A New Kind of Multilevel Solver for Second Order Steklov-Poincaré Operators

Qiya Hu

LSEC and Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, P.R. China. hqy@lsec.cc.ac.cn

Summary. In this paper we are concerned with the construction of a preconditioner for the Steklov-Poincaré operator arising from a non-overlapping domain decomposition method for second-order elliptic problems in three-dimensional domains. We first propose a new kind of multilevel decomposition of the finite element space on the interface associated with a general quasi-uniform triangulation. Then, we construct a multilevel preconditioner for the underlying Steklov-Poincaré operator. The new multilevel preconditioner enjoys optimal computational complexity, and almost optimal convergence rate.

1 Introduction

The construction of domain decomposition preconditioners has been investigated in various ways in the literature, see, for example, [7]. This kind of preconditioner involves a set of local solvers (Steklov-Poincaré or Poincaré-Steklov operators), which result in dense stiffness matrices. It seems difficult to design cheap inexact solvers (preconditioners) for Steklov-Poincaré operators, unless the underlying triangulation has some particular structures (refer to [8]).

In the present paper, we propose a new kind of multilevel technique for preconditioning Steklov-Poincaré operators. The two main ingredients of this technique are the introduction of a multilevel *domain* decomposition for each local interface, and the construction of a series of coarse solvers associated with such decomposition. One of the main differences between the new method and the traditional multilevel one is that a series of refined grids is unnecessary for the new method (compare [5, 6] and [9]). It will be shown that the new multilevel method has almost optimal convergence and optimal computational complexity.

The new idea advanced in this paper can be extended to some other non-overlapping domain decomposition methods. For example, we can use the new technique to develop a class of substructuring methods with inexact solvers (refer to [4]).

2 Preliminaries

Let Ω be a bounded polyhedron in $\mathbb{R}^3$. Consider the model problem

$$\begin{cases} -div(a\nabla u) = f, & in \ \ \Omega, \\ u = 0, & on \ \ \partial\Omega, \end{cases} \tag{1}$$

where the coefficient $a \in L^\infty(\Omega)$ is a positive function.

Let $\mathcal{T}_h = \{\tau_i\}$ be a regular and quasi-uniform triangulation of Ω with $\tau_i's$ being non-overlapping simplexes of size h. The set of nodes of $\mathcal{T}_h$ is denoted by $\mathcal{N}_h$. Then, let $V_h(\Omega)$ be the piecewise linear finite element subspace of $H_0^1(\Omega)$ associated with $\mathcal{T}_h$:

$$V_h(\Omega) = \{v \in H_0^1(\Omega) : \ v|_\tau \in \mathbb{P}_1 \ \ \forall \tau \in \mathcal{T}_h\},$$

where $\mathbb{P}_1$ is the space of linear polynomials. The finite element approximation of (1) is: find $u_h \in V_h(\Omega)$ such that

$$(a\nabla u_h, \ \nabla v_h) = (f, v_h), \quad \forall v_h \in V_h(\Omega). \tag{2}$$

We will apply a non-overlapping domain decomposition method to solve (2). For the ease of notation, we consider only the case with two subdomains (see [4] for the general case).

Let Ω be decomposed into the union of two polyhedrons Ω_1 and Ω_2, which can be written as the union of some elements in $\mathcal{T}_h$, and satisfy $\Omega_1 \cap \Omega_2 = \emptyset$. Without loss of generality, we assume that the coefficient $a(p)$ is a piecewise constant function, and that each subdomain Ω_k is chosen such that $a(p)$ is equal to a constant a_k in Ω_k $(k = 1, 2)$. Set

$$V_h(\Omega_k) = \{v|_{\Omega_k} : \ \forall v \in V_h(\Omega)\} \quad (k = 1, 2).$$

We denote by Γ the common face of Ω_1 and Ω_2 (i.e., $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$), and we define

$$V_h(\Gamma) = \{v|_\Gamma : \ \forall v \in V_h(\Omega)\}.$$

Let $\varphi_h = u_h|_\Gamma$ denote the Dirichlet interface unknown. After eliminating the interior variables from (2), one gets the interface equation (see [7] for the details)

$$S_h\varphi_h = g_h. \tag{3}$$

In the case of two subdomains, the operator S_h is the discrete Steklov-Poincaré operator. It is easy to see that S_h results in a dense stiffness matrix.

In the following, we propose a new technique for preconditioning S_h based on a multilevel domain decomposition for Γ.

