RASHO: A Restricted Additive Schwarz Preconditioner with Harmonic Overlap

Xiao-Chuan Cai* Maksymilian Dryja† Marcus Sarkis‡

1 Introduction

A restricted additive Schwarz (RAS) preconditioning technique was introduced recently for solving general nonsymmetric sparse linear systems [1, 3, 4, 7, 8, 9, 11]. The RAS preconditioner improves the classical additive Schwarz preconditioner (AS), [10], in the sense that it reduces the number of iterations of the iterative method, such as GMRES, and also reduces the communication cost per iteration when implemented on distributed memory computers. However, RAS in its original form is a nonsymmetric preconditioner and therefore the cannot be used with the Conjugate Gradient method (CG). In this paper, we provide an extension of RAS for symmetric positive definite problems using the so-called harmonic overlaps (RASHO). Both RAS and RASHO outperform their counterparts of the classical additive Schwarz variants. Roughly speaking, the design of RASHO is based on a much deeper understanding of the behavior of Schwarz type methods in the overlapping regions, and in the construction of the overlap. Under RASHO, the overlap is obtained by extending the nonoverlapping subdomains only in the directions that do not cut the boundaries of other subdomains, and all functions are made harmonic in the overlapping regions. As a result, the subdomain problems in RASHO are smaller than those of AS, and the communication cost is also smaller when implemented on distributed memory computers, since the right-hand sides of discrete harmonic systems are always zero that do not need to be communicated. We will show numerically that RASHO preconditioned CG takes less number of iterations than the corresponding AS preconditioned CG. An almost optimal convergence theory will be

*Department of Computer Science, University of Colorado, Boulder, CO 80309, (*cai@cs.colorado.edu*). The work was supported in part by the NSF grants ASC-9457534, ECS-9725504, and ACI-0072089.

†Faculty of Mathematics, Informatics and Mechanics, Warsaw University, Warsaw, (*dryja@mimuw.edu.pl*). This work was supported in part by the NSF grant CCR-9732208 and in part by the Polish Science Foundation grant 2 P03A 021 16.

‡Mathematical Sciences Department, Worcester Polytechnic Institute, Worcester, MA 01609, (*msarkis@wpi.edu*). The work was supported in part by the NSF grant CCR-9984404.

presented for the RASHO for elliptic problems discretized with finite element methods.

Recall that the basic building blocks of classical Schwarz type algorithms are the operations of the form $(R_i^\delta)^T(A_i^\delta)^{-1}R_i^\delta$, where A_i^δ is the subdomain matrix and R_i^δ is the restriction operator for the extended subdomain (formal definitions will be given later in the paper). The multiplication of the such an operator with a vector, v, is realized by solving the linear system

$$A_i^\delta w = R_i^\delta v \tag{1}$$

on each extended subdomain. The key idea of RAS is that equation (1) is replaced by

$$A_i^\delta w = \begin{cases} v & \text{inside the un-extended subdomain} \\ 0 & \text{in the overlapping part of the subdomain.} \end{cases} \tag{2}$$

Note that the solution of (2) is discrete harmonic in the overlapping part of the subdomain, and therefore carries minimum energy in some sense. In this paper, we further explore the idea of "harmonic overlap" and at the same time keep the symmetry of the preconditioner.

The algorithm to be discussed below is applicable for symmetric positive definite problems. In order to provide a complete mathematical analysis, we restrict ourselves to the Poisson problem discretized with a finite element method. We consider a simple variational problem: Find $u \in H_0^1(\Omega)$, such that

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

where

$$a(u,v) = \int_\Omega \nabla u \cdot \nabla v\, dx \quad \text{and} \quad f(v) = \int_\Omega f v\, dx \quad \text{for} \quad f \in L^2(\Omega).$$

