Robust Multilevel Restricted Schwarz Preconditioners and Applications*

Ernesto E. Prudencio[1] and Xiao-Chuan Cai[2]

[1] Advanced Computations Department, Stanford Linear Accelerator Center,
Menlo Park, CA 94025, prudenci@slac.stanford.edu

[2] Department of Computer Science, University of Colorado at Boulder, 430 UCB,
Boulder, CO 80309, cai@colorado.edu

Summary. We introduce a multi-level restricted Schwarz preconditioner with a special coarse-to-fine interpolation and show numerically that the new preconditioner works extremely well for some difficult large systems of linear equations arising from some optimization problems constrained by the incompressible Navier-Stokes equations. Performance of the preconditioner is reported for parameters including number of processors, mesh sizes and Reynolds numbers.

1 Introduction

There are two major families of techniques for solving Karush-Kuhn-Tucker (KKT, or optimality) Jacobian systems, namely the reduced space and the full space methods [2, 3, 12, 11]. When memory capability is an issue, reduced methods are preferred, although many sub-iterations might be needed to converge the outer-iterations and the parallel scalability is less ideal. As the processing speed and the memory capability of computers increase, full space methods become more popular because of their increased scalability. One of their main challenges, though, is how to handle the indefiniteness and ill-conditioning of those Jacobians. In addition, some of the solution components might present sharp jumps. Traditional multilevel preconditioning techniques do not work well because of the cross-mesh pollution; i.e., sharp jumps are smoothed out by inter-mesh operations.

We introduce a new multilevel restricted Schwarz preconditioner with a special coarse-to-fine interpolation and show numerically that it works extremely well for rather difficult large Jacobian systems arising from some

* The research was supported in part by the National Science Foundation, CCR-0219190 and ACI-0305666, and in part by the Department of Energy, DE-FC02-01ER25479.

optimization problems constrained by the incompressible Navier-Stokes equations. The preconditioner is not only scalable but also pollution free.

Many optimization problems constrained by PDEs can be written as

$$\begin{cases} \min\limits_{\mathbf{x}\in\mathbf{W}} \mathcal{F}(\mathbf{x}) \\ \text{s.t. } \mathbf{C}(\mathbf{x}) = \mathbf{0} \in \mathbf{Y}. \end{cases} \tag{1}$$

Here $\mathbf{W}$ and $\mathbf{Y}$ are normed spaces, $\mathbf{W}$ is the space of optimization variables, $\mathcal{F} : \mathbf{W} \to \mathbb{R}$ is the objective functional and $\mathbf{C} : \mathbf{W} \to \mathbf{Y}$ represents the PDEs. The associated Lagrangian functional $\mathcal{L} : \mathbf{W} \times \mathbf{Y}^* \to \mathbb{R}$ is defined as

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) \equiv \mathcal{F}(\mathbf{x}) + \langle \boldsymbol{\lambda}, \mathbf{C}(\mathbf{x}) \rangle_{\mathbf{Y}}, \quad \forall\ (\mathbf{x}, \boldsymbol{\lambda}) \in \mathbf{W} \times \mathbf{Y}^*,$$

where $\mathbf{Y}^*$ is the adjoint space of $\mathbf{Y}$, $\langle \cdot, \cdot \rangle_{\mathbf{Y}}$ denotes the duality pairing and variables $\boldsymbol{\lambda}$ are called Lagrange multipliers or adjoint variables. In many cases it is possible to prove that, if $\hat{\mathbf{x}}$ is a (local) solution of (1) then there exist Lagrange multipliers $\hat{\boldsymbol{\lambda}}$ such that $(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}})$ is a critical point of $\mathcal{L}$ [10]. So, with a discretize-then-optimize approach [9] and sufficient smoothness assumptions, a necessary condition for a solution of (1) is to solve the KKT system

$$\begin{pmatrix} \nabla_{\mathbf{x}} \mathcal{L} \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L} \end{pmatrix} = \begin{pmatrix} \nabla \mathcal{F} + [\boldsymbol{\nabla} \mathbf{C}]^T \boldsymbol{\lambda} \\ \mathbf{C} \end{pmatrix} = \mathbf{0}. \tag{2}$$

