INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2000; **00**:1–6 *Prepared using* nmeauth.cls *[Version: 2002/09/18 v2.02]*

The Generalized Global Basis (GGB) Method

H. Waisman[a], J. Fish[a]*, R. S. Tuminaro[b], and J. Shadid[c].

[a] *Department of Civil, Mechanical and Aerospace Engineering, Rensselaer Polythecnic Institute, Troy, NY 12180-3590*
[b] *Sandia National Laboratories, PO Box 969 MS 9217, Livermore, CA 94551*
[c] *Sandia National Laboratories, PO Box 5800 MS 1111, Albuquerque, NM 87185*

SUMMARY

In this work we present the Generalized Global-Basis (GGB) method aimed at enhancing performance of multilevel solvers for difficult systems such as those arising from indefinite and nonsymmetric matrices. The GGB method is based on the Global-Basis (GB) method [1], [2], which constructs an auxiliary coarse model from the largest eigenvalues of the iteration matrix. The GGB method projects these modes which would cause slow convergence to a coarse problem which is then used to eliminate these modes. Numerical examples show that best performance is obtained when GGB is accelerated by GMRES and used for problems with multiple right hand sides. In addition it is demonstrated that GGB method can enhance restarted GMRES strategies by retention of subspace information.

KEY WORDS: *Multilevel, Preconditioner, Indefinite, Nonsymmetric matrices, GMRES*

1. INTRODUCTION

Multilevel methods are very efficient solvers for symmetric positive definite systems arising from partial differential equations. For such systems, multilevel preconditioners require computational work proportional to the number of unknowns. However, when the system is nonsymmetric or highly indefinite, multilevel methods may not perform as well [3]. Such systems arise in a variety of applications including linearized Navier-Stokes equations, saddle-point and least squares problems with constraints and problems with an indefinite constitutive tensor arising as a result of damage/localization in solids. Some multilevel methods have been applied for certain weakly indefinite systems. However, the proposed strategies impose restrictions on the coarse grid, requiring that these grids are sufficiently fine for the algorithm to be convergent [4], [5]. For highly indefinite systems, convergence is not guaranteed unless the

*Correspondence to: Jacob Fish, Department of Civil, Mechanical and Aerospace Engineering, Rensselaer Polythecnic Institute, Troy, NY 12180-3590. email: fishj@rpi.edu

Contract/grant sponsor: Sandia National Laboratories, ASCI program; contract/grant number: DE-ACD4-94AL85000

Received 2003
Revised 2003

multilevel procedures are utilized in the context of the normal equations [6], [7]. Often, such an approach leads to systems which are much worse conditioned than the original system. A better approach is to use multilevel methods as preconditioners to Krylov iterative solvers [8], such as GMRES or QMR. However for some problems GMRES might stall in the first iterations until it resolves the indefinite modes of the system. This can often lead to poor convergence rates [9], [10], [11]. QMR, on the other hand, does not guarantee convergence for highly indefinite systems.

Recently, a Global-Basis two level method (GB) for highly indefinite systems has been developed [1], [2]. The method identifies the eigenvalues of the smoothing iteration matrix that are outside the region of convergence and constructs a coarse model using the corresponding eigenvectors. The method was shown to be robust for highly indefinite cases. Another method that has some similarities to the Global-Basis idea is the Recursive Projection Method (RPM) [12]. This method attempts to stablize unstable fixed point iterations by computing a projection onto the unstable subspace. In this paper, we focus on stabilizing the entire multilevel procedure by generalizing the Global-Basis method. The Generalized Global-Basis (GGB) method constructs an additional coarse grid correction spanned by the unresolved eigenmodes of the multilevel method. The idea is to filter out modes that are slow to converge and resolve them on an additional coarse grid. This accelerates the iterative process and yields rates of convergence similar to the application of the unaccelerated multilevel method applied to a positive definite system. We emphasize that any existing multilevel method can be applied as a preconditioner with the GGB acceleration. This allows any multilevel to be applied to indefinite systems. In terms of computational work, the method is the most efficient for problems that can reuse the information of the eigenspace. Examples include multiple right hand side problems, linear transient problems, variants of eigenvalues computations, nonlinear schemes where a modified Newton method can be applied. Another method for nonsymmetric matrices has been proposed in [13]. The method is based on updating preconditioners by a two level spectral shifting of the smallest eigenvalues of the preconditioned linear system. The GB and the GGB methods have some connections to these ideas. However, an important distinction is that the GB and GGB methods avoid the significant expense of computing small eigenvalues by instead focusing on the largest eigenvalues of the iteration operator.

