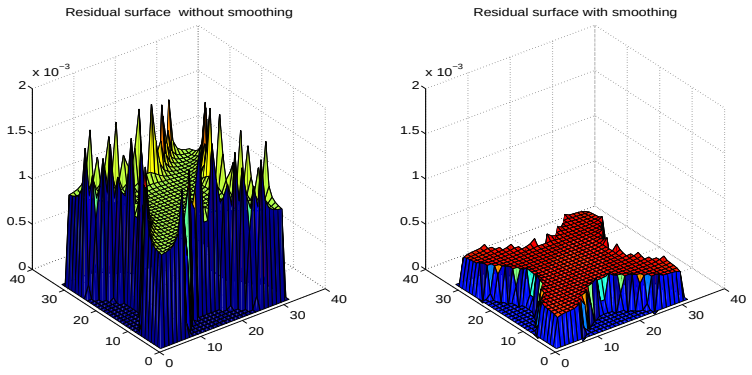


Fig. 1. The effect of smoothing of the interpolated coarse grid solution on the fine grid.

2.2 Schwarz Preconditioner

Let J be the Jacobian matrix on the fine grid and R_i^δ and R_i^0, the restriction operator from Ω to its overlapping and non-overlapping subdomains, respectively. Then the one-level restricted additive Schwarz (RAS) preconditioner [2] is

$$M_{RAS}^{-1} = \sum_{i=1}^{N_s} (R_i^0)^T J_i^{-1} R_i^\delta. \tag{7}$$

with $J_i = R_i^\delta \; J \; (R_i^\delta)^T$ and N_s is the number of subdomains, which is the same as the number of processors.

Let J_c be the Jacobian matrix on the coarse grid and I_h^H a restriction operator from the fine grid to the coarse grid. Then the two-level restricted Schwarz preconditioner is

$$M^{-1} = M_c^{-1} + M_{RAS}^{-1} - M_{RAS}^{-1} J M_c^{-1}$$

with $M_c^{-1} = (I_h^H)^T J_c^{-1} I_h^H$. We refer to [9, 10] for further analysis and examples of Schwarz preconditioning techniques.

3 Numerical Experiments

We report some results of our numerical experiments. Our solver is implemented using PETSc ([1]). We consider an obstacle problem: find $u(x)$ such that

$$\begin{cases} -\Delta u(x) + C \geq 0, & x \in \Omega, \\ u(x) \geq -\mathrm{d}(x, \partial\Omega), & x \in \Omega, \\ (u(x) + \mathrm{d}(x, \partial\Omega))(-\Delta u(x) + C) = 0, & x \in \Omega, \\ u(x) = 0, & x \in \partial\Omega, \end{cases} \tag{8}$$

where the $\mathrm{d}(x, \partial\Omega)$-operator measures the distance from a point x to the domain boundary $\partial\Omega$, and the parameter $C = 5$.

For the discretization we use the standard second-order five-point finite difference method on a uniform grid. The initial guess u_h^0 for the global Newton iteration is the obstacle from the discretization of $-\mathrm{d}(x, \partial\Omega)$ in (8). We stop the fine grid Newton iteration if

$$\|\mathcal{F}(u_h^k)\| \leq \max\{10^{-6}\|\mathcal{F}(u_h^0)\|, 10^{-10}\}.$$

The fine grid Jacobian system is solved with GMRES (30), and the iteration is stopped if the tolerance

$$\|\mathcal{F}(u_h^k) + J_k s_k\| \leq \max\{10^{-4}\|\mathcal{F}(u_h^k)\|, 10^{-10}\}$$

is satisfied. The subdomain problems are solved with LU factorization. Throughout this section,"np" stands for the number of processors which is the same as the number of subdomains, "INB" the number of inexact Newton iterations, "RAS" the number of RAS preconditioned GMRES iterations, and "Time" of the total computer time in seconds.

3.1 One-Level Results

We first study the one-level method with overlap $\delta = 3$. As shown in Table 1, on a fixed grid, the number of Newton iterations is independent of the number of processors, but the number of GMRES iterations increases as the number of processors increases for both the Fischer-Burmeister function and the minimum function. The major problem with the one-level method shows up, if we look at the scalability for a fixed number of processors. For each row in the table, every time we refine the grid by a factor of 2, the number of Newton iterations doubles. This problem prohibits the use of the method for high resolution applications.

3.2 Two-Level Results

In this subsection, we present the numerical results using the two-level approach in which a coarse grid is used in the nonlinear solver for generating a better initial guess and also in the linear solver for generating part of the Schwarz preconditioner. In the test, the initial guess for the global Newton iteration on the coarse grid is the obstacle ϕ in (2), and the tolerance conditions on the fine grid are the same as in the one-level method. We stop the coarse grid Newton iteration if

$$\|\mathcal{F}_H(u_H^k)\| \leq \max\{10^{-4}\|\mathcal{F}_H(u_H^0)\|, 10^{-10}\}.$$

Table 1. Results for the one-level method with overlap $\delta = 3$.

Mesh	256×256			512×512			$1,024 \times 1,024$			$2,048 \times 2,048$		
np	INB	RAS	Time	INB	RAS	Time	INB	RAS	Time	INB	RAS	Time
The Fischer-Burmeister function												
64	82	11.4	3.3	162	14.7	32.5	320	19.1	384.1	639	24.4	4781.1
128	82	13.6	2.1	162	17.5	17.4	320	22.2	180.4	639	30.8	2236.3
256	82	14.4	1.5	162	18.9	10.2	320	24.3	95.9	639	34.1	1110.0
512	82	17.2	1.1	162	22.6	7.5	320	32.3	62.3	639	38.5	568.9
The minimum function												
64	80	11.7	2.9	159	15.3	29.5	319	19.9	361.9	637	26.4	4673.0
128	80	14.0	1.9	159	18.3	16.1	319	23.7	173.5	637	33.7	2201.8
256	80	14.9	1.4	159	19.7	9.7	319	26.1	94.4	637	36.5	1104.8
512	80	17.7	1.3	159	23.8	7.3	319	34.5	62.0	637	41.1	567.5

In the test, the Jacobian system on the coarse grid is solved with a one-level RAS preconditioned GMRES (30) with the following stopping condition

$$\|\mathcal{F}_H(u_H^k) + J_H^k M_{H,RAS}^{-1}(M_{H,RAS} s_k)\| \leq \max\{10^{-4}\|\mathcal{F}_H(u_H^k)\|, 10^{-10}\},$$

where $M_{H,RAS}^{-1}$ is defined similar to (7) on the coarse grid. The subdomain problems are solved with LU factorization.