Each iteration of a Newton's method for (2) involves the Jacobian system

$$\begin{bmatrix} \boldsymbol{\nabla}_{\mathbf{xx}} \mathcal{L} & [\boldsymbol{\nabla} \mathbf{C}]^T \\ \boldsymbol{\nabla} \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{pmatrix} \mathbf{p}_{\mathbf{x}} \\ \mathbf{p}_{\boldsymbol{\lambda}} \end{pmatrix} = - \begin{pmatrix} \nabla_{\mathbf{x}} \mathcal{L} \\ \mathbf{C} \end{pmatrix}. \tag{3}$$

The paper is organized as follows. Section 2 introduces a preconditioner for (3), while in Section 3 we test it on some flow control problems and report its performance for combinations of parameters including number of processors, mesh sizes and Reynolds numbers. Final conclusions are given in Section 4.

2 Multilevel pollution removing restricted Schwarz

Schwarz methods can be used in one-level or multilevel approaches and, in each case, with a combination of additive and/or multiplicative algorithms [13]. They can be also used as linear [8] and nonlinear preconditioners [6].

Let Ω_h be a mesh of characteristic size $h > 0$, subdivided into non-overlapping subdomains Ω_j, $j = 1, \ldots, N_S$. Let $H > 0$ denote the characteristic diameter of $\{\Omega_j\}$ and let $\{\Omega_j^{'}\}$ be an overlapping partition with overlapping $\delta > 0$. From now on we only consider simple box domains, uniform meshes and simple box decompositions, i.e., all subdomains Ω_j and $\Omega_j^{'}$ are rectangular and made up of integral number of mesh cells. Let N and N_j denote the number of degrees of freedom associated to Ω_h and $\Omega_j^{'}$, respectively. Let $\mathbf{K}$ be a $N \times N$ matrix of a linear system

$$\mathbf{K}\mathbf{p} = \mathbf{b} \tag{4}$$

that needs to be solved during the application of an algorithm for the numerical solution of a discretized differential problem. Let d indicate the degree of freedom per mesh point. For simplicity let us assume that d is the same throughout the entire mesh. We define the $N_j \times N$ matrix $\mathbf{R}_j^\delta$ as follows: its $d \times d$ block element $(\mathbf{R}_j^\delta)_{\alpha,\beta}$ is either (a) an identity block if the integer indices $1 \leqslant \alpha \leqslant N_j/d$ and $1 \leqslant \beta \leqslant N/d$ are related to the same mesh point and this mesh point belongs to $\Omega_j^{'}$ or (b) a zero block otherwise. The multiplication of $\mathbf{R}_j^\delta$ with a $N \times 1$ vector generates a smaller $N_j \times 1$ vector by discarding all components corresponding to mesh points outside $\Omega_j^{'}$. The $N_j \times N$ matrix $\mathbf{R}_j^0$ is similarly defined, with the difference that its application to a $N \times 1$ vector also zeroes all those components corresponding to mesh points on $\Omega_j^{'} \setminus \Omega_j$. Let $\mathbf{B}_j^{-1}$ be either the inverse of or a preconditioner for $\mathbf{K}_j \equiv \mathbf{R}_j^\delta\ \mathbf{K}\ \mathbf{R}_j^{\delta^T}$. The one-level classical, right restricted (r-RAS) and left restricted (ℓ-RAS) additive Schwarz preconditioners for $\mathbf{K}$ are respectively defined as [5, 7, 8]

$$\mathbf{B}_{\delta\delta}^{-1} = \sum_{j=1}^{N_s} \mathbf{R}_j^{\delta^T} \mathbf{B}_j^{-1} \mathbf{R}_j^\delta,\ \mathbf{B}_{\delta 0}^{-1} = \sum_{j=1}^{N_s} \mathbf{R}_j^{\delta^T} \mathbf{B}_j^{-1} \mathbf{R}_j^0,\ \mathbf{B}_{0\delta}^{-1} = \sum_{j=1}^{N_s} \mathbf{R}_j^{0^T} \mathbf{B}_j^{-1} \mathbf{R}_j^\delta.$$

