

Preconditioned Iterative Methods on Sparse Subspaces

Kazufumi Ito*

Jari Toivanen*

December 10, 2004

Abstract

When some rows of the system matrix and a preconditioner coincide preconditioned iterations can be reduced to a sparse subspace. Taking an advantage of this property can lead to considerable memory and computational savings. This is particularly useful with the GMRES method. We consider the iterative solution of discretized partial differential equation on this sparse subspace. With a domain decomposition method and a fictitious domain method the subspace corresponds a small neighborhood of an interface. As numerical examples we solve the Helmholtz equation using a fictitious domain method and an elliptic equation with jump in the diffusion coefficient using a separable preconditioner.

Key words: Subspace iteration, preconditioning, Krylov subspace method, domain decomposition method, fictitious domain method, interface problem

1 Introduction

In this paper we consider the generic idea of preconditioned iterations on a subspace for solving a linear equation $Au = f$. Particularly we are interested in subspaces which have special sparsity structure, that is, a large part of components of vectors on a subspace are zero. Such subspaces arise when a part of the rows of the matrix A and a preconditioner coincide. In such a case the iterative solution can be carried out on a sparse subspace [4].

A commonly used class of iterative methods is the Krylov subspace methods. With such an iterative method iterants belong to a Krylov subspace which is sparse. This is especially useful property when we solve problems with the GMRES method [8]. This can lead to vast reduction on computational effort and memory requirement if the dimension of the sparse subspace is small. In Section 2 we will consider the Krylov subspace methods.

Our examples are based on discretized partial differential equations, but due to the algebraic definition of subspace iterations they are applicable for any linear systems. Probably the most well-known methods which can use sparse subspaces are domain decomposition methods with exact subdomain solvers; see [7, 9], for example. The basic idea is divide the domain into subdomains and then construct a preconditioner using exact solvers in them. Then the iterations can be reduced on the boundaries of the subdomains. Block Jacobi and Gauss-Seidel preconditioners are the simplest examples of methods in this class. We will give a description of these methods in Section 3.

*Center for Research in Scientific Computation, Box 8205, North Carolina State University, Raleigh, NC 27695-8205. Email: kito@ncsu.edu, jatoivan@ncsu.edu

Another class of methods capable using sparse subspaces are called algebraic fictitious domain methods [7] which are also often called as domain imbedding methods [1], capacitance matrix methods [2]. The basic idea of these methods is embed the original domain and problem into a larger simple-shaped domain where a more efficient solution method like a fast direct solver [10] can be used. These methods have lead especially efficient iterative solution procedures for scattering problems modeled using the Helmholtz equation [3, 5]. We will give an example of a fictitious domain method in Section 4.

Many problems have interfaces due to material parameter changes. For such problems it is often possible to construct an efficient preconditioner which coincides with the system matrix except the rows corresponding to unknowns near-by the interface. We will consider this kind of an example in Section 5. Also for complicated boundary conditions it might be useful to construct a preconditioner based on a simpler boundary condition. This leads to a similar situation as with interfaces.

2 General Formulation

We consider the iterative solution of a system of linear equations $Au = f$ with a non singular right preconditioner B . The square matrix A can be singular in which case a solution exists only if f belongs to the range of A . The system to be solved reads

$$AB^{-1}v = f \tag{2.1}$$

and after we have the vector v the solution vector u of the original problem is given by

$$u = B^{-1}v. \tag{2.2}$$

Other possibilities would be to consider the left preconditioned system $B^{-1}Au = B^{-1}f$ or to change the Euclidean inner product in the iterative method to be the B -inner product. In all these cases we can take an advantage of the sparsity of arising subspaces in a manner similar to the one considered in the following.

The sparse subspace is

$$X = \text{range}(A - B) \cap \text{range}(A). \tag{2.3}$$

That is, the i th component x_i of x on X can be nonzero only if the i th row of the matrices $A - B$ and A are both nonzero. From the sparsity point of view it is advantageous to make the dimension of X as small as possible. This should not be done in a way which increases the computational cost of the multiplication of a vector by B^{-1} or which deteriorates the conditioning of AB^{-1} , since this increases the number of iterations.