Using $\delta = 6$ and $\delta_c = 3$, we solve the test problem on several different fine grids with the two-level method and the results are summarized in Table 2, for both the Fischer-Burmeister function and the minimum function. The main concern is the size of the coarse grid H, which is taken as $h/2$, $h/4$ and $h/8$, where h is the size of the fine grid. In terms of the total number of Newton iterations, $H = h/2$ is certainly the best, but $H = h/8$ offers the best results in terms of the total compute time. Note that some cases, marked as "$*$", for the fine grid 256×256 are not available because the corresponding coarse grids are too coarse and the coarse Newton may not converge. The compute time includes the coarse grid calculation of the initial guess, the smoothing of the coarse solution, and the solving of the fine grid problem. Note that the minimum function approach is always faster than the Fischer-Burmeister function approach in terms of all measures.

We should mention that the use of smoothed grid sequencing plays an important role in the two-level methods. In Fig. 1, the surface plots of the residual function before and after the smoothing are shown and they are quite different. The cost of the smoothing step is very small and fewer number of Newton iterations is needed as a result of the smoothing.

4 Some Final Remarks

We have developed a family of parallel, highly scalable, two-grid algorithms for solving general complementarity problems. In addition to the fine grid, on which

Table 2. Results with different fine and coarse grids. The overlapping sizes of the coarse grid and the fine grid are $\delta_c = 3$ and $\delta = 6$, respectively. The preconditioner is the two-level RAS. h and H are the fine and coarse grid sizes, respectively.

Mesh	256×256			512×512			$1{,}024 \times 1{,}024$			$2{,}048 \times 2{,}048$		
np	INB	RAS	Time	INB	RAS	Time	INB	RAS	Time	INB	RAS	Time
The Fischer-Burmeister function												
$H = h/2$												
64	6	10.8	2.4	5	15.8	11.5	4	21.8	76.0	4	26.3	848.5
128	6	13.0	2.2	5	18.8	8.5	4	26.3	49.0	4	35.8	536.1
256	6	13.8	1.8	5	20.8	6.0	4	30.0	33.6	4	37.8	291.9
512	6	19.3	2.8	5	24.4	5.9	4	34.6	32.9	4	43.0	206.5
$H = h/4$												
64	$*$			7	15.3	6.2	7	19.0	33.9	6	25.8	201.7
128	$*$			7	18.4	4.8	7	22.6	21.9	6	32.7	120.8
256	$*$			7	20.3	3.6	7	25.4	25.4	6	38.7	71.6
512	$*$			7	23.9	4.3	7	33.9	12.1	6	43.7	56.6
$H = h/8$												
64	$*$			9	15.7	5.9	9	19.7	31.0	8	26.4	169.9
128	$*$			9	19.1	4.6	9	23.7	18.8	8	33.9	99.2
256	$*$			9	21.0	3.5	9	26.6	10.7	9	36.6	54.3
512	$*$			$*$			9	34.3	10.8	8	45.3	34.4
The minimum function												
$H = h/2$												
64	2	14.0	1.3	3	14.3	8.3	3	17.7	60.4	2	31.5	777.7
128	2	16.5	1.2	3	16.7	6.0	3	21.7	40.6	2	39.0	446.3
256	2	17.5	1.0	3	18.3	4.3	3	26.3	26.3	2	49.5	260.6
512	2	25.0	1.5	3	20.0	3.9	3	29.7	22.2	2	57.0	178.5
$H = h/4$												
64	$*$			4	15.5	3.9	4	18.8	21.1	5	19.6	160.3
128	$*$			4	18.8	2.9	4	24.8	14.6	5	23.2	93.1
256	$*$			4	20.3	2.3	4	28.0	9.2	5	24.4	50.8
512	$*$			4	24.5	2.8	4	33.0	7.7	5	30.6	39.7
$H = h/8$												
64	$*$			7	15.6	4.6	6	21.3	21.7	6	25.7	126.9
128	$*$			7	18.6	3.5	6	24.8	13.1	6	33.3	74.2
256	$*$			7	20.7	2.7	6	26.7	7.3	6	37.0	37.0
512	$*$			$*$			6	34.8	7.4	6	42.5	25.4

the PDE is discretized and the complementarity problem is solved, a coarse grid is introduced to accelerate the nonlinear convergence, and to precondition the linear Jacobian solver in a semismooth Newton iteration. With the help of a smoothed grid sequencing, a semismooth Newton method and a two-level restricted Schwarz preconditioner, we have showed numerically that the family of two-grid Newton-Krylov-Schwarz algorithms has a fast and robust convergence and that the rate of convergence is nearly independent of the number of unknowns of the problem and

the number of processors. Surprisingly good results were obtained for solving some rather difficult obstacle problems with millions of unknowns and on parallel machines with up to 512 processors.

References

1. S. Balay, K. Buschelman, W.D. Gropp, D. Kaushik, M. Knepley, L.C. McInnes, B.F. Smith, and H. Zhang. *PETSc Users Manual*. Argonne National Laboratory, Berkeley, CA, 2009.
2. X.-C. Cai and M. Sarkis. A restricted additive Schwarz preconditioner for general sparse linear systems. *SIAM J. Sci. Comput.*, 21:792–797, 1999.
3. R. Cottle, J.-S. Pang, and R. Stone. *The Linear Complementarity Problem*. Academic Press, Boston, MA, 1992.
4. M.C. Ferris and J.-S. Pang. Engineering and economic applications of complementarity problems. *SIAM Rev.*, 39:669–713, 1997.
5. A. Fischer. A special Newton-type optimization method. *Optimization*, 24:269–284, 1992.
6. P.T. Harker and J.-S. Pang. Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications. *Math. Prog.*, 48: 161–220, 1990.
7. C. Kanzow. Inexact semismooth Newton methods for large-scale complementarity problems. *Optim. Methods Softw.*, 19:309–325, 2004.
8. T. De Luca, F. Facchinei, and C. Kanzow. A semismooth equation approach to the solution of nonlinear complementarity problems. *Math. Prog.*, 75:407–439, 1996.
9. B. Smith, P. Bjørstad, and W. Gropp. *Domain Decomposition: Parallel Multilevel Methods for Eliptic Partial Differential Equations*. Cambridge University Press, New York, NY, 1996.
10. A. Toselli and O. Widlund. *Domain Decomposition Methods – Algorithms and Theory*. Springer, Berlin, 2005.