For the description of multilevel Schwarz preconditioners, let us use index $i = 0, 1, \ldots, L-1$ to designate any of the $L \geqslant 2$ levels. Let $\mathbf{I}_i$ denote the identity operator and, for $i > 0$, let $\mathbf{R}_i^T$ denote the interpolation from level $i-1$ to level i. Multilevel Schwarz preconditioners are obtained through the combination of one-level Schwarz preconditioners $\mathbf{B}_i^{-1}$ assigned to each level. Here we focus on multilevel preconditioners that use exact coarsest solvers $\mathbf{B}_0^{-1}$ and that can be seen as multigrid V-cycle algorithms [4] having Schwarz preconditioned Richardson working as the pre and the post smoother at each level $i > 0$, with $\mathbf{B}_{i,\mathrm{pre}}^{-1}$ preconditioning the $\mu_i \geqslant 0$ pre smoother iterations and $\mathbf{B}_{i,\mathrm{post}}^{-1}$ preconditioning the $\nu_i \geqslant 0$ post smoother iterations. Then, as iterative methods for (4), with $\mathbf{r}^{(\ell)}$ denoting the residual at iteration $\ell = 0, 1, 2, \ldots$, they can be described in the case $L = 2$ as

$$\mathbf{r}^{(\ell+1)} = (\mathbf{I}_1 - \mathbf{K}_1\mathbf{B}_{1,\mathrm{post}}^{-1})^{\nu_1}(\mathbf{I}_1 - \mathbf{K}_1\mathbf{R}_1^T\mathbf{B}_0^{-1}\mathbf{R}_1)(\mathbf{I}_1 - \mathbf{K}_1\mathbf{B}_{1,\mathrm{pre}}^{-1})^{\mu_1}\mathbf{r}^{(\ell)}. \tag{5}$$

Pollution removing interpolation constitutes a key procedure in our proposed multilevel preconditioner, due to the sharp jumps that often occur on the multiplier values over those regions of Ω_h where constraints are greatly affecting the behavior of the optimized system. Although the evidence of this discontinuity property of Lagrange multipliers is just empirical in our paper, it is consistent with their interpretation [11]: the value of a Lagrange multiplier at a mesh point gives the rate of change of the optimal objective function value w.r.t. to the respective constraint at that point.

In the case of the problem corresponding to Figure 2-b, for instance, an external force causes the fluid to move on a clockwise way and the boundary consists of rigid slip walls. The vertical walls greatly affect the overall vorticity throughout the domain, i.e., the value of the objective function, because they completely oppose the horizontal velocity component v_1. The values of λ_1 at the walls then reflect this situation. In contrast, λ_2 develops sharp jumps at the other two walls opposing v_2. In all our experiments the discontinuities are located only over the boundary and not around it, even for very fine meshes. Common coarse-to-fine interpolation techniques will then smooth the sharp jumps present in coarse solutions, with a gradual jumping, from interior mesh points towards boundary mesh points, appearing over those fine cells (elements, volumes) located inside coarse boundary ones. That is, the good correction information presided by the coarse solution is lost with a common interpolation. We refer to the smoothed jump as "pollution", in contrast to the "clean" sharp jump that is expected at the fine level as well.

We therefore propose a *modified* coarse-to-fine interpolation procedure that is based on a general and simple "removal of the pollution". Let $\mathbf{R}_i^T$ denote any unmodified interpolation procedure and $\boldsymbol{\mathcal{Z}}_i$ the operator that zeroes out, from a vector at level i, the Lagrange multipliers at all those mesh points with equations that have a greater influence on the objective function. For the case of PDEs describing physical systems, the number of such points are expected to be relatively small. Our modified interpolation is then expressed by

$$\mathbf{R}_{i,\mathrm{modif}}^T = \mathbf{R}_i^T - \boldsymbol{\mathcal{Z}}_i \mathbf{R}_i^T (\mathbf{I}_{i-1} - \boldsymbol{\mathcal{Z}}_{i-1}). \tag{6}$$

This procedure removes the smoothed contributions due to the coarse discontinuities, maintaining, at the fine level, the sharp jumps originally present at the coarse level. See Figure 1. Once $\mathbf{R}_i^T$ is available, (6) can be applied to *any* mesh in *any* dimension, with *any* number of components.