In order to perform iterations on the sparse subspace X we define $\hat{v} = v - f$. Then we have

$$AB^{-1}\hat{v} = f - AB^{-1}f = -(A - B)B^{-1}f = \hat{f} \in X, \tag{2.4}$$

where we have used the identity $AB^{-1} = (A - B)B^{-1} + I$. We define the equation

$$[(A - B)B^{-1} + I] \hat{v} = \hat{f}. \tag{2.5}$$

We solve (2.5) on the Krylov subspace

$$\text{span}\{r, AB^{-1}r, \dots, (AB^{-1})^{m-1}r\}, \tag{2.6}$$

which is a subspace of X provided that $r \in X$. Then any iterative method based on this Krylov subspace for the solution of (2.5) generates a sequence of approximate solutions \hat{v}^k on X if the initial guess v^0 belongs to X . Hence all required operations are carried out on the sparse subspace X .

The basic operations of the type

$$(A - B)B^{-1}x, \quad x \in X \tag{2.7}$$

performed during the iteration require the solution $B^{-1}x$ on the range of $(A - B)^T$. The dimension of this range is usually the same order as the dimension of X . For a Poisson or Helmholtz equation in a rectangular domain with a suitable discretization a special partial solution technique for $B^{-1}x$ can be used; see [3, 4, 5] and references therein. This can reduce the computational cost of these solutions considerably.

The dimension m of the subspace X depends on an application. In the following we consider linear systems arising from the finite element or finite difference discretization of partial differential equations. Furthermore, the subspaces are going to correspond an interface. Let n denote the dimension of A then m is $\mathcal{O}(n^{1/2})$ for problems in two-dimensional domains and m is $\mathcal{O}(n^{2/3})$ for problems in three-dimensional domains. For example, for a large three-dimensional problem n could be millions while m would be only order of tens of thousands. Hence, by taking advantage of the sparse subspace X we can reduced the memory usage by the factor of 100.

3 Domain Decomposition Methods

Let us consider the numerical solution of a partial differential equation, for example, the Poisson equation, in a domain Ω . A finite element or finite difference discretization leads to a linear system $Au = f$, where A has only a few nonzero entries per row. For simplicity we consider domain decomposition methods only with two subdomains, and linear finite elements or compact finite difference stencils for the discretization. The methods can be easy generalized for several subdomains and more general discretizations. In order to construct algebraically domain decomposition preconditioners we express A in the block form

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}. \tag{3.1}$$

Then block Jacobi and block Gauss-Seidel preconditioners are

$$B = \begin{pmatrix} A_{11} & 0 \\ 0 & A_{22} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{pmatrix}, \tag{3.2}$$

respectively. These preconditioners correspond Schwarz domain decomposition methods with an overlap of one mesh/grid step [9]. For the block Jacobi and block Gauss-Seidel preconditioners the sparse subspaces X are the ranges of the matrices

$$A - B = \begin{pmatrix} 0 & A_{12} \\ A_{21} & 0 \end{pmatrix} \quad \text{and} \quad A - B = \begin{pmatrix} 0 & A_{12} \\ 0 & 0 \end{pmatrix}, \tag{3.3}$$

respectively. The subspace for the block Jacobi corresponds to the unknowns on the boundaries of the overlap while the subspace for the block Gauss-Seidel corresponds to the unknowns on the boundaries of the overlap which are inside the first subdomain (corresponding to the first block).

For the block Jacobi preconditioner we have

$$AB^{-1} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} A_{11}^{-1} & 0 \\ 0 & A_{22}^{-1} \end{pmatrix} = \begin{pmatrix} I & A_{12}A_{22}^{-1} \\ A_{21}A_{11}^{-1} & I \end{pmatrix}. \quad (3.4)$$

Therefore, the eigenvalues of AB^{-1} depend on the strength of coupling blocks $A_{12}A_{22}^{-1}$ and $A_{21}A_{11}^{-1}$. For a strictly (block) diagonally dominant matrix A the eigenvalues belong to a disc centered at one with radius less than one. For the block Gauss-Seidel we have

$$AB^{-1} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} A_{11}^{-1} & 0 \\ -A_{22}^{-1}A_{21}A_{11}^{-1} & A_{22}^{-1} \end{pmatrix} = \begin{pmatrix} I - A_{12}A_{22}^{-1}A_{21}A_{11}^{-1} & A_{12}A_{22}^{-1} \\ 0 & I \end{pmatrix}. \quad (3.5)$$

The eigenvalues of AB^{-1} are the eigenvalues of $I - A_{12}A_{22}^{-1}A_{21}A_{11}^{-1}$ and ones. By using more detailed block structures for A and B which have also blocks corresponding to X shows that only the eigenvalues corresponding to X can be different from one.