A Posteriori Error Estimates for Semilinear Boundary Control Problems

Yanping Chen[1] and Zuliang Lu[2]

[1] School of Mathematical Sciences, South China Normal University, Guangzhou 510631, P.R.China, yanpingchen@scnu.edu.cn
[2] College of Mathematics and Computer Sciences, Chongqing Three Gorges University, Chongqing 404000, P.R.China, zulianglux@126.com

1 Introduction

In this paper we study the finite element approximation for boundary control problems governed by semilinear elliptic equations. Optimal control problems are very important model in science and engineering numerical simulation. They have various physical backgrounds in many practical applications. Finite element approximation of optimal control problems plays a very important role in the numerical methods for these problems. The approximation of optimal control by piecewise constant functions is well investigated by [7, 8]. The discretization for semilinear elliptic optimal control problems is discussed in [2]. Systematic introductions of the finite element method for optimal control problems can be found in [10].

As one of important kinds of optimal control problems, the boundary control problem is widely used in scientific and engineering computing. The literature on this problem is huge, see, e.g. [1, 9]. For some linear optimal boundary control problems, [11] investigates a posteriori error estimates and adaptive finite element methods. [3] discusses the numerical approximation of boundary optimal control problems governed by semilinear elliptic partial differential equations with pointwise constraints on the control. Although a priori error estimates and a posteriori error estimates of finite element approximation are widely used in numerical simulations, it is not yet been utilized in semilinear boundary control problems.

Recently, in [4, 5, 6], we have derived a priori error estimates and superconvergence for linear optimal control problems using mixed finite element methods. A posteriori error analysis of mixed finite element methods for general convex optimal control problems has been addressed in [13].

In this paper, we derive a posteriori error estimates for a class of boundary control problems governed by semilinear elliptic equation. The problem that we are interested in is the following semilinear boundary control problems:

$$\min_{u\in K\subset U} \big\{g(y) + j(u)\big\} \tag{1}$$

subject to the state equation

$$-\mathrm{div}(A\nabla y) + \phi(y) = f,\ x \in \Omega, \tag{2}$$

$$(A\nabla y)\cdot n = Bu + z_0,\ x \in \partial\Omega, \tag{3}$$

where the bounded open set $\Omega \subset \mathbb{R}^2$ is a convex polygon, g and j are convex functionals, $f \in L^2(\Omega)$, $z_0 \in U = L^2(\partial\Omega)$, B is a continuous linear operator from U to $L^2(\Omega)$, and K is a closed convex set of U. For any $R > 0$ the function $\phi(\cdot) \in W^{1,\infty}(-R, R)$, $\phi'(y) \in L^2(\Omega)$ for any $y \in H^1(\Omega)$, and $\phi'(y) \geq 0$. We assume that the coefficient matrix $A(x) = (a_{i,j}(x))_{2\times 2} \in (W^{1,\infty}(\Omega))^{2\times 2}$ is a symmetric positive definite matrix and there are constants $c_0,\ c_1 > 0$ satisfying for any vector $\mathbf{X} \in \mathbb{R}^2$, $c_0\|\mathbf{X}\|^2_{\mathbb{R}^2} \leq \mathbf{X}^t A\mathbf{X} \leq c_1\|\mathbf{X}\|^2_{\mathbb{R}^2}$.

In this paper, we adopt the standard notation $W^{m,p}(\Omega)$ for Sobolev spaces on Ω with a norm $\|\cdot\|_{m,p}$ given by $\|v\|^p_{m,p} = \sum\limits_{|\alpha|\leq m} \|D^\alpha v\|^p_{L^p(\Omega)}$, a semi-norm $|\cdot|_{m,p}$ given by $|v|^p_{m,p} = \sum\limits_{|\alpha|=m} \|D^\alpha v\|^p_{L^p(\Omega)}$. We set $W^{m,p}_0(\Omega) = \{v \in W^{m,p}(\Omega) : v|_{\partial\Omega} = 0\}$. For p=2, we use the notation $H^m(\Omega) = W^{m,2}(\Omega)$, $H^m_0(\Omega) = W^{m,2}_0(\Omega)$, and $\|\cdot\|_m = \|\cdot\|_{m,2}$, $|\cdot| = \|\cdot\|_{0,2}$. In addition C or c denotes a general positive constant independent of h.

The plan of this paper is as follows. In next section, we present the finite element discretization for semilinear boundary control problems. A posteriori error estimates are established for the boundary control problems in Sect. 3.

2 Finite Elements for Boundary Control Problems

We shall now describe the finite element discretization of general semilinear convex boundary control problems (1)–(3). Let $V = H^1(\Omega)$, $W = L^2(\Omega)$.

Let

$$a(y, w) = \int_\Omega (A\nabla y)\cdot\nabla w, \qquad \forall y, w \in V, \tag{4}$$

$$(f_1, f_2) = \int_\Omega f_1 f_2, \qquad \forall (f_1, f_2) \in W \times W, \tag{5}$$

$$(u, v)_U = \int_{\partial\Omega} uv, \qquad \forall (u, v) \in U \times U. \tag{6}$$

Then the boundary control problems (1)–(3) can be restated as

$$\min_{u\in K\subset U} \{g(y) + j(u)\} \tag{7}$$

subject to

$$a(y, w) + (\phi(y), w) = (f, w) + (Bu + z_0, w)_U, \qquad \forall w \in V, \tag{8}$$

where the inner product in $L^2(\Omega)$ or $L^2(\Omega)^2$ is indicated by $(\cdot,\cdot)$.

It is well known (see, e.g., [11]) that the optimal control problem has a solution (y, u), and that a pair (y, u) is the solution of (7)–(8) if and only if there is a co-state $p \in V$ such that triplet (y, p, u) satisfies the following optimality conditions:

$$a(y, w) + (\phi(y), w) = (f, w) + (Bu + z_0, w)_U, \quad \forall w \in V, \tag{9}$$

$$a(q, p) + (\phi'(y)p, q) = (g'(y), q), \quad \forall q \in V, \tag{10}$$

$$(j'(u) + B^*p, v - u)_U \geq 0, \quad \forall v \in K \subset U, \tag{11}$$

where B^* is the adjoint operator of B, g' and j' are the derivatives of g and j. In the rest of the paper, we shall simply write the product as $(\cdot, \cdot)$ whenever no confusion should be caused.