In the case of the problems in this paper, $\boldsymbol{\mathcal{Z}}_i$ zeroes the Lagrange multiplier components located at the boundary. In our tests we apply the modified interpolation only on the Lagrange multiplier components of coarse solutions, while the optimization variables continue to be interpolated with $\mathbf{R}_i^T$. Also, the restriction process remains $\mathbf{R}_i$ for all variables, that is, (5) becomes

$$\mathbf{r}^{(\ell+1)} = (\mathbf{I}_1 - \mathbf{K}_1 \mathbf{B}_{1,\mathrm{post}}^{-1})^{\nu_1} (\mathbf{I}_1 - \mathbf{K}_1 \mathbf{R}_{1,\mathrm{modif}}^T \mathbf{B}_0^{-1} \mathbf{R}_1)(\mathbf{I}_1 - \mathbf{K}_1 \mathbf{B}_{1,\mathrm{pre}}^{-1})^{\mu_1} \mathbf{r}^{(\ell)}.$$

The Lagrange multipliers reflect the eventual "discontinuity" on the type of equations (or their physical dimensions) between equations in different regions of $\overline{\Omega}$: in the case of the problems in Section 3, between those in Ω and those on $\partial\Omega$. From this point of view, it seems "natural" to apply different interpolations to the multiplier components depending on their location.

3 Numerical experiments

Our numerical experiments in this paper focus on optimal control problems [9], where the optimization space in (1) is generally given by $\mathbf{W}$=$\mathbf{S} \times \mathbf{U}$,

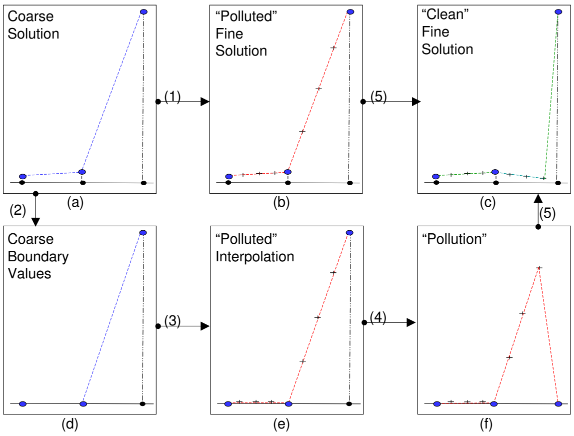

Fig. 1. Representation of the modified coarse-to-fine interpolation (6), with (a) input $\boldsymbol{\varphi}_{i-1}$ and (c) output $\boldsymbol{\varphi}_i$. The five steps are: (1) interpolation $\mathbf{R}_i^T\boldsymbol{\varphi}_{i-1}$, (2) coarse jump values $\tilde{\boldsymbol{\varphi}}_{i-1} = (\mathbf{I}_{i-1} - \boldsymbol{\mathcal{Z}}_{i-1})\boldsymbol{\varphi}_{i-1}$, (3) polluted $\tilde{\boldsymbol{\varphi}}_i = \mathbf{R}_i^T\tilde{\boldsymbol{\varphi}}_{i-1}$, (4) pollution isolation $\boldsymbol{\mathcal{Z}}_i\tilde{\boldsymbol{\varphi}}_i$, (5) pollution removal $\boldsymbol{\varphi}_i = \mathbf{R}_i^T\boldsymbol{\varphi}_{i-1} - \boldsymbol{\mathcal{Z}}_i\tilde{\boldsymbol{\varphi}}_i$.

