AN ALGORITHM FOR SOLVING NON-SYMMETRIC SYSTEMS ARISING FROM SMOOTHER FDM

J. Haslinger

Dept. of Num. Math., Charles University, Sokolovská 83, 186 75 Prague, Czech Republic

hasling@karlin.mff.cuni.cz

T. Kozubek

Dept. of Appl. Math., VŠB-TU, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic

tomas.kozubek@vsb.cz

R. Kučera

Dept. of Math. and Desc. Geom., VŠB-TU, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic

radek.kucera@vsb.cz

The contribution deals with a fast method for solving large scale algebraic saddle-point systems arising from a new fictitious domain formulation of elliptic boundary value problems.

Keywords: saddle-point system, fictitious domain method, Schur complement, orthogonal projectors, BiCGSTAB algorithm, multigrid

1 Introduction

The classical fictitious domain method (FDM) enforces boundary conditions in PDE's by Lagrange multipliers defined on the boundary γ of the original domain ω. Therefore the computed solution has a singularity on γ that can result in an intrinsic error. The basic idea of our modification consists in introducing new control variables (instead of Lagrange multipliers) defined on an auxiliary boundary Γ located outside of $\overline{\omega}$ [1]. In this approach, the singularity is moved away from $\overline{\omega}$ so that the computed solution is smoother in ω and the discretization error has a significantly higher rate of convergence in ω.

The respective finite element discretization leads typically to a non-symmetric saddle-point system

$$\begin{pmatrix} A & B_1^\top \\ B_2 & 0 \end{pmatrix} \begin{pmatrix} \hat{u} \\ \lambda \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}, \tag{1}$$

where an $(n \times n)$ diagonal block A is possibly singular and $(m \times n)$ off-diagonal blocks B_1, B_2 have full row-rank and they are highly sparse. Moreover, m is much smaller than n and the defect l of A, i.e., $l = n - rank\, A$, is much smaller than m. For solving such systems, it is convenient to use a method based on the Schur complement reduction.If A is singular, the reduced system has again a saddle-point structure. Fortunately after applying orthogonal projectors, we obtain an equation in terms of λ only that can be efficiently solved by the projected variant of the BiCGSTAB algorithm [1]. This procedure generalizes ideas used in FETI domain decomposition methods, in which A is symmetric, positive semidefinite and $B_1 = B_2$.

2 Fictitious domain method

Let us consider a non-homogeneous Dirichlet boundary value problem:

$$-\Delta u = f \quad \text{in } \omega, \qquad u = g \quad \text{on } \gamma, \tag{2}$$

where $\omega \subset R^2$ is a bounded domain with the Lipschitz boundary γ, $f \in L^2_{loc}(R^2)$ and $g \in H^{1/2}(\gamma)$ are given data.

Let $\Xi \supset \overline{\omega}$ be another domain with the Lipschitz boundary Γ, $\text{dist}(\Gamma, \gamma) = \delta$ for some $\delta > 0$ given. Finally, let $\Omega \supset \overline{\Xi}$ be a fictitious domain (a box, e.g.).We define a problem:

$$\left.\begin{array}{l} \textit{Find } (\hat{u}, \lambda) \in H^1_0(\Omega) \times H^{-1/2}(\Gamma) \textit{ such that} \\ \displaystyle\int_\Omega \nabla \hat{u} \cdot \nabla v \, dx = \int_\Omega f v \, dx + \langle \lambda, v \rangle_\Gamma \quad \forall v \in H^1_0(\Omega), \\ \langle \hat{u}, \mu \rangle_\gamma = \langle g, \mu \rangle_\gamma \quad \forall \mu \in H^{-1/2}(\gamma). \end{array}\right\} \tag{3}$$

If γ and Γ are smooth enough then the first component of the solution to (3) satisfies $\hat{u} \in H^{3/2-\epsilon}(\Omega)$ $\forall \epsilon > 0$, while its restrictions are smoother $\hat{u}_{|\Xi} \in H^2(\Xi)$, $\hat{u}_{|\Omega \setminus \overline{\Xi}} \in H^2(\Omega \setminus \overline{\Xi})$. It means that the singularity of $\hat{u}$ is located on Γ, where generally a non-zero jump of the normal derivative $\frac{\partial \hat{u}}{\partial \nu}$ occurs. Since Γ has a positive distance from γ, one can expect that our variant of FDM will increase the convergence rate of computed solutions in ω.