Let us consider the finite element approximation of the control problem (7)–(8). Let T_h be regular partition of Ω. Associated with T_h is a finite dimensional subspace V_h of $C(\bar{\Omega})$, such that $\chi|_\tau$ are polynomials of m-order ($m \geq 1$) $\forall \chi \in V_h$ and $\tau \in \mathrm{T}_h$. It is easy to see that $V_h \subset V$.

Let T_U^h be a partition of $\partial\Omega$ and $\partial\Omega = \bigcup_{s \in \mathrm{T}_U^h} \bar{s}$. Associated with T_U^h is another finite dimensional subspace U_h of $L^2(\partial\Omega)$, such that $\chi|_s$ are polynomials of m-order ($m \geq 0$) $\forall \chi \in U_h$ and $s \in \mathrm{T}_U^h$. Let h_τ (h_s) denote the maximum diameter of the element τ (s) in T_h (T_U^h), $h = \max_{\tau \in \mathrm{T}_h}\{h_\tau\}$, and $h_U = \max_{s \in \mathrm{T}_U^h}\{h_s\}$.

By the definition of finite element subspace, the finite element discretization of (7)–(8) is as follows: compute $(y_h, u_h) \in V_h \times U_h$ such that

$$\min_{u_h \in K_h \subset U_h} \{g(y_h) + j(u_h)\} \tag{12}$$

$$a(y_h, w_h) + (\phi(y_h), w_h) = (f, w_h) + (Bu_h + z_0, w_h)_U, \quad \forall w_h \in V_h, \tag{13}$$

where K_h is a non-empty closed convex set in U_h.

Again, it follows that the optimal control problem (12)–(13) has a solution (y_h, u_h), and that a pair (y_h, u_h) is the solution of (12)–(13) if and only if there is a co-state $p_h \in V_h$ such that triplet (y_h, p_h, u_h) satisfies the following optimality conditions:

$$a(y_h, w_h) + (\phi(y_h), w_h) = (f, w_h) + (Bu_h + z_0, w_h)_U, \quad \forall w_h \in V_h, \tag{14}$$

$$a(q_h, p_h) + (\phi'(y_h)p_h, q_h) = (g'(y_h), q_h), \quad \forall q_h \in V_h, \tag{15}$$

$$(j'(u_h) + B^*p_h, v_h - u_h)_U \geq 0, \quad \forall v_h \in K_h. \tag{16}$$

In the rest of the paper, we shall use some intermediate variables. For any control function $u_h \in K$, we first define the state solution $(y(u_h), p(u_h))$ which satisfies

$$a(y(u_h), w) + (\phi(y(u_h)), w) = (f, w) + (Bu_h + z_0, w)_U, \quad \forall w \in V, \tag{17}$$

$$a(q, p(u_h)) + (\phi'(y(u_h))p(u_h), q) = (g'(y(u_h)), q), \quad \forall q \in V. \tag{18}$$

The following Lemma is important in deriving a posteriori error estimates of residual type.

Lemma 1. *Let π_h be the standard Lagrange interpolation operator. Then for $m = 0$ or 1, $1 < q \leq \infty$ and $\forall v \in W^{2,q}(\Omega)$,*

$$\|v - \pi_h v\|_{W^{m,q}(\tau)} \leq C h_\tau^{2-m} |v|_{W^{2,q}(\tau)}. \tag{19}$$

3 A Posteriori Error Estimates

For given $u \in K$, let S be the inverse operator of the state equation (9), such that $y(u) = SBu$ is the solution of the state equation (9). Similarly, for given $u_h \in K_h$, $y_h(u_h) = S_h B u_h$ is the solution of the discrete state Eq. (14). Let

$$S(u) = g(SBu) + j(u),$$

$$S_h(u_h) = g(S_h B u_h) + j(u_h).$$

It is clear that S and S_h are well defined and continuous on K and K_h. Also the functional S_h can be naturally extended on K. Then (7) and (12) can be represented as

$$\min_{u \in K} \{S(u)\}, \tag{20}$$

$$\min_{u_h \in K_h} \{S_h(u_h)\}. \tag{21}$$

It can be shown that

$$\begin{aligned}
&(S'(u), v) = (j'(u) + B^* p, v),\\
&(S'(u_h), v) = (j'(u_h) + B^* p(u_h), v),\\
&(S_h'(u_h), v) = (j'(u_h) + B^* p_h, v),
\end{aligned}$$

where $p(u_h)$ is the solution of the equations (17)–(18).

In many applications, $S(\cdot)$ is uniform convex near the solution u (see, e.g., [12]). The convexity of $S(\cdot)$ is closely related to the second order sufficient conditions of the control problem, which are assumed in many studies on numerical methods of the problem. If $S(\cdot)$ is uniformly convex, then there is a $c > 0$, such that

$$(S'(u) - S'(u_h), u - u_h) \geq c \|u - u_h\|_{0,\partial\Omega}^2, \tag{22}$$

where u and u_h are the solutions of (7) and (12), respectively. We will assume the above inequality throughout this paper.

Now we establish the following a posteriori error estimates, which can be proved similarly to the proofs given in [12].

Theorem 1. *Let u and u_h be the solutions of (7) and (12), respectively. Assume that $K_h \subset K$ for any element $s \in \mathrm{T}_U^h$, $(B^* p_h + j'(u_h))|_s \in H^1(s)$ and that there exists a $v_h \in K_h$ such that*

$$|(B^*p_h + j'(u_h), v_h - u)_U| \le C \sum_s h_s |B^*p_h + j'(u_h)|_{1,s} \|u - u_h\|_{0,s}. \quad (23)$$

Then

$$\|u - u_h\|^2_{0,\partial\Omega} \le C \left(\eta_1^2 + \|p_h - p(u_h)\|^2_{0,\partial\Omega}\right), \quad (24)$$

where

$$\eta_1^2 = \sum_s h_s^2 |B^*p_h + j'(u_h)|^2_{1,s}.$$

Proof. It follows from (22) that for all $v_h \in K_h$,

$$\begin{aligned} c\|u - u_h\|^2_{0,\partial\Omega} \le & S'(u)(u - u_h) - S'(u_h)(u - u_h) \\ = & (j'(u) + B^*p, u - u_h)_U - (j'(u_h) + B^*p(u_h), u - u_h)_U \\ \le & -(j'(u_h) + B^*p(u_h), u - u_h)_U \\ \le & (j'(u_h) + B^*p_h, u_h - v_h)_U + (j'(u_h) + B^*p_h, v_h - u)_U \\ & + (B^*(p_h - p(u_h)), u - u_h)_U \\ \le & (j'(u_h) + B^*p_h, v_h - u)_U + (B^*(p_h - p(u_h)), u - u_h)_U. \end{aligned} \quad (25)$$

