Domain Decomposition Methods for the Helmholtz Equation: A Numerical Investigation

Martin J. Gander and Hui Zhang

[1] University of Geneva martin.gander@unige.ch
[2] hui.zhang@unige.ch

1 Introduction

We are interested in solving the Helmholtz equation

$$\begin{cases} -\triangle u(x,y,z) - k^2(x,y,z)\, u(x,y,z) = g(x,y,z), & (x,y,z) \in \Omega, \\ \partial_n u(x,y,z) - \mathbf{i}k(x,y,z)\, u(x,y,z) = 0, & (x,y,z) \in \partial\Omega, \end{cases} \tag{1}$$

where $k := 2\pi f/c$ is the wavenumber with frequency $f \in \mathbf{R}$ and $c := c(x,y,z)$ is the velocity of the medium, which varies in space. The geophysical model SEG–SALT is used as a benchmark problem on which we will test some existing domain decomposition methods in this paper. In this model, the domain Ω is defined as $(0,13{,}520) \times (0,13{,}520) \times (0,4{,}200)$ m^3, the velocity is described as piecewise constants on $676 \times 676 \times 210$ cells and varies from 1,500 to 4,500 m/s, and the source g is a Dirac function at the point $(6{,}000, 6{,}760, 10)$.

To discretize the problem (1) on a coarser mesh, the velocity is sub-sampled to less number of cells such that every cell has a constant velocity and contains one or more mesh elements. Then the problem (1) is discretized with $Q1$ finite elements (i.e. trilinear local basis functions on brick elements).

We first test the direct solver $A\backslash b$ in Matlab; the results are listed in Table 1 where nw is the number of wavelength along the x-direction at the lowest velocity. At $f = 2$, the direct solver runs out of memory after 6 h on a computer with 64 GB of memory. The inefficiency in both memory and time of the direct solver for large scale problems calls for cheaper iterative methods. For a review of current iterative methods for the Helmholtz equation, we refer to [6]. In this work, we focus on domain decomposition methods which are easily parallelized.

2 Overview of Some Existing Methods

Due to the indefiniteness of the Helmholtz equation, the classical Schwarz method with Dirichlet transmission conditions fails to converge. As a remedy, [5] introduced

R. Bank et al. (eds.), *Domain Decomposition Methods in Science and Engineering XX*, Lecture Notes in Computational Science and Engineering 91,
DOI 10.1007/978-3-642-35275-1__24, © Springer-Verlag Berlin Heidelberg 2013

Table 1. Test of the direct solver (backslash in Matlab)

f	1/4	1/2	1	2
nw	2.25	4.5	9	18
mesh	$24 \times 24 \times 8$	$48 \times 48 \times 16$	$96 \times 96 \times 32$	$192 \times 192 \times 64$
CPU	1.28s	27.51s	829.91s	> 6h

first-order absorbing transmission conditions to replace the Dirichlet transmission conditions. This type of interface condition was also adopted in [7] to regularize subdomain problems. More general local transmission conditions of zero or second order were proposed and analyzed in [10, 11] with parameters optimized for accelerating convergence. More advanced and even non-local transmission conditions can be used, see [3, 12, 18], and also [2, 13] in this volume. In this paper, however, we will restrict ourselves to local transmission conditions.

Another remedy is to modify the usual coarse problem, which probably originated from the multigrid context, first suggested by Achi Brandt and presented in [19]. In their paper [7], Farhat et al. used plane waves on the interface as basis of the coarse space. The idea turns out to be very successful and was followed by Farhat et al. [8], Kimn and Sarkis [15], and Li and Tu [17], and will also be used for the optimized Schwarz methods in this paper. Note that, however, the coarse problem does not change the underlying subdomain problems.

In the following paragraphs, we will give a brief introduction to these methods at the (almost) continuous level.

2.1 The Non-overlapping Methods

We partition the domain into non-overlapping subdomains denoted by $\overline{\Omega} := \cup_i \overline{\Omega_i}$, and we call the set of points shared by more than two subdomains (or shared by two subdomains and the outer boundary $\partial\Omega$) corners. In three dimensions, this includes vertices and edges. We call all the points shared by exactly two subdomains the interface Γ, and in particular a connected component of the interface shared by Ω_i and Ω_j is called interface segment Γ_{ij}.