For simplicity, let Ω be a bounded polygonal region in $\Re^2$ with a diameter of size $O(1)$. The extension of the algorithm and results to $\Re^3$ can be carried out easily. Let $\mathcal{T}^h(\Omega)$ be a shape regular, quasi-uniform triangulation, of size $O(h)$, of Ω and $\mathcal{V}(\Omega) \subset H_0^1(\Omega)$ the finite element space consisting of continuous piecewise linear functions associated with the triangulation. We are interested in solving the following discrete problem associated with (3): Find $u^* \in \mathcal{V}$ such that

$$a(u^*, v) = f(v), \quad \forall\, v \in \mathcal{V}. \tag{4}$$

Using the standard basis functions, (4) can be rewritten as a linear system of equations

$$Au^* = f. \tag{5}$$

For simplicity, we understand u^* and f both as functions and vectors depending on the situation.

2 Notations

Let n be the total number of interior nodes of $\mathcal{T}^h(\Omega)$ and W the set of the nodes. We assume that a node-based partitioning has been applied and resulted in N nonoverlapping subsets $W_i^0, i = 1, \ldots, N$, whose union is W. For each W_i^0, we define a region Ω_i^R as the union of all the elements of $\mathcal{T}^h(\Omega)$ that have all three vertices on $W_i^0 \cup \partial\Omega$. We denote H as the representative size of the subregion Ω_i^R. We define the overlapping partition of W as follows. Let $\{W_i^1\}$ be the one-overlap partition of W, where $W_i^1 \supset W_i^0$ is obtained by including all the immediate neighboring vertices of the vertices in W_i^0. Using the idea recursively, we can define a δ-overlap partition $W = \bigcup_{i=1}^N W_i^\delta$. δh is approximately the extend of the extension.

We next define a subregion of Ω induced by a set of nodes of $\mathcal{T}^h(\Omega)$ as follows. Let Z be a subset of W. The induced subregion, denoted as $\Omega(Z)$, is defined as the union of: (1) the set Z itself; (2) the union all the open elements (triangles) of $\mathcal{T}^h(\Omega)$ that have at least one vertex in Z; and (3) the union of the open edges of these triangles that have at least one endpoint as a vertex of Z. Note that $\Omega(Z)$ is always an open region. The extended region Ω_i^δ is defined as $\Omega(W_i^\delta)$. We introduce the subspace

$$\mathcal{V}_i^\delta \equiv \mathcal{V} \cap H_0^1(\Omega_i^\delta) \text{ extended by zero to } \Omega \backslash \Omega_i^\delta.$$

It is easy to check that

$$\mathcal{V} = \mathcal{V}_1^\delta + \mathcal{V}_2^\delta + \cdots + \mathcal{V}_N^\delta.$$

This decomposition is used in defining the classical additive Schwarz algorithm without a coarse space. Let us define $P_i^\delta : \mathcal{V} \to \mathcal{V}_i^\delta$ by

$$a(P_i^\delta u, v) = a(u, v), \quad \forall u \in \mathcal{V}, \qquad \forall v \in \mathcal{V}_i^\delta. \tag{6}$$

Then, the classical one-level additive Schwarz operator has the form

$$P^\delta = P_1^\delta + \cdots + P_N^\delta.$$

Let $\Gamma_i^\delta = \partial\Omega_i^\delta \backslash \partial\Omega$; i.e., the part of the boundary of Ω_i^δ that does belong to the Dirichlet part of the boundary. We define the interface overlapping boundary Γ^δ as the union of all Γ_i^δ; i.e., $\Gamma^\delta = \cup_{i=1}^N \Gamma_i^\delta$. We then define the following subsets of W:

- $W^{\Gamma^\delta} \equiv W \cap \Gamma^\delta$ (interface nodes)
- $W_i^{\Gamma^\delta} \equiv W^{\Gamma^\delta} \cap W_i^\delta$ (local interface nodes)
- $W_{i,in}^{\Gamma^\delta} \equiv W^{\Gamma^\delta} \cap W_i^0$ (local internal interface nodes)
- $W_{i,cut}^{\Gamma^\delta} \equiv W_i^{\Gamma^\delta} \backslash W_{i,in}^{\Gamma^\delta}$ (local cut interface nodes)