It is easy to see that

$$\begin{aligned} (B^*(p_h - p(u_h)), u - u_h)_U &\le C\|B^*(p_h - p(u_h))\|^2_{0,\partial\Omega} + \frac{\delta}{2}\|u - u_h\|^2_{\partial\Omega} \\ &\le C\|p_h - p(u_h)\|^2_{0,\partial\Omega} + \frac{\delta}{2}\|u - u_h\|^2_{\partial\Omega}, \end{aligned} \quad (26)$$

where δ is an arbitrary positive constant. Then (25)-(26) and (23) imply that

$$\begin{aligned} c\|u - u_h\|^2_{0,\partial\Omega} \le & C \sum_s h_s^2 |B^*p_h + j'(u_h)|^2_{1,s} \\ & + C\|p_h - p(u_h)\|^2_{0,\partial\Omega} + \frac{\delta}{2}\|u - u_h\|^2_{\partial\Omega} \\ = & C\eta_1^2 + C\|p_h - p(u_h)\|^2_{0,\partial\Omega} + \frac{\delta}{2}\|u - u_h\|^2_{\partial\Omega}. \end{aligned} \quad (27)$$

Then (24) follows from (27).

Now, we are able to derive the following result.

Theorem 2. *Let $(y(u_h), p(u_h))$ and (y_h, p_h) be the solutions of (17)–(18) and (14)–(15), respectively. Assume that g' is Lipschitz continuous in a neighborhood of y. Then*

$$\|p(u_h) - p_h\|^2_{1,\Omega} \le C(\eta_2^2 + \eta_3^2) + C\|y(u_h) - y_h\|^2_{0,\Omega}, \quad (28)$$

where

$$\eta_2^2 = \sum_{\tau\in \mathrm{T}_h} h_\tau^2 \int_\tau \left(g'(y_h) + \operatorname{div}(A^*\nabla p_h) - \phi'(y_h)p_h\right)^2,$$

$$\eta_3^2 = \sum_{l\cap\partial\Omega=\phi} h_l \int_l [A^*\nabla p_h \cdot n]^2 + \sum_{l\subset\partial\Omega} h_l \int_l (A^*\nabla p_h \cdot n)^2,$$

where l is a face of an element τ, $[(A^*\nabla p_h \cdot n)]$ is the A-normal derivative jump over the interior face l, defined by

$$[(A^*\nabla p_h \cdot n)]_l = (A^*\nabla p_h|_{\tau_l^1} - A^*\nabla p_h|_{\tau_l^2}) \cdot n,$$

where n is the unit normal vector on $l = \bar\tau_l^1 \cap \bar\tau_l^2$ outwards τ_l^1.

Analogously to Theorem 2, we show the following estimate:

Theorem 3. *Let $(y(u_h), p(u_h))$ and (y_h, p_h) be the solutions of (17)–(18) and (14)–(15), respectively. Assume that g' is Lipschitz continuous in a neighborhood of y. Then*

$$\|y(u_h) - y_h\|_{1,\Omega}^2 \le C(\eta_4^2 + \eta_5^2), \tag{29}$$

where

$$\eta_4^2 = \sum_{\tau\in \mathrm{T}_h} h_\tau^2 \int_\tau \left(f + \operatorname{div}(A\nabla y_h) - \phi(y_h)\right)^2,$$

$$\eta_5^2 = \sum_{l\cap\partial\Omega=\phi} h_l \int_l [A\nabla y_h \cdot n]^2 + \sum_{l\subset\partial\Omega} h_l \int_l (A\nabla y_h \cdot n - Bu_h - z_0)^2,$$

where l is a face of an element τ, $[(A\nabla y_h \cdot n)]$ is the A-normal derivative jump over the interior face l, defined by

$$[(A\nabla y_h \cdot n)]_l = (A\nabla y_h|_{\tau_l^1} - A\nabla y_h|_{\tau_l^2}) \cdot n,$$

where n is the unit normal vector on $l = \bar\tau_l^1 \cap \bar\tau_l^2$ outwards τ_l^1.

Now, we are able to derive our main result.

Theorem 4. *Let (y, p, u) and (y_h, p_h, u_h) be the solutions of (9)–(11) and (14)–(16), respectively. Assume that all the conditions of Theorems 1–3 hold. Then*

$$\|u - u_h\|_{0,\partial\Omega}^2 + \|y - y_h\|_{1,\Omega}^2 + \|p(u_h) - p_h\|_{1,\Omega}^2 \le C\sum_{i=1}^5 \eta_i^2, \tag{30}$$

where η_i, $i = 1, 2, 3, 4, 5$ are defined in Theorem 1, Theorem 2, and Theorem 3.

Proof. From Theorems 1–3 and the trace theorem we can see that

$$\begin{aligned}\|u - u_h\|_{0,\partial\Omega}^2 &\le C\left(\eta_1^2 + \|p_h - p(u_h)\|_{0,\partial\Omega}^2\right) \le C\left(\eta_1^2 + \|p_h - p(u_h)\|_{1,\Omega}^2\right)\\ &\le C\left(\eta_1^2 + \eta_2^2 + \eta_3^2 + \|y_h - y(u_h)\|_{0,\Omega}^2\right) \le C\sum_{i=1}^5 \eta_i^2.\end{aligned} \tag{31}$$

Note that

$$\|y - y_h\|_{1,\Omega} \le \|y_h - y(u_h)\|_{1,\Omega} + \|y_h - y(u_h)\|_{1,\Omega}, \tag{32}$$
$$\|p - p_h\|_{1,\Omega} \le \|p_h - p(u_h)\|_{1,\Omega} + \|p_h - p(u_h)\|_{1,\Omega}. \tag{33}$$

It follows from (9) and (17) that

$$a(y - y(u_h), w) + (\phi(y) - \phi(y(u_h)), w) = (B(u - u_h), w), \quad \forall w \in V. \tag{34}$$