with $\mathbf{S}$ being the state space and $\mathbf{U}$ the control space. Upon discretization, one has $n{=}n_s{+}n_u$, where n_s (n_u) is the number of discrete state (control) variables. More specifically, we treat the boundary control of two-dimensional steady-state incompressible Navier-Stokes equations in the velocity-vorticity formulation: $\mathbf{v} = (v_1, v_2)$ is the velocity and ω is the vorticity. Let $\Omega \subset \mathbb{R}^2$ be an open and bounded smooth domain, Γ its boundary, $\boldsymbol{\nu}$ the unit outward normal vector along Γ and $\mathbf{f}$ a given external force defined in Ω. Let $L^2(\Omega)$ and $L^2(\Gamma)$ be the spaces of square Lebesgue integrable functions in Ω and Γ respectively. The problems consist of finding $(\mathbf{s}, \mathbf{u}) = (v_1, v_2, \omega, u_1, u_2) \in L^2(\Omega)^3 \times L^2(\Gamma)^2 = \mathbf{S} \times \mathbf{U}$ such that the minimization

$$\min_{(\mathbf{s},\mathbf{u})\in\mathbf{S}\times\mathbf{U}} \mathcal{F}(\mathbf{s}, \mathbf{u}) = \frac{1}{2}\int_\Omega \omega^2 \, d\Omega + \frac{c}{2}\int_\Gamma \|\mathbf{u}\|_2^2 \, d\Gamma \tag{7}$$

is achieved subject to the constraints

$$\left\{ \begin{array}{ll} -\Delta v_1 - \frac{\partial \omega}{\partial x_2} & = 0 \ \text{ in } \Omega, \\ -\Delta v_2 + \frac{\partial \omega}{\partial x_1} & = 0 \ \text{ in } \Omega, \\ -\Delta \omega + Re \ v_1 \frac{\partial \omega}{\partial x_1} + Re \ v_2 \frac{\partial \omega}{\partial x_2} - Re \ \text{curl} \ \mathbf{f} & = 0 \ \text{ in } \Omega, \\ \mathbf{v} - \mathbf{u} & = \mathbf{0} \ \text{ on } \Gamma, \\ \omega + \frac{\partial v_1}{\partial x_2} - \frac{\partial v_2}{\partial x_1} & = 0 \ \text{ on } \Gamma, \\ \int_\Gamma \mathbf{v} \cdot \boldsymbol{\nu} \, d\Gamma & = 0, \end{array} \right. \tag{8}$$

where $\text{curl}\ \mathbf{f} = -\partial f_1/\partial x_2 + \partial f_2/\partial x_1$. The parameter $c > 0$ is used to adjust the relative importance of the control norms on achieving the minimization, so indirectly constraining their sizes. The physical objective in (7)-(8) is the minimization of turbulence [9]. The last constraint is due to the mass conservation law, making $m \neq n_s$ and causing the complexity Jacobian computation to increase, since non-neighboring mesh points become coupled by the integral. We restrict our numerical experiments to *tangential* boundary control problems, i.e., $\mathbf{u} \cdot \boldsymbol{\nu} = 0$ on Γ, so that $m = n_s$.

Here we only report tests for $\Omega = (0,1) \times (0,1)$, $c = 10^{-2}$ and $\mathbf{f} = (f_1, f_2) = \big(-\sin^2(\pi x_1)\,\cos(\pi x_2)\,\sin^2(\pi x_2), \sin^2(\pi x_2)\,\cos(\pi x_1)\,\sin^2(\pi x_1)\big)$. For comparison, we solve simulation problems with $\mathbf{v} \cdot \boldsymbol{\nu} = 0$ and $\partial \mathbf{v}/\partial \boldsymbol{\nu} = 0$ on Γ.

We performed tests on a cluster of Linux PCs and developed our software using the Portable, Extensible Toolkit for Scientific Computing (PETSc) from Argonne National Laboratory [1]. Table 1 shows the efficacy of the modified interpolation process, which performs much better than the unmodified one, causing the two-level preconditioner to outperform the one-level preconditioner. Table 2 shows the flexibility of the two-level preconditioner, which provides a similar average number of Krylov iterations throughout all seven situations in the table. Figure 2-*a* shows the controlled velocity field: the movement near the boundary is less intense. Figures 2-*c* and 2-*d* clearly show the stabilization on the average number of Krylov iterations provided by the two-level preconditioner with modified interpolation. The one-level preconditioner fails with 100 processors for $Re = 250$ and $Re = 300$.