An outline of the paper is as follows. In Section 2, we briefly summarize multilevel V cycle principles. We give an overview of the original Global-Basis method and motivate the GGB scheme. In Sections 3 and 4, analysis of the GB and GGB methods, respectively, are conducted on a model Helmholtz problem. In section 5, we study performance for three classes of problems: (i) 1D Helmholtz operator using a standard multilevel method to precondition GGB. (ii) 2D elastic wave propagation in a cracked membrane using softening zones using a Generalized Aggregation Multigrid (GAM) solver [18], and (iii) 3D thermal-convection flow using a piecewise constant multigrid solver [19]. We also discuss scalability and computational cost. In all cases we demonstrate the convergence characteristics of GGB and its beneficial effect on GMRES.

2. MOTIVATION AND GOALS

Consider an $N \times N$ linear algebraic system of equations

$$Ku = f \tag{1}$$

with the solution $u \in \mathbb{R}^N$, and the right hand side vector $f \in \mathbb{R}^N$. The system matrix $K \in \mathbb{R}^{N \times N}$ is considered to be generally nonsymmetric indefinite. A generic two-level multigrid V-cycle is illustrated in Figure 1. We introduce the following notation: the prolongation operator from the coarse grid to the fine grid $Q : \mathbb{R}^m \rightarrow \mathbb{R}^N$, where the coarse model is $m \times m$ and $m < N$. The restriction operator is taken to be transpose of the prolongation operator i.e. $Q^T : \mathbb{R}^N \rightarrow \mathbb{R}^m$. The coarse grid operator is defined by $K_0 = Q^T K Q$, and the smoothing or relaxation procedure is given by $M \in \mathbb{R}^{N \times N}$. The subscript zero is used to denote variables on the coarse space.

Step 1: Pre-smoothing

$$u^{i+1} \leftarrow u^i + M^{-1}(f - Ku^i) \qquad \text{do } \nu_1$$

Step 2: Coarse grid correction

a. Residual restriction:

$$r_0^i = Q^T(f - Ku^i)$$

b. Coarse grid solution:

$$u_0^i = K_0^{-1} r_0^i \quad \text{where} \quad K_0 \leftarrow Q^T K Q$$

b. Correction:

$$u^{i+1} \leftarrow u^i + Q u_0^i$$

Step 3: Post-smoothing

$$u^{i+1} \leftarrow u^i + M^{-1}(f - Ku^i) \qquad \text{do } \nu_2$$

Figure 1. Generic two-level multigrid V-cycle

If the error after iteration i is $e^i = u - u^i$, than reduction of the error after one V-cycle is controlled by the multilevel iteration matrix, given as

$$e^{i+1} = S^{\nu_2} T S^{\nu_1} e^i = R e^i, \tag{2}$$

where ν_1 and ν_2 are the number of pre and post smoothing. $S \in \mathbb{R}^{N \times N}$ is the smoothing iteration matrix written as

$$S = I - M^{-1} K, \tag{3}$$

and $T \in \mathbb{R}^{N \times N}$ is the coarse grid correction given by

$$T = I - Q(Q^T K Q)^{-1} Q^T K \tag{4}$$

where I is the $N \times N$ identity matrix. T is a projector satisfying $T = T^2$ with a spectral radius of $\rho(T) = 1$. The two-level cycle is guaranteed to converge iff the spectral radius of the two-level iteration matrix is inside the unit circle, $\rho(R) < 1$. The asymptotic rate of convergence is defined as

$$c = \lim_{i \rightarrow \infty} \frac{\left\| e^{i+1} \right\|}{\left\| e^i \right\|} = \rho(R). \tag{5}$$