3 Algorithm

Let us return to the system (1) resulting from a finite element discretization of (3), in which we use same notation for the discrete analogies of $\hat{u}$, λ, f and g. Our algorithm is based on the *generalized Schur complement* of A in (1) that is defined by

$$\mathcal{S} = \begin{pmatrix} -B_2 A^\dagger B_1^\top & B_2 N \\ M^\top B_1^\top & 0 \end{pmatrix},$$

where $A^\dagger$ is a generalized inverse to A and columns of N and M span the null-spaces $\mathbb{N}(A)$ and $\mathbb{N}(A^\top)$, respectively.

Theorem 1 [1] *Let us assume that (1) has a unique solution. Its second component λ is the first component of a solution to*

$$\begin{pmatrix} F & G_1^\top \\ G_2 & 0 \end{pmatrix} \begin{pmatrix} \lambda \\ \alpha \end{pmatrix} = \begin{pmatrix} d \\ e \end{pmatrix}, \tag{4}$$

where $F := B_2 A^\dagger B_1^\top, G_1 := -N^\top B_2^\top, G_2 := -M^\top B_1^\top, d := B_2 A^\dagger f - g$ and $e := -M^\top f$. The first component $\hat{u}$ is given by the formulae

$$\hat{u} = A^\dagger (f - B_1^\top \lambda) + N\alpha.$$

Let us point out that (4) is formally the same saddle-point system as (1), but its size is considerably smaller. We will modify it by two orthogonal projectors

$$P_1 := I - G_1^\top (G_1 G_1^\top)^{-1} G_1, \qquad P_2 := I - G_2^\top (G_2 G_2^\top)^{-1} G_2,$$

on the null-spaces $\mathbb{N}(G_1)$, $\mathbb{N}(G_2)$, respectively. The following results are keys to the algorithm.

Lemma 1 [1] *The linear operator $P_1 F : \mathbb{N}(G_2) \mapsto \mathbb{N}(G_1)$ is invertible.*

Theorem 2 [1] *Let $\lambda_{\mathbb{N}} \in \mathbb{N}(G_2)$, $\lambda_{\mathbb{R}} \in \mathbb{R}(G_2^\top)$. Then $\lambda = \lambda_{\mathbb{N}} + \lambda_{\mathbb{R}}$ is the first component of a solution to (4) iff*

$$\lambda_{\mathbb{R}} = G_2^\top (G_2 G_2^\top)^{-1} e$$

and

$$P_1 F \lambda_{\mathbb{N}} = P_1 (d - F\lambda_{\mathbb{R}}).$$

The second component α is given by

$$\alpha = (G_1 G_1^\top)^{-1} G_1 (d - F\lambda).$$

Algorithm: Projected Schur Complement Method (PSCM)

Step 1.a:	Assemble $G_1 = -N^\top B_2^\top$, $G_2 = -M^\top B_1^\top$, $d = B_2 A^\dagger f - g$ and $e = -M^\top f$.
Step 1.b:	Assemble $H_1 = (G_1 G_1^\top)^{-1}$ and $H_2 = (G_2 G_2^\top)^{-1}$.
Step 1.c:	Assemble $\lambda_{\mathbb{R}} = G_2^\top H_2 e$.
Step 1.d:	Assemble $\tilde{d} = P_1 (d - F\lambda_{\mathbb{R}})$.
Step 1.e:	Solve the equation $P_1 F \lambda_{\mathbb{N}} = \tilde{d}$ on $\mathbb{N}(G_2)$.
Step 1.f:	Compute $\lambda = \lambda_{\mathbb{N}} + \lambda_{\mathbb{R}}$.
Step 2:	Compute $\alpha = H_1 G_1 (d - F\lambda)$.
Step 3:	Compute $\hat{u} = A^\dagger (f - B_1^\top \lambda) + N\alpha$.