- $W^{\delta}_{i,ovl} \equiv (W^{\delta}_{i} \backslash W^{\Gamma^{\delta}}_{i}) \cap (\bigcup_{j \neq i} W^{\delta}_{j})$ (local overlapping nodes)

- $W^{\delta}_{i,non} \equiv W^{\delta}_{i} \backslash (W^{\Gamma^{\delta}}_{i} \cup W^{\delta}_{i,ovl})$ (local nonoverlapping nodes)

- $W^{\delta}_{i,in} \equiv W^{\delta}_{i,non} \cup W^{\Gamma^{\delta}}_{i,in}$ (internal nodes)

We note that the notions of subdomains, harmonic overlaps, the classification of nodal points can all be defined in terms of the graph of the sparse matrix.

We frequently use functions that are discrete harmonic at certain nodes. Let $x_k \in W$ be a mesh point and $\phi_{x_k}(x) \in \mathcal{V}$ the finite element basis function associated with x_k; i.e., $\phi_{x_k}(x_k) = 1$, and $\phi_{x_k}(x_j) = 0, j \neq k$. We say $u \in \mathcal{V}$ is discrete harmonic at x_k if $a(u, \phi_{x_k}) = 0$. If u is discrete harmonic at a set of nodal points Z, we say u is discrete harmonic in $\Omega(Z)$.

Our new algorithm will be built on $\widetilde{\mathcal{V}}^{\delta}_{i}$ defined as a subspace of $\mathcal{V}^{\delta}_{i}$. $\widetilde{\mathcal{V}}^{\delta}_{i}$ consists of all functions that vanish on the cuting nodes $W^{\Gamma^{\delta}}_{i,cut}$ and discrete harmonic at the nodes $W^{\delta}_{i,ovl}$. Note that the support of the subspace $\widetilde{\mathcal{V}}^{\delta}_{i}$ is

$$\widetilde{W}^{\delta}_{i} \equiv W^{\delta}_{i} \backslash W^{\Gamma^{\delta}}_{i,cut}$$

and, since the values at the harmonic nodes are not independent, they can not be counted toward the degree of freedoms. The dimension of $\widetilde{\mathcal{V}}^{\delta}_{i}$ is $dim(\widetilde{\mathcal{V}}^{\delta}_{i}) = |W^{\delta}_{i,in}|$. Let $\widetilde{\Omega}^{\delta}_{i} \equiv \Omega(\widetilde{W}^{\delta}_{i})$ be the induced domain. It is easy to see that $\widetilde{\Omega}^{\delta}_{i}$ is the same as Ω^{δ}_{i} but with cuts. We have then $\widetilde{\mathcal{V}}^{\delta}_{i} = \mathcal{V} \cap H^1_0(\widetilde{\Omega}^{\delta}_{i})$ and discrete harmonic on $\Omega^{\delta}_{i,ovl} \equiv \Omega(W^{\delta}_{i,ovl})$. We define $\widetilde{\mathcal{V}}^{\delta} \subset \mathcal{V}^{\delta}$ as

$$\widetilde{\mathcal{V}}^{\delta} \equiv \widetilde{\mathcal{V}}^{\delta}_{1} + \cdots + \widetilde{\mathcal{V}}^{\delta}_{N},$$

which is a direct sum.