Let $w = y - y(u_h)$, we have that

$$\|y - y(u_h)\|_{1,\Omega} \le C\|B(u - u_h)\|_{0,\Omega} \le C\|u - u_h\|_{0,\partial\Omega}. \tag{35}$$

Similarly, from (10) and (18) imply that

$$\begin{aligned} &a(q, p - p(u_h)) + (\phi'(y)(p - p(u_h)), q) \\ &= (g'(y) - g'(y(u_h)), q) + ((\phi'(y(u_h)) - \phi'(y))p(u_h), q), \quad \forall q \in V. \end{aligned} \tag{36}$$

Let $q = p - p(u_h)$, using (31), (35), and the trace theorem, we have that

$$\begin{aligned} \|p - p(u_h)\|_{1,\Omega}^2 &\le (g'(y) - g'(y(u_h)), p - p(u_h)) \\ &\quad + ((\phi'(y(u_h)) - \phi'(y))p(u_h), p - p(u_h)) \\ &\le \|g'(y) - g'(y(u_h))\|_{0,\Omega}\|p - p(u_h)\|_{0,\Omega} \\ &\quad + \|\phi'(y(u_h)) - \phi'(y)\|_{0,\Omega}\|p(u_h)\|_{0,4,\Omega}\|p - p(u_h)\|_{0,4,\Omega} \\ &\le C\|y - y(u_h)\|_{0,\Omega}^2 + C\delta\|p - p(u_h)\|_{1,\Omega}^2 \\ &\le C\|u - u_h\|_{0,\partial\Omega}^2 + C\delta\|p - p(u_h)\|_{1,\Omega}^2 \\ &\le C\sum_{i=1}^{5}\eta_i^2 + C\delta\|p - p(u_h)\|_{1,\Omega}^2. \end{aligned}$$

Then, for δ sufficiently small,

$$\|p - p(u_h)\|_{1,\Omega}^2 \le C\sum_{i=1}^{5}\eta_i^2, \tag{37}$$

and thus (30) follows from (31), (35), and (37).

Acknowledgments The authors express their thanks to the referees for their helpful suggestions, which led to improvements of the presentation.

References

1. W. Alt and U. Mackenroth. Convergence of finite element approximations to state constrained convex parabolic boundary control problems. *SIAM J. Control Optim.*, 27:718–736, 1989.
2. N. Arada, E. Casas, and F. Tröltzsch. Error estimates for the numerical approximation of a semilinear elliptic control problem. *Comp. Optim. Appl.*, 23:201–229, 2002.
3. N. Arada, E. Casas, and F. Tröltzsch. Error estimates for the numerical approximation of a boundary semilinear elliptic control problem. *Comp. Optim. Appl.*, 31:193–219, 2005.
4. Y. Chen, L. Dai, and Z. Lu. Superconvergence of rectangular mixed finite element methods for constrained optimal control problem. *Adv. Appl. Math. Mech.*, 2:56–75, 2010.
5. Y. Chen and W.B. Liu. A posteriori error estimates for mixed finite element solutions of convex optimal control problems. *J. Comp. Appl. Math.*, 211:76–89, 2008.
6. Y. Chen and Z. Lu. Error estimates of fully discrete mixed finite element methods for semilinear quadratic parabolic optimal control problems. *Comput. Methods Appl. Mech. Eng.*, 199:1415–1423, 2010.
7. F.S. Falk. Approximation of a class of optimal control problems with order of convergence estimates. *J. Math. Anal. Appl.*, 44:28–47, 1973.
8. T. Geveci. On the approximation of the solution of an optimal control problem governed by an elliptic equation. *RAIRO: Numer. Anal.*, 13:313–328, 1979.
9. I. Lasiecka. Ritz-galerkin approximation of the time optimal boundary control problem for parabolic systems with dirichlet boundary conditions. *SIAM J. Control Optim.*, 22: 477–500, 1984.
10. J.L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer, Berlin, 1971.
11. W.B. Liu and N.N. Yan. A posteriori error estimates for convex boundary control problems. *SIAM J. Numer. Anal.*, 39:73–99, 2002.
12. W.B. Liu and N.N. Yan. A posteriori error estimates for control problems governed by nonlinear elliptic equation. *Appl. Numer. Math.*, 47:173–187, 2003.
13. Z. Lu and Y. Chen. A posteriori error estimates of triangular mixed finite element methods for semilinear optimal control problems. *Adv. Appl. Math. Mech.*, 1:242–256, 2009.

Editorial Policy

1. Volumes in the following three categories will be published in LNCSE:

i) Research monographs
ii) Tutorials
iii) Conference proceedings

Those considering a book which might be suitable for the series are strongly advised to contact the publisher or the series editors at an early stage.

2. Categories i) and ii). Tutorials are lecture notes typically arising via summer schools or similar events, which are used to teach graduate students. These categories will be emphasized by Lecture Notes in Computational Science and Engineering. **Submissions by interdisciplinary teams of authors are encouraged.** The goal is to report new developments – quickly, informally, and in a way that will make them accessible to non-specialists. In the evaluation of submissions timeliness of the work is an important criterion. Texts should be well-rounded, well-written and reasonably self-contained. In most cases the work will contain results of others as well as those of the author(s). In each case the author(s) should provide sufficient motivation, examples, and applications. In this respect, Ph.D. theses will usually be deemed unsuitable for the Lecture Notes series. Proposals for volumes in these categories should be submitted either to one of the series editors or to Springer-Verlag, Heidelberg, and will be refereed. A provisional judgement on the acceptability of a project can be based on partial information about the work: a detailed outline describing the contents of each chapter, the estimated length, a bibliography, and one or two sample chapters – or a first draft. A final decision whether to accept will rest on an evaluation of the completed work which should include

– at least 100 pages of text;
– a table of contents;
– an informative introduction perhaps with some historical remarks which should be accessible to readers unfamiliar with the topic treated;
– a subject index.

3. Category iii). Conference proceedings will be considered for publication provided that they are both of exceptional interest and devoted to a single topic. One (or more) expert participants will act as the scientific editor(s) of the volume. They select the papers which are suitable for inclusion and have them individually refereed as for a journal. Papers not closely related to the central topic are to be excluded. Organizers should contact the Editor for CSE at Springer at the planning stage, see *Addresses* below.

In exceptional cases some other multi-author-volumes may be considered in this category.