A Neumann-Dirichlet domain decomposition preconditioner is based on a nonoverlapping decomposition of the domain Ω . On the first domain we impose a Neumann boundary condition on the interface between the subdomains while in the second subdomain we impose a Dirichlet boundary condition on the interface. This preconditioner can be obtained by modifying the block Gauss-Seidel preconditioner. It has the form

$$B = \begin{pmatrix} A_{11} + D & 0 \\ A_{21} & A_{22} \end{pmatrix}. \quad (3.6)$$

For the Poisson equation discretized with linear finite elements D is such a diagonal matrix that the rows sums of $A_{11} + D$ corresponding to $\text{range}(A_{12})$ are zero. Effectively this means that the first subdomain is reduced by the overlap and that a Neumann boundary condition is posed in the first subdomain on the interface between the subdomains.

For the Neumann-Dirichlet domain decomposition the subspace X is the range of the matrix

$$A - B = \begin{pmatrix} D & A_{12} \\ 0 & 0 \end{pmatrix}, \quad (3.7)$$

which coincides with the subspace for the block Gauss-Seidel preconditioner. We have

$$AB^{-1} = \begin{pmatrix} (A_{11} - A_{12}A_{22}^{-1}A_{21})(A_{11} + D)^{-1} & A_{12}A_{22}^{-1} \\ 0 & I \end{pmatrix}. \quad (3.8)$$

The choice $D = -A_{12}A_{22}^{-1}A_{21}$ leads to perfect conditioning. In this case, we would need to solve linear systems with the Schur complement matrix $A_{11} - A_{12}A_{22}^{-1}A_{21}$ which is usually computationally too expensive. Choosing D instead as in the Neumann-Dirichlet domain decomposition still leads to optimal conditioning for the Poisson equation; see [7, 9] and references therein. Here optimal means that the eigenvalues of AB^{-1} belong to a disc centered at one with radius less than one and independent of the mesh/grid step size.

4 Algebraic Fictitious Domain Methods

Here we are solving a partial differential equation in a domain Ω which can be embedded to a larger, simple shaped domain Π . The aim is to solve more efficiently an extended problem in Π . We use the notation $A_{11}u_1 = f_1$ for the system of linear equations obtained using a finite element or finite difference method in Ω . Furthermore, we denote the matrix

obtained by discretizing the extended problem in Π by B . Often there is an efficient way, for example, a fast direct solver [10], to solve problems with B . Then it is natural idea to use B as a preconditioner. The dimension of B is larger than the dimension of A_{11} . We introduce a compatible block presentation of B as

$$B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}, \quad (4.1)$$

where the matrix block B_{11} corresponds to A_{11} . Now we extend A_{11} to A in (2.1). The most trivial way is the zero extension, that is,

$$A = \begin{pmatrix} A_{11} & 0 \\ 0 & 0 \end{pmatrix}. \quad (4.2)$$

The matrix A is singular but this does not cause any difficulties with our sparse subspace method. In general the trivial extension by zeros does not lead to good conditioning for AB^{-1} . Thus, we use the following two forms of extensions which are

$$A = \begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix} \quad \text{and} \quad A = \begin{pmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{pmatrix}. \quad (4.3)$$

They are called as the upper extension and the lower extension, respectively. One possible upper extension is given by $A_{12} = B_{12}$ and $A_{22} = B_{22}$, and similarly one lower extension is given by $A_{21} = B_{21}$ and $A_{22} = B_{22}$. For some problems these extensions are optimal in the same sense as in Section 3. Then, for the upper and lower extensions the subspace X is given by the ranges of the matrices

$$\begin{pmatrix} A_{11} - B_{11} & 0 \\ -B_{21} & 0 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} A_{11} - B_{11} & -B_{12} \\ 0 & 0 \end{pmatrix}, \quad (4.4)$$

respectively. When A_{11} and B_{11} coincide then these subspaces are about the same size.

Now we discuss the conditioning of the matrix AB^{-1} . Usually it is easier to study the inverse of AB^{-1} . For example, let A be the lower extension of A_{11} as in (4.3) with $A_{21} = B_{21}$ and $A_{22} = B_{22}$. Then we have

$$(AB^{-1})^{-1} = BA^{-1} = \begin{pmatrix} (B_{11} - B_{12}B_{22}^{-1}B_{21})A_{11}^{-1} & B_{12}B_{22}^{-1} \\ 0 & I \end{pmatrix}. \quad (4.5)$$

Furthermore, if we assume $A_{11} = B_{11}$ then this matrix is the same as the matrix for the block Gauss-Seidel domain decomposition (3.5) with the matrices A and B interchanged. Thus, the analysis of fictitious domain methods and domain decomposition methods are very similar.