3 RAS with harmonic overlap

Let $\widetilde{P}^{\delta}_{i} : \widetilde{\mathcal{V}}^{\delta} \to \widetilde{\mathcal{V}}^{\delta}_{i}$ be a projection operator satisfying

$$a(\widetilde{P}^{\delta}_{i} u, v) = a(u, v), \;\; \forall u \in \widetilde{\mathcal{V}}^{\delta}, \quad \forall v \in \widetilde{\mathcal{V}}^{\delta}_{i}. \tag{7}$$

The RASHO operator can be defined as

$$\widetilde{P}^{\delta} = \widetilde{P}^{\delta}_{1} + \cdots + \widetilde{P}^{\delta}_{N}. \tag{8}$$

Note that the solution u^* of (5) is not, generally speaking, in the subspace $\widetilde{\mathcal{V}}^{\delta}$, therefore, the operator $\widetilde{P}^{\delta}$ cannot be used to solve the linear system (5) directly. We will need to modify the right-hand side of the system; see Lemma 3.1. We will also show that the elimination of the variables associated with the overlapping nodes is not needed in order to apply $\widetilde{P}^{\delta}$ to a vector $v \in \widetilde{\mathcal{V}}^{\delta}$.

We now introduce the matrix form of (8). We define the restriction operator, or a matrix, $\widetilde{R}_i^\delta$: $W \rightarrow \widetilde{W}_i^\delta$ as follows. Let $v = (v_1, \ldots, v_n)^T$ be a vector corresponding to the nodal values of a function $u \in \mathcal{V}$; namely for any node $x_i \in W$, $v_i = u(x_i)$. For convenience, we say "v is defined on W". Its restriction on $\widetilde{W}_i^\delta$, $\widetilde{R}_i^\delta v$, is defined as

$$\left(\widetilde{R}_i^\delta v\right)(x_i) = \begin{cases} v_i & \text{if } x_i \in \widetilde{W}_i^\delta \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Use this restriction operator, we define the subdomain stiffness matrix as

$$\widetilde{A}_i^\delta = \widetilde{R}_i^\delta \, A \, (\widetilde{R}_i^\delta)^T,$$

which can also be obtained by the discretization of the original problem on $\widetilde{W}_i^\delta$ with zero Dirichlet data on nodes $W \setminus \widetilde{W}_i^\delta$. If $\widetilde{A}_i^\delta$ is subspace-invertible, we have

$$\widetilde{P}_i^\delta = (\widetilde{R}_i^\delta)^T \left(\widetilde{A}_i^\delta\right)^{-1} \widetilde{R}_i^\delta \, A$$

and

$$\widetilde{P}^\delta = \left((\widetilde{R}_1^\delta)^T (\widetilde{A}_1^\delta)^{-1} \widetilde{R}_1^\delta + \cdots + (\widetilde{R}_N^\delta)^T (\widetilde{A}_N^\delta)^{-1} \widetilde{R}_N^\delta\right) \, A. \tag{10}$$

The next lemma shows how to modify the system (5) so that its solution belongs to $\widetilde{\mathcal{V}}^\delta$. A proof can be found in [2].

Lemma 3.1 *Let u^* and f be the exact solution and the right-hand side of (5), and*

$$w = \sum_{i=1}^{N} (\widetilde{R}_i^\delta)^T (\widetilde{A}_i^\delta)^{-1} \widetilde{R}_i^0 f, \tag{11}$$

then, we have $\widetilde{u}^* = u^* - w \in \widetilde{\mathcal{V}}^\delta$ is the solution of the modified linear system of equations

$$A\widetilde{u}^* = f - Aw = \widetilde{f}.$$

We remark that RASHO has several advantages over the classical AS. Let us recall AS briefly. Let

$$\left(R_i^\delta v\right)(x_i) = \begin{cases} v_i & \text{if } x_i \in W_i^\delta \\ 0 & \text{otherwise.} \end{cases} \tag{12}$$