3 Multilevel Decompositions for $V_h(\Gamma)$

This section is devoted to establishing a stable multilevel decomposition of $V_h(\Gamma)$ based on a multilevel domain decomposition of Γ.

3.1 Multilevel Decomposition for Γ

The sketch of the multilevel decomposition can be described as follows. We first decompose Γ into the union of several non-overlapping polygons, and then further decompose each resulting polygon into the union of several smaller non-overlapping polygons. We can repeat this process such that each polygon generated by the final decomposition contains only a few nodes.

For convenience, a set of *closed* polygons on the same plane is called *non-overlapping* if the intersection of two neighboring polygons of this set is a common edge or vertex of the two polygons. Let J and m_k $(k = 1, \cdots, J)$ be given positive integers, and set $M_k = m_1 \cdots m_k$, for $k = 1, \cdots, J$.

The first-level decomposition. Decompose Γ into the union of non-overlapping closed polygons $\Gamma_1^{(1)}, \ldots, \Gamma_{m_1}^{(1)}$ in the standard way. We assume that all the polygons $\Gamma_r^{(1)}$ have almost the same "size" d_1.

Successively continuing this procedure, we get a hierarchical decompositions of Γ.

The second-level decomposition. Let each $\Gamma_r^{(1)}$ be further decomposed into the union of m_2 non-overlapping closed sub-polygons of $\Gamma_r^{(1)}$.

The k-level decomposition for $2 \le k \le J$. After generating Γ_r^{k-1} at the $(k-1)$-level decomposition, we decompose each $\Gamma_r^{(k-1)}$ into the union of m_k non-overlapping sub-polygons.

Finally, we get the multilevel decomposition for Γ

$$\Gamma = \bigcup_{r=1}^{m_1} \Gamma_r^{(1)} = \bigcup_{r=1}^{M_2} \Gamma_r^{(2)} = \cdots = \bigcup_{r=1}^{M_J} \Gamma_r^{(J)}.$$

For a fixed k, the closed sub-polygons $\Gamma_r^{(k)}$ $(r = 1, \cdots, M_k)$ satisfy the following conditions:

(a) each $\Gamma_r^{(k)}$ has size d_k for some $d_k \in (h, 1)$;

(b) the union of all $\Gamma_r^{(k)}$ $(r = 1, \cdots, M_k)$ constitutes a non-overlapping decomposition for Γ.

Remark 1. Each $\Gamma_r^{(k)}$ may not be the union of some elements of Γ, so the multilevel decomposition described above can be constructed in a simple manner. Note that there is no extra restriction on the triangulation on Γ (in fact the subdivision of the interface Γ does not relate to the triangulation).

3.2 Multilevel Decomposition for $V_h(\Gamma)$

The desired multilevel decomposition involves a set of small local subspaces and a series of coarse subspaces.

Small local subspaces. Let φ_Γ^p denote the nodal basis function of $V_h(\Gamma)$ associated with the node p on Γ. Set

$$V_h(\Gamma_r^{(J)}) = span\{\varphi_\Gamma^p : \ p \in \Gamma_r^{(J)}\} \quad (r = 1, \cdots, M_J).$$

Coarse subspaces. For convenience, define $M_0 = 1$ and $\Gamma_1^{(0)} = \Gamma$. For $k < J$, let $\mathcal{F}_{\Gamma_r^{(k)}}$, $\mathcal{E}_{\Gamma_r^{(k)}}$ and $\mathcal{V}_{\Gamma_r^{(k)}}$ denote respectively the set of the m_{k+1} sub-polygons, the set of the edges and the set of vertices generated by the $(k+1)$-th level decomposition

$$\Gamma_r^{(k)} = \bigcup_{l=1}^{m_{k+1}} \Gamma_{m_{k+1}(r-1)+l}^{(k+1)} \quad (r = 1, \cdots, M_k).$$

For a sub-polygon $\mathrm{F} \in \mathcal{F}_{\Gamma_r^{(k)}}$, set $\mathrm{F}^{in} = \mathrm{F} \backslash \partial \mathrm{F}$ and define the sub-polygon basis $\varphi_{\mathrm{F}} \in V_h(\Gamma)$ by [1]

$$\varphi_{\mathrm{F}}(p) = \begin{cases} 1, & \text{if } \ p \in \mathrm{F}^{in} \cap \mathcal{N}_h, \\ 0, & \text{if } \ p \in (\Gamma \backslash \mathrm{F}^{in}) \cap \mathcal{N}_h. \end{cases}$$