Table 1. Resulted average number $\overline{\ell}$ of Krylov iterations per Newton iteration with Re=250, right preconditioned GMRES, a 280×280 mesh ($631,688$ variables), 49 processors, relative overlapping $\delta/H = 1/4$ and a 70×70 coarse mesh, for different combinations of number L of levels, linear interpolation type, number σ of pre and post smoother iterations, and RAS preconditioner.

L	Linear Interpolation Type	σ	RAS preconditioner	
			ℓ-RAS	r-RAS
1	–	–	$\overline{\ell} = 336$	$\overline{\ell} = 973$
2	Unmodified	1	$\overline{\ell} = 1,110$	$\overline{\ell} = 1,150$
2	Unmodified	2	$\overline{\ell} = 356$	$\overline{\ell} = 222$
2	Modified	1	$\boldsymbol{\overline{\ell} = 21}$	$\boldsymbol{\overline{\ell} = 28}$

4 Conclusions

We have developed a multilevel preconditioner for PDE-constrained optimization that showed a robust performance when tested on some boundary flow control problems. Our main contribution consisted of the combination of a general multigrid V-cycle preconditioner with (1) RAS preconditioned

Table 2. Resulted average number $\overline{\ell}$ of Krylov iterations per Newton iteration with *Re*=300, right preconditioned GMRES and a 70 × 70 coarse mesh, for different situations of number N_p of processors and mesh size. To each situation corresponds a combination of the number σ of Richardson iterations, the RAS preconditioner and the relative overlapping δ/H used in the pre and post smoothers. The number of variables is $2,517,768$ in the case of finest mesh.

N_p	$\frac{\delta}{H}$	140×140	280×280	560×560
25	$\frac{1}{4}$	$\sigma = 1$; r-RAS; $\boldsymbol{\overline{\ell} = 20}$	$\sigma = 1$; r-RAS; $\boldsymbol{\overline{\ell} = 23}$	–
49	$\frac{1}{2}$	$\sigma = 1$; r-RAS; $\boldsymbol{\overline{\ell} = 18}$	$\sigma = 1$; r-RAS; $\boldsymbol{\overline{\ell} = 21}$	–
100	$\frac{1}{2}$	$\sigma = 1$; ℓ-RAS; $\boldsymbol{\overline{\ell} = 18}$	$\sigma = 1$; ℓ-RAS; $\boldsymbol{\overline{\ell} = 25}$	$\sigma = 2$; r-RAS; $\boldsymbol{\overline{\ell} = 27}$