Then the AS operator takes the following matrix form

$$P^\delta = \left((R_1^\delta)^T (A_1^\delta)^{-1} R_1^\delta + \cdots + (R_N^\delta)^T (A_N^\delta)^{-1} R_N^\delta\right) \, A, \tag{13}$$

where $A_i^\delta = R_i^\delta A (R_i^\delta)^T$. We remark that the size of the matrix A_i^δ is $|W_i^\delta|$, which is bigger than the size of the matrix $\widetilde{A}_i^\delta$, which is $|\widetilde{W}_i^\delta|$. In a distributed memory implementation, the operation $R_i^\delta v$ involves moving data from one processor to another, but the operation $\widetilde{R}_i^\delta v$ does not involve any communication. In RASHO, if $u \in \widetilde{\mathcal{V}}^\delta$, then it is easy to see that

$$\widetilde{R}_i^\delta A u = \widetilde{R}_{i,in}^\delta A u, \tag{14}$$

where $\widetilde{R}_{i,in}^\delta$ is defined as

$$\left(\widetilde{R}_{i,in}^\delta v\right)(x_i) = \begin{cases} v_i & \text{if } x_i \in W_{i,in}^\delta \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

Therefore, for functions in $\widetilde{\mathcal{V}}^\delta$, we can rewrite $\widetilde{P}^\delta$, as in (10), in the following form

$$\widetilde{P}^\delta = \left((\widetilde{R}_1^\delta)^T (\widetilde{A}_1^\delta)^{-1} \widetilde{R}_{1,in}^\delta + \cdots + (\widetilde{R}_N^\delta)^T (\widetilde{A}_N^\delta)^{-1} \widetilde{R}_{N,in}^\delta \right) A. \tag{16}$$

Although the operator (16) does not look like a symmetric operator, but it is indeed symmetric when applying to functions in the subspace $\widetilde{\mathcal{V}}^\delta$. The form (14) takes the advantage of the fact that the operator $\widetilde{R}_{i,in}^\delta$ is communication-free in the sense that it needs only the residual associated with nodes in $W_{i,in}^{\Gamma^\delta} \subset \Omega_i^0$. We note, however, that to compute the residual at nodes $W_{i,in}^{\Gamma^\delta}$ some communications are required. The processor associated with subdomain Ω_i needs to obtain the local solution from the neighboring subdomains at nodes connected to $W_{i,in}^{\Gamma^\delta}$. It is important to note that the amount of communications does not depend on the size of the overlap since only one layer of nodes is required. This shows that in terms of communications, the RASHO is superior to AS and RAS.

The following theorem provides an estimate of the condition number of the RASHO operator $\widetilde{P}^\delta$ in terms of the mesh sizes h and H, and the overlapping factor δ. It is interesting to see that, for the small overlap case, our condition number estimate is equivalent to the estimate for the AS preconditioner [5], while for generous overlap, our estimate is equivalent to the estimate for iterative substructuring algorithms [6].

Theorem 3.1 *[2] The RASHO operator $\widetilde{P}^\delta$ is symmetric in the inner product $a(\cdot,\cdot)$, nonsingular, and bounded in the following sense*

$$C_0^{-2} a(u,u) \le a(\widetilde{P}^\delta u, u) \le C_1 \ a(u,u) \quad \forall u \in \widetilde{\mathcal{V}}^\delta. \tag{17}$$

Here

$$C_0^2 = C\left(\left(1 + \log\left(\frac{H}{h}\right)\right) + \frac{1}{H^2}\left(1 + \log\left(\frac{(\delta+1)h}{h}\right) + \frac{H}{(2\delta+1)h}\right)\right).$$

The constants $C, C_1 > 0$ are independent of h, H, and δ.

Table 1: RASHO and AS preconditioned CG for solving the Poisson's equation on a 128×128 mesh decomposed into $2 \times 2 = 4$ subdomains with overlap = *ovlp*. The AS/CG results are shown in ( ). The "+1" is for the preprocessing step needed for RASHO.

ovlp	iter	cond	max	min
0	42 (42)	129.(129.)	1.98 (1.98)	0.0154 (0.0154)
1	24+1 (28)	48.4 (86.3)	1.94 (4.00)	0.0402 (0.0464)
2	20+1 (23)	33.3 (51.8)	1.91 (4.00)	0.0574 (0.0773)
3	18+1 (20)	27.2 (37.0)	1.89 (4.00)	0.0694 (0.1081)

4 Numerical experiments

We present some numerical results for solving the Poisson's equation on the unit square with zero Dirichlet boundary conditions. We compare the performance of RASHO/CG and AS/CG in terms of the number of iterations and the condition numbers. We pay particular attention to the dependence on the number of subdomains and the size of the overlap.