If we know the Neumann, Dirichlet or Robin data (denoted by λ) of the exact solution on the interface, then we can recover the exact solution from the corresponding boundary value problems defined on subdomains (as long as they are well-posed) with *continuous constraints at corners*. Since on every subdomain there is a recovered solution that gives Dirichlet, Neumann or Robin traces on the interface, we expect for each interface segment Γ_{ij} the traces from Ω_i and Ω_j to be equal. The above process indeed sets up an equation, denoted by $F\lambda = d$, for the interface data λ of the exact solution. For the Helmholtz equation, an additional coarse problem is introduced such that $(I - FQ(Q^*FQ)^{-1}Q^*)F\lambda = (I - FQ(Q^*FQ)^{-1}Q^*)d$ is solved, where the columns of Q are traces of plane waves on the interface.

From the above point of view, we summarize some existing non-overlapping domain decomposition methods in Table 2. The (first-order) absorbing boundary data is defined as $\lambda := \partial_{\mathbf{n}} u - \mathbf{i}ku$. The lumped preconditioner is the stiffness submatrix $A_{\Gamma\Gamma}$ corresponding to the interface. The first three methods share interface data (up to a sign for the normal derivative) on their common interface segments, and are therefore one-field methods. This is in contrast to the last method, since optimized Schwarz methods have two sets of unknowns on each interface segment, and thus belong to the class of two-field methods. Note also that we do not have suitable preconditioners for the last two methods, which can be a subject for future study.

Table 2. The non-overlapping methods

Algorithms	Unknowns	Matching	Precond.
FETI-DPH ([8])	Neumann	Dirichlet	DtN/lumped
BDDC-H ([17])	Dirichlet	Neumann	NtD
FETI-H ([7])	Absorbing	Dirichlet	(none)
Optimized Schwarz ([10])	two-field Robin	two-field Robin	(none)

2.2 The Overlapping Methods

We partition the domain into overlapping subdomains. We will use the *substructured form*[3] as for the non-overlapping methods in Sect. 2.1. Note that in an overlapping setting, subdomains can not share the same interface data, since the interfaces are in different locations, and therefore all overlapping methods are in some sense two field methods, like the non-overlapping optimized Schwarz methods. The interface data used (both as unknowns and matching conditions) and related references are: Dirichlet [16], absorbing [4, 15], Neumann [14], Robin [9]. A coarse problem as in Sect. 2.1 is adopted but without corner constraints.

3 Numerical Experiments

All the experiments were done in Matlab with sequential codes. We use GMRES with zero initial guess to solve the substructured systems until the relative residual is less than 10^{-6} or the maximum iteration number is attained. The domain is partitioned in a Cartesian way. If we vary the mesh size, then the velocity in (1) is sub-sampled on the coarsest mesh of $24 \times 24 \times 8$.

We introduce the following acronyms:

FL/FD: FETI-DPH with the lumped/DtN preconditioner
FH: FETI-H with corner constraints
O0/O2: non-overlapping optimized Schwarz of zero/second order

[3] Though most of the overlapping methods in the literature are not in this form, we found by numerical experiments it may be cheaper in both time and memory.

OD/ON/OR: overlapping method with Dirichlet/Neumann/absorbing data
OO0/OO2: overlapping optimized Schwarz of zero/second order

For the overlapping methods, the overlapping region has a thickness of two mesh elements and the matching conditions are imposed on faces, edges and vertices, respectively, without repeats on any degrees of freedom. Due to the absence of relevant results, the parameters for the optimized Schwarz methods are not respecting overlapping (except OO0), coarse problem and medium heterogeneity. The plane waves used are along six directions that are normal to the *x*-*y*, *y*-*z* and *z*-*x* planes, respectively.

We found that all the methods outperform the direct solver in CPU time (see Table 1) on the $96 \times 96 \times 32$ mesh. We are interested in how the convergence of these methods depends on the *frequency* f in (1), the *mesh size* h, the *partition* $N_x \times N_y \times N_z$ or the subdomain size H and medium heterogeneity. At $f = 1$ the domain contains nine wavelength along the x-direction, which corresponds to the problem on the unit cube with the wavenumber 18π.