Let us consider a modification of the above upper extension which is motivated by the Neumann-Dirichlet domain decomposition. Here the original problem corresponding to A_{11} is a Dirichlet boundary value problem. We consider the extension with $A_{22} = B_{22} + D$, that is,

$$A = \begin{pmatrix} A_{11} & B_{12} \\ 0 & B_{22} + D \end{pmatrix}. \quad (4.6)$$

Then we have

$$(AB^{-1})^{-1} = \begin{pmatrix} B_{11}A_{11}^{-1} & (I - B_{11}A_{11}^{-1})B_{12}(B_{22} + D)^{-1} \\ B_{21}A_{11}^{-1} & (B_{22} - B_{21}A_{11}^{-1}B_{12})(B_{22} + D)^{-1} \end{pmatrix}. \quad (4.7)$$

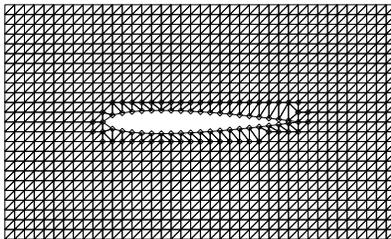


Figure 1: A 41×25 mesh for the NACA0012 airfoil and the nodes associated to X .

n	m	k
81×49	171	31
161×97	339	37
321×193	671	47
641×385	1334	56

Table 1: The dimensions and the number of GMRES iterations for four different meshes.

If $A_{11} = B_{11}$ then the choice $D = -B_{21}B_{11}^{-1}B_{12}$ would lead to perfect conditioning, but it is not computationally feasible. As before choosing D so that $B_{22} + D$ corresponds a Neumann boundary value problem leads to good conditioning and small computational cost. The following example uses a generalization of this extension.

We consider the time-harmonic electromagnetic scattering by the NACA0012 airfoil in TM-mode. The scattered electric field u is the solution of the Helmholtz equation

$$\Delta u + k^2 u = 0 \quad \text{in } \Omega, \quad u = g \quad \text{on } \partial\Omega \setminus \partial\Pi, \quad \mathcal{B}u = 0 \quad \text{on } \partial\Pi, \quad (4.8)$$

where k is the wave number, Π is a rectangular domain enclosing the airfoil $\Pi \setminus \Omega$, g is the incident field, and \mathcal{B} is a second-order absorbing boundary condition operator [3].

The length of the airfoil is one and its leading edge is at the origin. The rectangular domain Π is $[-0.5, 1.5] \times [-0.6, 0.6]$ and the wave number k is 8π . Thus, the airfoil is four wavelength long. The incident angle of a plane wave is 45° . The linear finite element discretization with mass lumping is performed on a rectangular mesh which is locally fitted to the boundary of Ω . We use an upper extension called the absorbing extension for A_{11} ; see [3, 5], for example. The preconditioner B is the same operator discretized on the fully rectangular mesh. Linear systems with B can be efficiently solved using the PSCR-method; see [3] and references therein. A coarse mesh and the sparse subspace X are shown in Figure 1. We have used four meshes having 10, 20, 40, and 80 nodes per wavelength. Table 1 gives the dimension n of A , the dimension m of X , and the number k of GMRES iterations to reduce the norm of the residual by the factor 10^{-6} .

5 Interface Problems

We have a domain Π which is divided into nonoverlapping subdomains. On each subdomain a partial differential equation is given and there are continuity conditions for the solution on the interfaces between the subdomains. In this case it might be easier to construct a preconditioner B which satisfies the partial differential equations in the interior of the subdomains, but which does not satisfy the given continuity conditions. A discretization

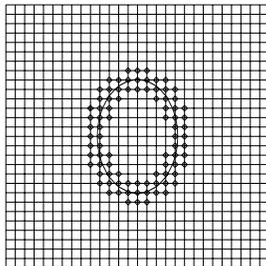


Figure 2: A coarse grid for the interface problem and the grid points associated to X .

n	m	k
49×49	144	13
99×99	270	18
199×199	542	21

Table 2: The dimensions and the number of GMRES iterations for three different grids.

leads to a system of linear equations in the block form

$$Au = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{21} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = f, \quad (5.1)$$

where the first block row corresponds to the union of the interiors of the subdomains and the second block row corresponds to a small neighborhood of the interface. Then the preconditioner has the compatible block form

$$B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & B_{22} \end{pmatrix}. \quad (5.2)$$

The dimension of X is the same as the dimension of the matrix block A_{22} .