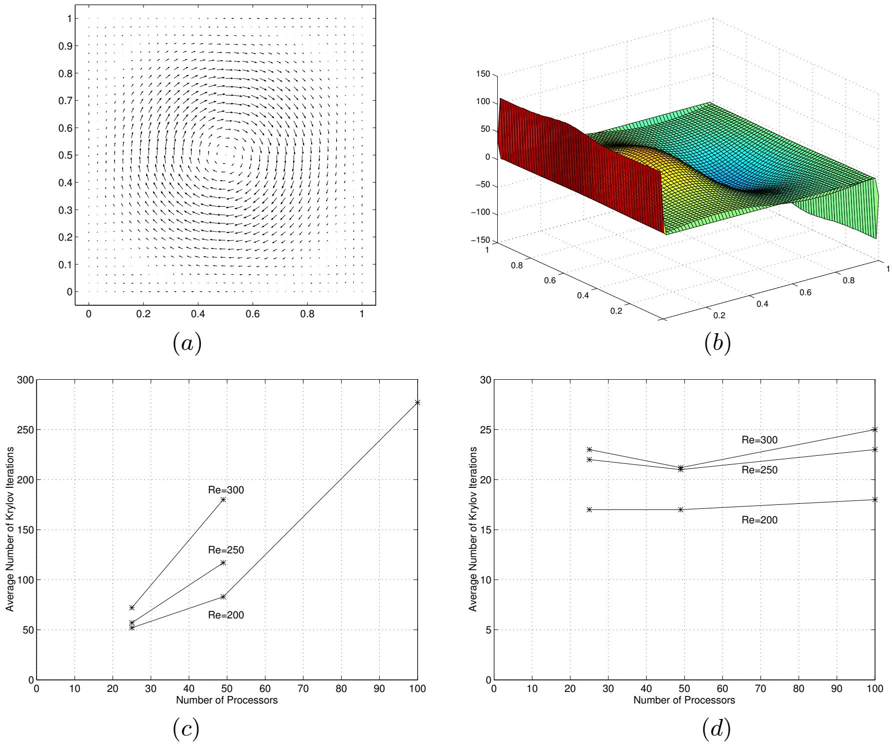

Fig. 2. Information on cavity flow problems: (a) controlled velocity field with *Re* = 200 and (b) corresponding Lagrange multiplier λ_1; results for (c) one-level and (d) two-level preconditioner with right-preconditioned GMRES, a 280 × 280 mesh ($631,688$ variables), and a 70 × 70 coarse mesh.

Richardson smoothers and (2) a modified interpolation procedure that removes the pollution often generated by the application of common interpolation techniques to the Lagrange multipliers. Such combination was key for the success of the two-level method in our experiments and the consequent improvement over the one-level method, handling flow control problems with higher Reynolds number, finer meshes and more processors. Surprisingly, RAS preconditioners performed much better than the classical ones.

Multilevel Schwarz is a flexible algorithm, and since it is also fully coupled (in contrast to operator-splitting, Schur complement, reduced space techniques), the original sparsity of a discretized PDE constrained optimization problem is maintained throughout its entire application and fewer sequential preconditioning steps are needed. We expect this preconditioner to have wide applications in other areas of computational science and engineering.

References

1. S. Balay, K. Buschelman, W. D. Gropp, D. Kaushik, M. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, *Portable, Extensible Toolkit for Scientific Computation (PETSc) home page.* http://www.mcs.anl.gov/petsc, 2004.
2. G. Biros and O. Ghattas, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization, part I: The Krylov-Schur solver*, SIAM J. Sci. Comput., (to appear).
3. ———, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization, part II: The Lagrange-Newton solver and its application to optimal control of steady viscous flows*, SIAM J. Sci. Comput., (to appear).
4. W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, SIAM, 2nd ed., 2000.
5. 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 (2003), pp. 1209–1231.
6. X.-C. Cai and D. E. Keyes, *Nonlinearly preconditioned inexact Newton algorithms*, SIAM J. Sci. Comput., 24 (2002), pp. 183–200.
7. X.-C. Cai and M. Sarkis, *A restricted additive Schwarz preconditioner for general sparse linear systems*, SIAM J. Sci. Comput., 21 (1999), pp. 792–797.
8. M. Dryja and O. Widlund, *Domain decomposition algorithms with small overlap*, SIAM J. Sci. Comp., 15 (1994), pp. 604–620.
9. M. D. Gunzburger, *Perspectives in Flow Control and Optimization*, SIAM, 1st ed., 2003.
10. A. D. Ioffe and V. M. Tihomirov, *Theory of Extremal Problems*, North-Holland Publishing Company, 1979. Translation from Russian edition ©NAUKA, Moscow, 1974.
11. E. E. Prudencio, *Parallel Fully Coupled Lagrange-Newton-Krylov-Schwarz Algorithms and Software for Optimization Problems Constrained by Partial Differential Equations*, PhD thesis, Department of Computer Science, University of Colorado at Boulder, 2005.
12. E. Prudencio, R. Byrd, and X. C. Cai, *Parallel full space SQP Lagrange-Newton-Krylov-Schwarz algorithms for PDE-constrained optimization problems*, (submitted).
13. A. Toselli and O. Widlund, *Domain Decomposition Methods - Algorithms and Theory*, Springer-Verlag, 2005.