In the following tables, the numbers outside/inside parentheses are the iteration numbers with/without plane waves, respectively, and a bar is used instead of 200 when the maximum iteration number is reached. We use e/w to represent the number of elements per wavelength at the lowest velocity. The smallest iteration numbers among the non-overlapping methods and those among the overlapping methods are in bold. Note that for the FETI-DPH method with DtN preconditioner the amount of work per iteration is about 1.5 times that for the others, and construction of the preconditioner also leads to double LU factorizations in the setup stage.

In Tables 3 and 4, we increase the frequency with fh or f^3h^2 [1] kept constant.

Table 3. Dependence on the frequency (fh =constant)

f	FL	FD	FH	O0	O2	OD	OR	ON	OO0	OO2
					partition $3 \times 3 \times 1$					
$\frac{1}{4}$	6 (15)	**4** (8)	9 (15)	15 (21)	8 (14)	8 (20)	8 (12)	9 (20)	7 (15)	**6** (14)
$\frac{1}{2}$	15 (30)	**9** (12)	18 (33)	29 (34)	19 (20)	23 (34)	12 (15)	24 (37)	12 (17)	**11** (13)
1	44 (51)	**20** (23)	75 (93)	43 (48)	25 (25)	51 (58)	17 (17)	57 (66)	22 (25)	**14** (15)
					partition scaling with mesh: $H/h = 8$ (see also the first row for $f = \frac{1}{4}$)					
$\frac{1}{2}$	8 (46)	**5** (30)	10 (73)	17 (71)	10 (50)	14 (73)	11 (33)	21 (103)	**8** (55)	**8** (51)
1	9 (183)	**7** (-)	11 (-)	21 (-)	12 (-)	27 (-)	**15** (74)	152 (-)	16 (-)	**15** (-)
					partition scaling with mesh: $H/h = 16$ (see also the second row for $f = \frac{1}{2}$)					
1	39 (127)	32 (103)	74 (-)	59 (113)	**27** (39)	76 (171)	26 (38)	114 (-)	26 (53)	**22** (32)

We see that more iterations are usually needed for larger frequency except in the middle of Table 4.

In Table 5, the frequency is fixed and the mesh is refined. From the table, the iteration numbers with plane waves almost remain constant.

Table 4. Dependence on the frequency (f^3h^2=constant)

f	FL	FD	FH	O0	O2	OD	OR	ON	OO0	OO2	
partition $3\times3\times1$ (see also the first row in Table 3 for $f=0.25$)											
0.40	12 (25)	**6** (11)	14 (25)	30 (33)	18 (21)	18 (29)	11 (14)	19 (32)	**9** (15)	**9** (13)	t4.2
0.63	27 (41)	**11** (15)	33 (49)	37 (42)	25 (26)	38 (46)	16 (17)	39 (50)	15 (20)	**13** (14)	t4.3
partition scaling with mesh: $H/h=8$(see also the first row in Table 4 for f $=0.25$)											
0.40	7 (36)	**5** (23)	10 (54)	15 (58)	9 (40)	12 (60)	10 (29)	13 (73)	**7** (40)	**7** (40)	t4.4
0.63	7 (127)	**5** (100)	9 (149)	14 (156)	8 (112)	14 (160)	11 (65)	20 (-)	**7** (123)	**7** (117)	t4.5
partition scaling with mesh: $H/h=16$ (see also the first row for $f=0.40$)											
0.63	15 (89)	**8** (53)	18 (119)	43 (125)	18 (75)	33 (113)	16 (35)	36 (112)	**13** (75)	**13** (75)	t4.6

Table 5. Dependence on the mesh size ($f=\frac{1}{4}$)

e/w	FL	FD	FH	O0	O2	OD	OR	ON	OO0	OO2	
partition $3\times3\times1$ (see also the first row in Table 4 for $e/w=10$)											
20	10 (19)	**5** (9)	13 (20)	17 (26)	9 (17)	14 (28)	11 (15)	13 (27)	8 (16)	**6** (16)	t5.2
40	15 (25)	**6** (10)	18 (25)	21 (32)	11 (20)	21 (39)	15 (19)	19 (36)	9 (17)	**8** (17)	t5.3
partition $H/h=8$ (see also the first row in Table 4 for $e/w=10$)											
20	7 (21)	**5** (12)	10 (32)	14 (47)	8 (32)	10 (46)	9 (25)	10 (44)	7 (29)	**6** (30)	t5.4
40	6 (19)	**4** (13)	9 (36)	14 (92)	7 (63)	9 (90)	9 (46)	9 (91)	7 (56)	**6** (59)	t5.5
partition $H/h=16$ (see also the first row for $e/w=20$)											
40	11 (34)	**6** (15)	14 (47)	17 (60)	10 (38)	15 (63)	12 (28)	13 (52)	**7** (33)	**7** (35)	t5.6