The heart of the algorithm consists in Step 1.e. Its solution can be computed by a *projected* Krylov subspace method. The projected BiCGSTAB algorithm [1] can be derived from the non-projected one by choosing an initial iterate $\lambda_{\mathbb{N}}^0$ on $\mathbb{N}(G_2)$, projecting the initial residual in $\mathbb{N}(G_2)$ and replacing the operator $P_1 F$ by its projected version $P_2 P_1 F$. Finally, let us point out that convergence of the projected BiCGSTAB algorithm can be accelerated by a multigrid technique.

4 Numerical experiments

Let ω be the ellipse, $\omega \equiv \{(x,y) \in R^2 | \ (x-0.5)^2/0.4^2 + (y-0.5)^2/0.2^2 < 1\}$, and the fictitious domain $\Omega = (0,1) \times (0,1)$. We will assume that the right hand-sides f and g in (2) are chosen appropriately to the exact solution $\hat{u}_{ex}(x,y) = 100\left((x-0.5)^3 - (y-0.5)^3\right) - x^2$.

The space $H_0^1(\Omega)$ in (3) is replaced by $H_{per}^1(\Omega)$ enabling us to use the Fourier direct method [2] to compute actions of $A^\dagger$, where A is the positive semidefinite discrete Laplacian resulting from the discretization of $H_{per}^1(\Omega)$ by piecewise bilinear functions defined on a rectangulation of Ω with a stepsize h. The spaces $H^{-1/2}(\Gamma)$ and $H^{-1/2}(\gamma)$ are approximated by piecewise constant functions defined on partitions of polygonal approximations of Γ and γ, respectively.

In Table 1, we report the errors of the approximate solutions with respect to the stepsize h in the $H^1(\omega)$-norm together with the number of BiCGSTAB iterations. We compare the classical FDM based on Lagrange multipliers and our modification (3), in which the auxiliary boundary Γ arises by shifting γ in the direction of the outward normal vector with $\delta = 8h$. From the computed errors, we determine the convergence rates (the last row of the table) that are considerably higher for modified case.

Table 1: Comparisons of the methods.

	Classical		Modified		Modified+Multigrid	
Step h	Iters.	$\mathrm{Err}_{H^1(\omega)}$	Iters.	$\mathrm{Err}_{H^1(\omega)}$	Iters.	$\mathrm{Err}_{H^1(\omega)}$
1/128	8	1.9647e+0	13	1.6878e-2	11	1.8988e-2
1/256	9	1.2884e+0	25	7.7891e-3	13	7.6303e-3
1/512	12	8.6517e-1	40	4.0160e-3	19	3.8638e-3
1/1024	18	6.0510e-1	58	1.9098e-3	21	1.7758e-3
1/2048	25	4.4015e-1	86	9.9299e-4	31	9.8213e-4
Conv. rates:		0.54		1.02		1.07

Acknowledgement

This work has been supported by the National Program of Research "Information Society" under project 1ET400300415, by the grant IAA1075402 of the Grant Agency of the Czech Academy of Sciences and by the Research Project MSM6198910027 of the Czech Ministry of Education.

References

[1] J. Haslinger, T. Kozubek, R. Kučera and G. Peichl: *Projected Schur complement method for solving non-symmetric saddle-point systems arising from fictitious domain approaches.* Num. Lin. Algebra Appl. 14(9) (2007), 713-739.

[2] R. Kučera: *Complexity of an algorithm for solving saddle-point systems with singular blocks arising in wavelet-Galerkin discretizations.* Appl. Math. 50(3) (2005), 291-308.