4. Only works in English will be considered. For evaluation purposes, manuscripts may be submitted in print or electronic form, in the latter case, preferably as pdf- or zipped ps-files. Authors are requested to use the LaTeX style files available from Springer at http://www. springer.com/authors/book+authors?SGWID=0-154102-12-417900-0.

For categories ii) and iii) we strongly recommend that all contributions in a volume be written in the same LaTeX version, preferably LaTeX2e. Electronic material can be included if appropriate. Please contact the publisher.

Careful preparation of the manuscripts will help keep production time short besides ensuring satisfactory appearance of the finished book in print and online.

5. The following terms and conditions hold. Categories i), ii) and iii):

Authors receive 50 free copies of their book. No royalty is paid.
Volume editors receive a total of 50 free copies of their volume to be shared with authors, but no royalties.

Authors and volume editors are entitled to a discount of 33.3 % on the price of Springer books purchased for their personal use, if ordering directly from Springer.

6. Commitment to publish is made by letter of intent rather than by signing a formal contract. Springer-Verlag secures the copyright for each volume.

Addresses:

Timothy J. Barth
NASA Ames Research Center
NAS Division
Moffett Field, CA 94035, USA
barth@nas.nasa.gov

Michael Griebel
Institut für Numerische Simulation
der Universität Bonn
Wegelerstr. 6
53115 Bonn, Germany
griebel@ins.uni-bonn.de

David E. Keyes
Mathematical and Computer Sciences
and Engineering
King Abdullah University of Science
and Technology
P.O. Box 55455
Jeddah 21534, Saudi Arabia
david.keyes@kaust.edu.sa

and

Department of Applied Physics
and Applied Mathematics
Columbia University
500 W. 120 th Street
New York, NY 10027, USA
kd2112@columbia.edu

Risto M. Nieminen
Department of Applied Physics
Aalto University School of Science
and Technology
00076 Aalto, Finland
risto.nieminen@tkk.fi

Dirk Roose
Department of Computer Science
Katholieke Universiteit Leuven
Celestijnenlaan 200A
3001 Leuven-Heverlee, Belgium
dirk.roose@cs.kuleuven.be

Tamar Schlick
Department of Chemistry
and Courant Institute
of Mathematical Sciences
New York University
251 Mercer Street
New York, NY 10012, USA
schlick@nyu.edu

Editor for Computational Science
and Engineering at Springer:
Martin Peters
Springer-Verlag
Mathematics Editorial IV
Tiergartenstrasse 17
69121 Heidelberg, Germany
martin.peters@springer.com

Lecture Notes in Computational Science and Engineering

1. D. Funaro, *Spectral Elements for Transport-Dominated Equations.*
2. H.P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming.
3. W. Hackbusch, G. Wittum (eds.), *Multigrid Methods V.*
4. P. Deuflhard, J. Hermans, B. Leimkuhler, A.E. Mark, S. Reich, R.D. Skeel (eds.), *Computational Molecular Dynamics: Challenges, Methods, Ideas.*
5. D. Kröner, M. Ohlberger, C. Rohde (eds.), *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws.*
6. S. Turek, *Efficient Solvers for Incompressible Flow Problems.* An Algorithmic and Computational Approach.
7. R. von Schwerin, ***M**ulti **B**ody **S**ystem **SIM**ulation.* Numerical Methods, Algorithms, and Software.
8. H.-J. Bungartz, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.*
9. T.J. Barth, H. Deconinck (eds.), *High-Order Methods for Computational Physics.*
10. H.P. Langtangen, A.M. Bruaset, E. Quak (eds.), *Advances in Software Tools for Scientific Computing.*
11. B. Cockburn, G.E. Karniadakis, C.-W. Shu (eds.), *Discontinuous Galerkin Methods.* Theory, Computation and Applications.
12. U. van Rienen, *Numerical Methods in Computational Electrodynamics.* Linear Systems in Practical Applications.
13. B. Engquist, L. Johnsson, M. Hammill, F. Short (eds.), *Simulation and Visualization on the Grid.*
14. E. Dick, K. Riemslagh, J. Vierendeels (eds.), *Multigrid Methods VI.*
15. A. Frommer, T. Lippert, B. Medeke, K. Schilling (eds.), *Numerical Challenges in Lattice Quantum Chromodynamics.*
16. J. Lang, *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems.* Theory, Algorithm, and Applications.
17. B.I. Wohlmuth, *Discretization Methods and Iterative Solvers Based on Domain Decomposition.*
18. U. van Rienen, M. Günther, D. Hecht (eds.), *Scientific Computing in Electrical Engineering.*
19. I. Babuška, P.G. Ciarlet, T. Miyoshi (eds.), *Mathematical Modeling and Numerical Simulation in Continuum Mechanics.*
20. T.J. Barth, T. Chan, R. Haimes (eds.), *Multiscale and Multiresolution Methods.* Theory and Applications.
21. M. Breuer, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.*
22. K. Urban, *Wavelets in Numerical Simulation.* Problem Adapted Construction and Applications.

23. L.F. Pavarino, A. Toselli (eds.), *Recent Developments in Domain Decomposition Methods.*

24. T. Schlick, H.H. Gan (eds.), *Computational Methods for Macromolecules: Challenges and Applications.*

25. T.J. Barth, H. Deconinck (eds.), *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*.

26. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations*.

27. S. Müller, *Adaptive Multiscale Schemes for Conservation Laws*.

28. C. Carstensen, S. Funken, W. Hackbusch, R.H.W. Hoppe, P. Monk (eds.), *Computational Electromagnetics*.

29. M.A. Schweitzer, *A Parallel Multilevel Partition of Unity Method for Elliptic Partial Differential Equations*.

30. T. Biegler, O. Ghattas, M. Heinkenschloss, B. van Bloemen Waanders (eds.), *Large-Scale PDE-Constrained Optimization*.

31. M. Ainsworth, P. Davies, D. Duncan, P. Martin, B. Rynne (eds.), *Topics in Computational Wave Propagation*. Direct and Inverse Problems.

32. H. Emmerich, B. Nestler, M. Schreckenberg (eds.), *Interface and Transport Dynamics.* Computational Modelling.

33. H.P. Langtangen, A. Tveito (eds.), *Advanced Topics in Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming.

34. V. John, *Large Eddy Simulation of Turbulent Incompressible Flows.* Analytical and Numerical Results for a Class of LES Models.