Next, we compare the iteration numbers for different partitions with both the frequency and the mesh size fixed in Table 6. One can see that with plane waves

Table 6. Dependence on the partition

	FL	FD	FH	O0	O2	OD	OR	ON	OO0	OO2	
$\frac{H}{H_0}$	$f=\frac{1}{2}$, mesh and velocity $48\times48\times16$ and H_0 partition $3\times3\times1$										t6.2
1	15 (30)	**9** (12)	18 (33)	28 (35)	19 (21)	22 (34)	12 (15)	23 (37)	**11** (17)	**11** (14)	t6.3
$\frac{1}{2}$	8 (47)	**5** (30)	10 (73)	16 (72)	9 (51)	14 (75)	11 (34)	21 (105)	8 (62)	**7** (57)	t6.4
$\frac{1}{4}$	**4** (22)	**4** (21)	7 (48)	10 (95)	7 (72)	7 (97)	8 (52)	11 (-)	6 (83)	**5** (78)	t6.5
	$f=1$, mesh and velocity $96\times96\times32$ and H_0 partition $3\times3\times1$										t6.6
1	46 (54)	**22** (24)	79 (97)	45 (49)	26 (26)	54 (61)	17 (18)	60 (69)	22 (26)	**15** (16)	t6.7
$\frac{1}{2}$	43 (133)	35 (109)	82 (-)	63 (117)	**28** (40)	82 (176)	27 (39)	136 (-)	28 (56)	**24** (34)	t6.8
$\frac{1}{4}$	10 (184)	**8** (-)	14 (-)	26 (-)	16 (40)	32 (-)	**17** (-)	- (-)	25 (-)	22 (-)	t6.9
N_x	$f=1$, mesh and velocity $96\times96\times32$ and partition $N_x\times1\times1$										t6.10
8	117 (125)	79 (75)	171 (184)	66 (70)	**28** (28)	94 (99)	**23** (24)	100 (104)	51 (46)	**23** (25)	t6.11
16	184 (-)	192 (199)	- (-)	131 (137)	**45** (47)	- (-)	46 (47)	- (-)	72 (81)	**43** (45)	t6.12
32	- (-)	- (-)	- (-)	172 (173)	**87** (90)	- (-)	86 (90)	182 (88)	148 (136)	**84** (87)	t6.13

using more subdomains can both increase and decrease the iteration numbers. It is interesting that for the strip-wise partition only the methods based on transmission conditions (O0, O2, OR, OO0 and OO2) work reliably, though with substantial iteration numbers, and the plane waves do not help much.

Last, we study the influence of the heterogeneity in the velocity. The experiments are carried out on artificial velocity models to have high contrasts. The frequency is fixed as $f = \frac{1}{2}$. The lowest velocity is fixed as $c_{\min} = 1{,}500$ and different levels of highest velocity $c_{\max} = \rho c_{\min}$ are considered. It can be seen from Table 7 that the iteration numbers vary only little.