When an edge $\mathrm{E} \in \mathcal{E}_{\Gamma_r^{(k)}}$ contains some nodes, we define the edge basis $\varphi_{\mathrm{E}} \in V_h(\Gamma)$ by

$$\varphi_{\mathrm{E}}(p) = \begin{cases} 1, & \text{if } \ p \in \mathrm{E} \cap \mathcal{N}_h, \\ 0, & \text{if } \ p \in (\Gamma \backslash \mathrm{E}) \cap \mathcal{N}_h. \end{cases}$$

Similarly, when a vertex $\mathrm{V} \in \mathcal{V}_{\Gamma_r^{(k)}}$ is just a node, we define the vertex basis $\varphi_{\mathrm{V}} \in V_h(\Gamma)$ by

$$\varphi_{\mathrm{V}}(p) = \begin{cases} 1, & \text{if node } \ p = \mathrm{V}, \\ 0, & \text{if node } \ p \neq \mathrm{V}. \end{cases}$$

Now, we define the coarse subspace

$$V_h^0(\Gamma_r^{(k)}) = span\{\varphi_{\mathrm{F}},\ \varphi_{\mathrm{E}},\ \varphi_{\mathrm{V}} :\ \mathrm{F} \in \mathcal{F}_{\Gamma_r^{(k)}},\ \mathrm{E} \in \mathcal{E}_{\Gamma_r^{(k)}},\ \mathrm{V} \in \mathcal{V}_{\Gamma_r^{(k)}}\}$$

$$(k = 0, \cdots, J-1;\ r = 1, \cdots, M_k).$$

Remark 2. In most situations, there is no node on an edge E, and a vertex V is not a node. Then, the coarse subspace reduces to

$$V_h^0(\Gamma_r^{(k)}) = span\{\varphi_{\mathrm{F}} :\ \mathrm{F} \in \mathcal{F}_{\Gamma_r^{(k)}}\}.$$

In such case, we have that $dim(V_h^0(\Gamma_r^{(k)})) = m_{k+1}$.

With the local subspaces and the coarse subspaces defined above, we get the multilevel space decomposition of $V_h(\Gamma)$

$$V_h(\Gamma) = \sum_{k=0}^{J-1} \sum_{r=1}^{M_k} V_h^0(\Gamma_r^{(k)}) + \sum_{r=1}^{M_J} V_h(\Gamma_r^{(J)}).$$

Remark 3. In applications, the above multilevel decomposition would be generated in a suitable manner such that both each local subspace $V_h(\Gamma_r^{(J)})$ and each coarse subspace $V_h^0(\Gamma_r^{(k)})$ have a low dimension.

[1] Thanks to Prof. R. Hiptmair, who told the author that the basis φ_{F} can be also defined using an aggregation framework. Our method seems to be cheaper than the aggregation method (refer to Remark 1).

3.3 Main Result

Let $\langle\cdot,\ \cdot\rangle$ denote the inner product on Γ. For ease of notation, we define

$$\|\varphi_h\|^2_{*,\Gamma} = \langle S_h\varphi_h, \varphi_h\rangle \cong (a_1+a_2)|\varphi_h|^2_{H^{\frac{1}{2}}_{00}(\Gamma)} \qquad \varphi_h \in V_h(\Gamma).$$

The following result follows from [4].

Theorem 1. *For any $\phi_h \in V_h(\Gamma)$, there exist functions*

$$\phi^{(k)}_{r,\,0} \in W^0_h(\Gamma^{(k)}_r)\ \ (0 \le k \le J-1)\ \ \textit{and}\ \ \phi^{(J)}_r \in V_h(\Gamma^{(J)}_r)$$

such that

$$\phi_h = \sum_{k=0}^{J-1}\sum_{r=1}^{M_k} \phi^{(k)}_{r\ ,0} + \sum_{r=1}^{M_J} \phi^{(J)}_r \tag{4}$$

and

$$\sum_{k=0}^{J-1}\sum_{r=1}^{M_k} \|\phi^{(k)}_{r\ ,0}\|^2_{*,\Gamma} + \sum_{r=1}^{M_J} \|\phi^{(J)}_r\|^2_{*,\Gamma} \lesssim J[1+\log(1/h)]^2\|\phi_h\|^2_{*,\Gamma} \quad (J \ge 1). \tag{5}$$

4 Multilevel Preconditioner for S_h

In this section, we construct a multilevel preconditioner for S_h based on the multilevel decomposition introduced in the previous section.