35. E. Bänsch (ed.), *Challenges in Scientific Computing - CISC 2002.*

36. B.N. Khoromskij, G. Wittum, *Numerical Solution of Elliptic Differential Equations by Reduction to the Interface.*

37. A. Iske, *Multiresolution Methods in Scattered Data Modelling.*

38. S.-I. Niculescu, K. Gu (eds.), *Advances in Time-Delay Systems.*

39. S. Attinger, P. Koumoutsakos (eds.), *Multiscale Modelling and Simulation.*

40. R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Wildlund, J. Xu (eds.), *Domain Decomposition Methods in Science and Engineering.*

41. T. Plewa, T. Linde, V.G. Weirs (eds.), *Adaptive Mesh Refinement – Theory and Applications.*

42. A. Schmidt, K.G. Siebert, *Design of Adaptive Finite Element Software.* The Finite Element Toolbox ALBERTA.

43. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations II.*

44. B. Engquist, P. Lötstedt, O. Runborg (eds.), *Multiscale Methods in Science and Engineering.*

45. P. Benner, V. Mehrmann, D.C. Sorensen (eds.), *Dimension Reduction of Large-Scale Systems.*

46. D. Kressner, *Numerical Methods for General and Structured Eigenvalue Problems.*

47. A. Boriçi, A. Frommer, B. Joó, A. Kennedy, B. Pendleton (eds.), *QCD and Numerical Analysis III*.

48. F. Graziani (ed.), *Computational Methods in Transport.*

49. B. Leimkuhler, C. Chipot, R. Elber, A. Laaksonen, A. Mark, T. Schlick, C. Schütte, R. Skeel (eds.), *New Algorithms for Macromolecular Simulation.*

50. M. Bücker, G. Corliss, P. Hovland, U. Naumann, B. Norris (eds.), *Automatic Differentiation: Applications, Theory, and Implementations.*

51. A.M. Bruaset, A. Tveito (eds.), *Numerical Solution of Partial Differential Equations on Parallel Computers.*

52. K.H. Hoffmann, A. Meyer (eds.), *Parallel Algorithms and Cluster Computing.*

53. H.-J. Bungartz, M. Schäfer (eds.), *Fluid-Structure Interaction.*

54. J. Behrens, *Adaptive Atmospheric Modeling.*

55. O. Widlund, D. Keyes (eds.), *Domain Decomposition Methods in Science and Engineering XVI.*

56. S. Kassinos, C. Langer, G. Iaccarino, P. Moin (eds.), *Complex Effects in Large Eddy Simulations.*

57. M. Griebel, M.A Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations III.*

58. A.N. Gorban, B. Kégl, D.C. Wunsch, A. Zinovyev (eds.), *Principal Manifolds for Data Visualization and Dimension Reduction.*

59. H. Ammari (ed.), *Modeling and Computations in Electromagnetics: A Volume Dedicated to Jean-Claude Nédélec.*

60. U. Langer, M. Discacciati, D. Keyes, O. Widlund, W. Zulehner (eds.), *Domain Decomposition Methods in Science and Engineering XVII.*

61. T. Mathew, *Domain Decomposition Methods for the Numerical Solution of Partial Differential Equations.*

62. F. Graziani (ed.), *Computational Methods in Transport: Verification and Validation.*

63. M. Bebendorf, *Hierarchical Matrices.* A Means to Efficiently Solve Elliptic Boundary Value Problems.

64. C.H. Bischof, H.M. Bücker, P. Hovland, U. Naumann, J. Utke (eds.), *Advances in Automatic Differentiation.*

65. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations IV.*

66. B. Engquist, P. Lötstedt, O. Runborg (eds.), *Multiscale Modeling and Simulation in Science.*

67. I.H. Tuncer, Ü. Gülcat, D.R. Emerson, K. Matsuno (eds.), *Parallel Computational Fluid Dynamics 2007.*

68. S. Yip, T. Diaz de la Rubia (eds.), *Scientific Modeling and Simulations.*

69. A. Hegarty, N. Kopteva, E. O'Riordan, M. Stynes (eds.), *BAIL* 2008 – *Boundary and Interior Layers.*

70. M. Bercovier, M.J. Gander, R. Kornhuber, O. Widlund (eds.), *Domain Decomposition Methods in Science and Engineering XVIII.*

71. B. Koren, C. Vuik (eds.), *Advanced Computational Methods in Science and Engineering.*

72. M. Peters (ed.), *Computational Fluid Dynamics for Sport Simulation.*

73. H.-J. Bungartz, M. Mehl, M. Schäfer (eds.), *Fluid Structure Interaction II - Modelling, Simulation, Optimization.*

74. D. Tromeur-Dervout, G. Brenner, D.R. Emerson, J. Erhel (eds.), *Parallel Computational Fluid Dynamics 2008.*

75. A.N. Gorban, D. Roose (eds.), *Coping with Complexity: Model Reduction and Data Analysis.*

76. J.S. Hesthaven, E.M. Rønquist (eds.), *Spectral and High Order Methods for Partial Differential Equations.*

77. M. Holtz, *Sparse Grid Quadrature in High Dimensions with Applications in Finance and Insurance.*

78. Y. Huang, R. Kornhuber, O.Widlund, J. Xu (eds.), *Domain Decomposition Methods in Science and Engineering XIX.*

For further information on these books please have a look at our mathematics catalogue at the following URL: www.springer.com/series/3527

Monographs in Computational Science and Engineering

1. J. Sundnes, G.T. Lines, X. Cai, B.F. Nielsen, K.-A. Mardal, A. Tveito, *Computing the Electrical Activity in the Heart.*

For further information on this book, please have a look at our mathematics catalogue at the following URL: www.springer.com/series/7417

Texts in Computational Science and Engineering

1. H. P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming. 2nd Edition
2. A. Quarteroni, F. Saleri, P. Gervasio, *Scientific Computing with MATLAB and Octave.* 3rd Edition
3. H. P. Langtangen, *Python Scripting for Computational Science*. 3rd Edition
4. H. Gardner, G. Manduchi, *Design Patterns for e-Science*.
5. M. Griebel, S. Knapek, G. Zumbusch, *Numerical Simulation in Molecular Dynamics*.
6. H. P. Langtangen, *A Primer on Scientific Programming with Python.*
7. A. Tveito, H. P. Langtangen, B. F. Nielsen, X. Cai, *Elements of Scientific Computing.*

For further information on these books please have a look at our mathematics catalogue at the following URL: www.springer.com/series/5151