Table 7. Influence of medium heterogeneity

ρ	FL	FD	FH	O0	O2	OD	OR	ON	OO0	OO2
mesh $48 \times 48 \times 16$, partition $8 \times 1 \times 1$ and $c = c_{\min}, c_{\max}$ on subdomains										
1	58 (76)	37 (46)	83 (94)	60 (64)	**28** (29)	70 (81)	27 (26)	69 (79)	37 (44)	**24** (24)
10^2	28 (36)	42 (58)	30 (37)	37 (55)	**26** (31)	37 (53)	27 (29)	63 (75)	15 (26)	**13** (22)
10^4	32 (36)	49 (58)	33 (37)	45 (55)	**26** (31)	43 (53)	29 (30)	71 (75)	19 (26)	**17** (22)
as above except partition $6 \times 6 \times 2$										
1	9 (90)	**7** (62)	12 (124)	26 (79)	15 (39)	18 (97)	14 (35)	22 (117)	**10** (46)	12 (34)
10^2	12 (59)	**10** (104)	17 (51)	25 (78)	15 (46)	17 (67)	12 (34)	29 (100)	**8** (42)	9 (37)
10^4	14 (58)	**11** (104)	19 (51)	27 (79)	17 (47)	19 (68)	12 (34)	33 (100)	**8** (42)	10 (37)
mesh $48 \times 48 \times 16$, partition $1 \times 8 \times 1$ and $c = c_{\min}, c_{\max}$ on $8 \times 1 \times 1$ cells										
1	70 (81)	40 (50)	105 (114)	73 (75)	**27** (28)	74 (80)	28 (27)	62 (66)	34 (37)	**24** (24)
10^2	51 (59)	30 (34)	69 (84)	58 (67)	**26** (28)	56 (67)	**23** (26)	51 (59)	26 (28)	**23** (26)
10^4	52 (59)	30 (34)	70 (85)	58 (67)	**26** (28)	56 (68)	**23** (26)	51 (59)	26 (28)	**23** (26)
mesh $84 \times 84 \times 24$, partition $6 \times 6 \times 2$ and $c = c_{\min}, c_{\max}$ on $7 \times 7 \times 3$ cells										
1	12 (105)	**8** (65)	16 (144)	34 (96)	19 (41)	24 (121)	17 (37)	25 (111)	**12** (46)	15 (34)
10^2	10 (68)	**7** (34)	14 (107)	29 (109)	17 (48)	26 (111)	13 (45)	21 (106)	**11** (47)	12 (40)
10^4	11 (68)	**7** (34)	15 (107)	31 (109)	18 (48)	26 (110)	14 (45)	21 (107)	**11** (47)	12 (40)
mesh $48 \times 48 \times 16$, partition $6 \times 6 \times 2$ and c random constants on elements										
10^2	7 (16)	**5** (10)	10 (21)	14 (61)	9 (41)	14 (60)	11 (37)	12 (59)	**7** (35)	8 (38)
10^4	8 (15)	**6** (9)	11 (20)	12 (67)	8 (46)	14 (67)	15 (61)	25 (86)	**8** (39)	**8** (42)
as above except partition $3 \times 3 \times 1$										
1	22 (38)	**10** (16)	26 (45)	28 (37)	19 (21)	26 (36)	13 (15)	27 (36)	15 (21)	**12** (14)
10^2	11 (17)	**6** (8)	15 (20)	18 (33)	11 (21)	16 (35)	15 (23)	16 (42)	**7** (17)	8 (19)
10^4	12 (17)	**6** (8)	16 (21)	15 (39)	9 (24)	18 (40)	16 (31)	17 (52)	**8** (20)	9 (22)

4 Conclusions

For the SEG–SALT model on the cube domain, we get the following conclusions: among the non-overlapping methods, the FETI-DPH method with DtN preconditioner performs best in terms of iteration numbers. Among the overlapping methods, the optimized Schwarz method of second order is usually the best. With a fixed number of plane waves, all the methods can slow down for larger frequencies on properly refined meshes. They also deteriorate for fixed frequency on finer meshes, unless when using plane waves and more subdomains. A smaller subdomain size can both increase and decrease the iteration numbers, and the experiments indicate the existence of some optimal choice. For strip-wise partitions, only the methods based on transmission conditions work well, and plane waves do not help much. We also find the performance of all the method is only little affected by the heterogeneity in the velocity we considered, but other kinds of heterogeneity still need to be investigated.

Acknowledgments The authors thank Paul Childs for providing the velocity data of the geophysical SEG–SALT model. This work was partially supported by the University of Geneva. The second author was also partially supported by the NSFC Tianyuan Mathematics Youth Fund 10926134.

Bibliography

[1] Ivo Babuska, Frank Ihlenburg, Ellen T. Paik, and Stefan A. Sauter. A generalized finite element method for solving the Helmholtz equation in two dimensions with minimal pollution. *Comput. Methods Appl. Mech. Engrg.*, 128(3-4): 325–359, 1995.