4.1 Coarse Solvers

We want to consider a coarse solver $M^{(k)}_{r,\,0}:\ V^0_h(\Gamma^{(k)}_r) \to V^0_h(\Gamma^{(k)}_r)$ satisfying

$$\langle (M^{(k)}_{r,\,0})^{-1} S_h\phi_h, S_h\phi_h\rangle \cong \langle \phi_h, S_h\phi_h\rangle, \quad \forall \phi_h \in V^0_h(\Gamma^{(k)}_r).$$

The desired coarse solver can be defined by

$$\begin{aligned}(M^{(k)}_{r,\,0})^{-1}\phi_h = {} & \frac{1}{\lambda'_k}\sum_{\mathrm{F}\in\mathcal{F}_{\Gamma^{(k)}_r}} \langle\phi_h,\varphi_{\mathrm{F}}\rangle\varphi_{\mathrm{F}} + \sum_{\mathrm{E}\in\mathrm{e}_{\Gamma^{(k)}_r}} \frac{1}{\lambda^k_{\mathrm{E}}}\langle\phi_h,\varphi_{\mathrm{E}}\rangle\varphi_{\mathrm{E}} \\ & + \frac{1}{\lambda''_k}\sum_{\mathrm{V}\in\mathcal{V}_{\Gamma^{(k)}_r}} \langle\phi_h,\varphi_{\mathrm{V}}\rangle\varphi_{\mathrm{V}}, \quad \phi_h \in V^0_h(\Gamma^{(k)}_r).\end{aligned}$$

Here,

$$\begin{aligned}\lambda'_k &= (a_1+a_2)d_k\log(d_k/h) \cong \langle S_h\varphi_{\mathrm{F}},\varphi_{\mathrm{F}}\rangle,\\ \lambda^k_{\mathrm{E}} &= (a_1+a_2)\|\varphi_{\mathrm{E}}\|^2_{0,\,\mathrm{E}} \cong \langle S_h\varphi_{\mathrm{E}},\varphi_{\mathrm{E}}\rangle\end{aligned}$$

and

$$\lambda''_k = h(a_1+a_2) \cong \langle S_h\varphi_{\mathrm{V}},\varphi_{\mathrm{V}}\rangle.$$

4.2 Local Solvers

Inspired by the ideas in [3], we define the inverse of a local solver instead of the local solver itself.

Precisely, let us define the operator

$$K\varphi(q) = \frac{1}{4\pi}\int_{\Gamma}\frac{1}{|p-q|}\varphi(p)dp, \quad q \in \Gamma.$$

Since

$$\langle K\varphi, \varphi\rangle \cong \|\varphi\|^2_{-\frac{1}{2},\Gamma} \quad \forall\varphi \in H^{-\frac{1}{2}}(\Gamma),$$

we choose a local solver $M_r^{(J)} : V_h(\Gamma_r^{(J)}) \to V_h(\Gamma_r^{(J)})$ such that

$$(M_r^{(J)})^{-1} \cong (a_1 + a_2)^{-1} K|_{V_h(\Gamma_r^{(J)})}.$$

Thus, we can define $(M_r^{(J)})^{-1}$ by

$$\langle (M_r^{(J)})^{-1}\varphi_h, \psi_h\rangle = \frac{1}{4\pi(a_1+a_2)}\int_{\Gamma_r^{(J)}}\int_{\Gamma_r^{(J)}}\frac{\varphi_h(p)\psi_h(q)}{|p-q|}ds(p)ds(q),$$
$$\varphi_h \in V_h(\Gamma_r^{(J)}),\ \forall\psi_h \in V_h(\Gamma_r^{(J)}).$$

The above integrals can be calculated by the formulas introduced in [2]. Since each $\Gamma_r^{(J)}$ contains only a few nodes, it is cheap to calculate the stiffness matrix of $(M_r^{(J)})^{-1}$.

4.3 The Final Preconditioner

As usual, we define the L^2-projectors

$$Q_{r,\,0}^{(k)} : V_h(\Gamma) \to V_h^0(\Gamma_r^{(k)}), \quad Q_r^{(J)} : V_h(\Gamma) \to V_h(\Gamma_r^{(J)}).$$

Then, the desired preconditioner can be defined as follows

$$M_J^{-1} = \sum_{k=0}^{J-1}\sum_{r=1}^{M_k}(M_{r,\,0}^{(k)})^{-1}Q_{r,\,0}^{(k)} + \sum_{r=1}^{M_J}(M_r^{(J)})^{-1}Q_r^{(J)}. \tag{6}$$

The following result can be proved as in [1] (by using Theorem 1).