As an example we consider an elliptic equation with a jump in the diffusion coefficient. Let Π be the unit square and Ω is a subdomain of Π . We consider the solution of the equation

$$-\nabla \cdot \beta \nabla u = f \quad \text{in } \Pi \setminus \partial\Omega, \quad [u] = \left[\beta \frac{\partial u}{\partial n} \right] = 0 \quad \text{on } \partial\Omega, \quad u = 0 \quad \text{on } \partial\Pi, \quad (5.3)$$

where $[\cdot]$ denotes the jump. The function β is 10 in Ω and 1 in $\Pi \setminus \Omega$. Scaling the equation by $1/\beta$, we obtain the Poisson equation with the interface condition for fluxes on $\partial\Omega$. We discretize the scaled equation with a second-order accurate finite difference scheme [6]. The preconditioner B is the Laplace equation discretized using the standard five point finite difference stencil. Figure 2 shows a coarse grid and the grid points associated to the subspace X . Table 2 gives the dimension n of the problem, the dimension m of the subspace X , and the number k of GMRES iterations to reduce the norm of the residual by the factor 10^{-6} for three different grids.

6 Conclusions

We have consider a simple idea to reduce preconditioner iterations on a sparse subspace. We showed that this is particularly useful when solving discretized partial differential equations

with a domain decomposition method or a fictitious domain method. Two numerical examples in two-dimensional domains demonstrated this benefit. For large scale three-dimensional computations taking an advantage of this sparse subspace can be extremely useful. For example in [3], a discretized Helmholtz equation with 1.3 billion unknowns was solved with the GMRES method by reducing the iterations onto a sparse subspace with the dimension less than 10 million. Here we have considered only a few common examples where such subspaces arise. There is a vast body of other problems where the same idea can be employed.

Acknowledgements

We are thankful for Dr. Yuri Kuznetsov and Dr. Tuomo Rossi for many fruitful discussions on iterative methods on subspaces.

References

- [1] C. BÖRGERS AND O. B. WIDLUND, *On finite element domain imbedding methods*, SIAM J. Numer. Anal., 27 (1990), pp. 963–978.
- [2] B. L. BUZBEE, F. W. DORR, J. A. GEORGE, AND G. H. GOLUB, *The direct solution of the discrete Poisson equation on irregular regions*, SIAM J. Numer. Anal., 8 (1971), pp. 722–736.
- [3] E. HEIKKOLA, T. ROSSI, AND J. TOIVANEN, *A parallel fictitious domain method for the three-dimensional Helmholtz equation*, SIAM J. Sci. Comput., 24 (2003), pp. 1567–1588.
- [4] Y. A. KUZNETSOV, *Matrix iterative methods in subspaces*, in Proceedings of the International Congress of Mathematicians, Vol. 1, 2 (Warsaw, 1983), Warsaw, 1984, PWN, pp. 1509–1521.
- [5] Y. A. KUZNETSOV AND K. N. LIPNIKOV, *3D Helmholtz wave equation by fictitious domain method*, Russian J. Numer. Anal. Math. Modelling, 13 (1998), pp. 371–387.
- [6] Z. LI AND K. ITO, *Maximum principle preserving schemes for interface problems with discontinuous coefficients*, SIAM J. Sci. Comput., 23 (2001), pp. 339–361.
- [7] G. I. MARCHUK, Y. A. KUZNETSOV, AND A. M. MATSOKIN, *Fictitious domain and domain decomposition methods*, Sov. J. Numer. Anal. Math. Modelling, 1 (1986), pp. 3–35.
- [8] Y. SAAD AND M. H. SCHULTZ, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 857–869.
- [9] B. F. SMITH, P. E. BJØRSTAD, AND W. D. GROPP, *Domain decomposition*, Cambridge University Press, Cambridge, 1996.
- [10] P. N. SWARZTRAUBER, *The methods of cyclic reduction and Fourier analysis and the FACR algorithm for the discrete solution of Poisson’s equation on a rectangle*, SIAM Review, 19 (1977), pp. 490–501.