[2] Yassine Boubendir, Xavier Antoine, and Christophe Geuzaine. New non-overlapping domain decomposition algorithm for Helmholtz equation. In *Twentieth International Conference on Domain Decomposition Methods*, page in this volume, 2011.

[3] Yassine Boubendir, Xavier Antoine, and Christophe Geuzaine. A quasi-optimal non-overlapping domain decomposition algorithm for the Helmholtz equation. *J. Comput. Phys.*, 231(2):262–280, 2012.

[4] Xiao-Chuan Cai, Mario A. Casarin, Frank W. Elliott, Jr., and Olof B. Widlund. Overlapping Schwarz algorithms for solving Helmholtz's equation. In *Domain decomposition methods, 10 (Boulder, CO, 1997)*, volume 218 of *Contemp. Math.*, pages 391–399. Amer. Math. Soc., Providence, RI, 1998.

[5] Bruno Després. Domain decomposition method and the Helmholtz problem. In *Mathematical and numerical aspects of wave propagation phenomena (Strasbourg, 1991)*, pages 44–52. SIAM, Philadelphia, PA, 1991.

[6] Olivier G. Ernst and Martin J. Gander. Why it is difficult to solve Helmholtz problems with classical iterative methods. In *Numerical Analysis of Multiscale Problems*. Durham LMS Symposium 2010, Springer Verlag, 2011.

[7] Charbel Farhat, Antonini Macedo, and Michel Lesoinne. A two-level domain decomposition method for the iterative solution of high frequency exterior Helmholtz problems. *Numer. Math.*, 85(2):283–308, 2000.

[8] Charbel Farhat, Philip Avery, Radek Tezaur, and Jing Li. FETI-DPH: a dual-primal domain decomposition method for acoustic scattering. *J. Comput. Acoust.*, 13(3):499–524, 2005.

[9] Martin J. Gander. Optimized Schwarz methods for Helmholtz problems. In *Domain decomposition methods in science and engineering (Lyon, 2000)*, Theory Eng. Appl. Comput. Methods, pages 247–254. Internat. Center Numer. Methods Eng. (CIMNE), Barcelona, 2002.

[10] Martin J. Gander, Frédéric Magoulès, and Frédéric Nataf. Optimized Schwarz methods without overlap for the Helmholtz equation. *SIAM J. Sci. Comput.*, 24 (1):38–60 (electronic), 2002.

[11] Martin J. Gander, Laurence Halpern, and Frédéric Magoulés. An optimized Schwarz method with two-sided Robin transmission conditions for the Helmholtz equation. *Int. J. Numer. Meth. Fluids*, 55(2):163–175, 2007.

[12] Souad Ghanemi. A domain decomposition method for Helmholtz scattering problems. In *Ninth International Conference on Domain Decomposition Methods*, pages 105–112, 1998.

[13] Murthy N. Guddati and Senganal Thirunavukkarasu. Improving the convergence rate of Schwarz methods for Helmholtz equation. In *Twentieth International Conference on Domain Decomposition Methods*, page in this volume, 2011.

[14] Jung-Han Kimn and Blaise Bourdin. Numerical implementation of overlapping balancing domain decomposition methods on unstructured meshes. In *Domain decomposition methods in science and engineering XVI*, volume 55 of *Lect. Notes Comput. Sci. Eng.*, pages 309–315. Springer, Berlin, 2007.

[15] Jung-Han Kimn and Marcus Sarkis. Restricted overlapping balancing domain decomposition methods and restricted coarse problems for the Helmholtz problem. *Comput. Methods Appl. Mech. Engrg.*, 196(8):1507–1514, 2007.

[16] An Leong and Olof B. Widlund. Extension of two-level Schwarz preconditioners to symmetric indefinite problems. Technical report, New York University, New York, NY, USA, 2008.

[17] Jing Li and Xuemin Tu. Convergence analysis of a balancing domain decomposition method for solving a class of indefinite linear systems. *Numer. Linear Algebra Appl.*, 16(9):745–773, 2009.

[18] Bruno Stupfel. Improved transmission conditions for a one-dimensional domain decomposition method applied to the solution of the Helmholtz equation. *J. Comput. Phys.*, 229(3):851–874, 2010.

[19] Shlomo Ta'asan. *Multigrid methods for highly oscillatory problems*. PhD thesis, Weizmann Institute of Science, Rehovot, Israel, 1984.