In order to use RASHO/CG, we need to modify the linear system by forcing its modified solution to belong to $\widetilde{\mathcal{V}}^{\delta}$. To do so, we use the formula (11). The stopping condition for CG is to reduce the energy norm of the initial residual by a factor of 10^{-6}. The exact solution of the equation is taken to be $u(x, y) = e^{5(x+y)} \sin(\pi x) \sin(\pi y)$. All subdomain problems are solved exactly. The iteration count (iter), the condition number (cond), the maximum (max) and minimum (min) eigenvalues of the preconditioned matrix are summerized in Table 1, and Table 2. It is clear that the newly introduced RASHO/CG is always better than the classical AS/CG in terms of the iteration counts and the condition numbers. Although we do not have any parallel results to report at this point, we are confident that RASHO/CG would be even better than AS/CG on a parallel computers with distributed memory since much less communication is required.

References

[1] S. Balay, W. Gropp, L. McInnes, and B. Smith, *The Portable Extensible Toolkit for Scientific Computing (PETSc)*, `www.mcs.anl.gov/petsc`, 2000.

[2] X.-C. Cai, M. Dryja, and M. Sarkis, *Restricted additive Schwarz preconditioners with harmonic overlap for symmetric positive definite linear systems*, 2001. (in preparation)

Table 2: RASHO and AS preconditioned CG for solving the Poisson's equation on a $32 * DOM \times 32 * DOM$ mesh decomposed into $DOM \times DOM$ subdomains with overlap = 1.

$DOM \times DOM$	iter	cond	max	min
2×2	19+1 (20)	26.8 (43.7)	1.89 (4.00)	0.0708 (0.0916)
4×4	39+1 (42)	86.9 (145.)	1.95 (4.00)	0.0225 (0.0276)
8×8	75+1 (78)	328. (550.)	1.97 (4.00)	0.0060 (0.0073)
16×16	147+1(156)	1295(2168.)	1.98 (4.00)	0.0015 (0.0018)

[3] X.-C. Cai, C. Farhat, and M. Sarkis, *A minimum overlap restricted additive Schwarz preconditioner and applications in 3D flow simulations*, Contemporary Mathematics, 218 (1998), pp. 479-485.

[4] 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.

[5] M. Dryja and O. Widlund, *Domain decomposition algorithms with small overlap*, SIAM J. Sci. Comput., 15 (1994), pp. 604–620.

[6] M. Dryja, B. Smith, and O. Widlund, *Schwarz analysis of iterative substructuring algorithms for elliptic problems in three dimensions*, SIAM J. Numer. Anal., 31(1994), pp. 1662–1694.

[7] A. Frommer and D. Szyld, *An algebraic convergence theory for restricted additive Schwarz methods using weighted max norms*, SIAM J. Numer. Anal., 2001. (to appear)

[8] W. Gropp, D. Kaushik, D. Keyes, and B. Smith, *Performance modeling and tuning of an unstructured mesh CFD application*, Proceedings of SC2000, IEEE Computer Society, 2000.

[9] M. Lesoinne, M. Sarkis, U. Hetmaniuk, and C. Farhat, *A linearized method for the frequency analysis of three-dimensional fluid/structure interaction problems in all flow regimes*, Comp. Meth. Appl. Mech. Eng., 2001. (to appear).

[10] B. Smith, P. Bjørstad, and W. Gropp, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, 1995.

[11] A. Quarteroni and A. Valli, *Domain Decomposition Methods for Partial Differential Equations*, Oxford Science Publications, 1999.