Theorem 2. *Assume that the sequence $\{m_k\}$ is uniformly bounded. Then, we have*

$$cond(M_J^{-1}S_h) \le CJ^2[1+\log(1/h)]^2. \tag{7}$$

Hereafter, C is a constant independent of h, of d_k and of the jumps of the coefficient $a(p)$ across the interface.

Remark 4. Our method can be extended to the case of multiple subdomains and interfaces with "crossedges". The two main changes in this extension are that we need to construct a suitable coarse subspace involving the "crossedges", and a multilevel decomposition for each interface (see [4] for the details). For this general case, the term $\log(1/h)$ in (7) would be replaced by $\log(H/h)$, H being the "size" of the subdomains.

Remark 5. We conjecture that the factor J in (7) (and (5)) can be dropped (see the numerical results in Section 6). Unfortunately, we fail to prove this conjecture.

5 Computational Complexity

Let $n_\Gamma = O((1/h)^2)$ be the number of the nodes on Γ, and let $N_\Gamma(J)$ denote the computational complexity for implementing the action of $M^{-1}(J)$.

Proposition 1. *Let $m \geq 2$ be a given positive integer. Set $J = [\log_m n_\Gamma]$, and choose m_k by*

$$m_1 = m_2 = \cdots = m_J = m. \tag{8}$$

Then,

$$N_\Gamma(J) = O(n_\Gamma), \tag{9}$$

which is optimal.

6 Numerical Experiments

Consider the elliptic problem (1) with $\Omega = [0,\ 2] \times [0,\ 1]^2$, and

$$a(x, y, z) = \begin{cases} 10^{-5}, & \text{if } (x, y, z) \in [0,\ 1]^3, \\ 1, & \text{otherwise.} \end{cases}$$

The source function f is chosen in a suitable manner.

Decompose Ω into two cubes with edge length equal to 1, and use the standard $\mathbb{P}_1$ elements on each cube. Finally, decompose each $\Gamma_r^{(k)}$ $(k \leq J-1)$ into four squares with the same size (i.e., $m_k = 4$). We solve the interface equation (3) by PCG iteration with preconditioner M_J^{-1}, considering a tolerance $tol = 10^{-5}$. Some numerical results are reported in table 1.

Table 1. Number of iterations

$1/h$	$J=1$	$J=2$	$J=3$	$J=4$
8	11	11	/	/
16	15	16	15	/
32	19	20	21	20

Table 1 shows that the number of iterations for the new methods depend slightly on the ratio $1/h$ and is independent of the level J.

7 Conclusions

We have introduced a new multilevel preconditioner for Steklov-Poincaré operators. Here, the traditional nested grids are unnecessary. The preconditioner not only features almost optimal convergence, but also optimal computational complexity.

The future works will focus on developing a substructuring method with inexact solvers (almost finished, see [4] for an initial version), and on studying the preconditioning similar operators.

Acknowledgement. Thanks to Dr. Dudu Sun and Dr. Yang Liu for performing the numerical experiments. The work is supported by The Key Project of Natural Science Foundation of China G10531080 and National Basic Research Program of China G2005CB321702.

References

[1] J. Bramble, J. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Math. Comput.*, 55:1–22, 1990.

[2] S. Erichsen and S. Sauter. Efficient automatic quadrature in 3-d Galerkin BEM. *Comput. Methods Appl. Mech. Engrg.*, 157:215–224, 1998.

[3] Q. Hu. Preconditioning Poincaré-Steklov operators arising from domain decompositions with mortar multipliers. *IMA J. Numer. Anal.*, 24:643–669, 2004.

[4] Q. Hu. Non-overlapping domain decomposition methods with a new class of multilevel solvers. Research report, ICMSEC (China), No. ICM-05-22, http://icmsec.cc.ac.cn, 2005.

[5] B. Smith and O. Widlund. A domain decomposition algorithm using hierarchical basis. *SIAM J. Sci. Stat. Comput.*, 11:1212–1220, 1990.

[6] C. Tong, T. Chan, and C. Kuo. A domain decomposition preconditioner based on a change to a multilevel nodal basis. *SIAM J. Sci. Stat. Comput.*, 12:1486–1495, 1991.

[7] A. Toselli and O. Widlund. *Domain Decomposition Methods – Algorithms and Theory.* Springer, Berlin, 2005.

[8] J. Xu and S. Zhang. Preconditioning the Steklov-Poincaré operator by using Green's function. *Math. Comp.*, 66:125–138, 1997.

[9] H. Yserentant. On the multi-level splitting of finite element spaces. *Numer. Math.*, 49:379–412, 1986.