Multilevel methods consist of two major elements: smoothing and coarse grid correction. When symmetric positive definite (SPD) systems are considered, classical iterative methods, used as

smoothers, eliminate the oscillatory components of the error leaving the smooth components almost untouched. This motivates the use of a coarse grid correction, where smooth components of the error are effectively approximated on a coarser grid. However, for difficult systems such as indefinite and nonsymmetric systems, smoothing may leave some oscillatory modes untouched, and thus standard multilevel methods might magnify these modes rather than reducing them [3].

For certain indefinite problems, multilevel methods used as preconditioners to Krylov type methods might perform well provided that the spectrum of the overall iteration matrix is clustered around zero [11]. For highly indefinite problems some multilevel preconditioners can be accelerated leading to rapid convergence, while some can cause GMRES to stall until slow converging modes are captured [9], [11]. One way to improve the convergence rate is to use GMRES as a smoother for the multilevel on the coarse grid and as an accelerator on the fine grid [14]. The method was implemented on a Helmholtz type problem. However, it is not clear whether this provides a truly efficient solution technique.

The global-basis (GB) method has been shown to be robust [1], [2]. This method accelerates a standard iterative method by identifing all the modes that are outside the region of convergence by solving for the largest eigenvalues $\lambda_i > |1|$ of the smoothing iteration matrix for smoothing

$$S\phi_i = \lambda_i \phi_i \qquad i = 1, ..., N. \tag{6}$$

A prolongation operator is then assembled using these global-basis modes as columns and used in the context of the two-level method. The authors in [1], [2] demostrated robustness of the method for symmetric indefinite systems. They employed a Lanczos method with look ahead and re-orthogonalization for the eigenvectors computations. The main shortcomings is that the convergence rate is governed by the properties of the smoother and for some highly indefinite systems there might be a significant percentage of error modes outside the unit circle that causes the entire computational cost to be dominated by the cost of solving the eigen-problem.

The basic idea of the generalized method, termed as Generalized Global Basis (GGB) method is this. Instead of accelerating a simple smoothing iteration, we filter out the modes which are unresolved by a multilevel method. Figure 2 schematically illustrates the architecture of the method. Black circles denote local smoothing at each level, and GMRES/QMR is an outer accelerator. We use the multilevel method as a preconditioner, filtering its slowly converging modes onto a coarse grid, where they are exactly resolved. Mode identification is done by solving the following eigenvalue problem for the multilevel iteration matrix

$$R\phi_i = \lambda_i \phi_i \qquad i = 1, \ldots, N. \tag{7}$$

Since we are interested in modes corresponding to large magnitude eigenvalues $\lambda_i > |1|$, implicitly restarted Arnoldi method [15] from ARPACK [16], is employed.

The iteration matrix of a single GGB cycle, shown in Figure 2, without an external accelerator can be written as

$$e^{i+1} = (STS)^{\nu_2} F_{GGB} (STS)^{\nu_1} e^i = R^{\nu_2} F_{GGB} R^{\nu_1} e^i, \tag{8}$$

where STS is the multilevel iteration matrix and ν_1, ν_2 correspond to the number of V-cycles. F_{GGB} is the multilevel method correction operator, or filter, given as

$$F_{GGB} = I - Q_f (Q_f^* K Q_f)^{-1} Q_f^* K, \tag{9}$$

where the prolongation operator Q_f is spanned by the highest modes of R. For indefinite nonsymmetric or complex problems the multilevel iteration matrix R will most likely be nonsymmetric causing the eigenvectors to be complex. For this reason we use the adjoint notation. In the next sections, we study convergence characteristics of GB and GGB methods

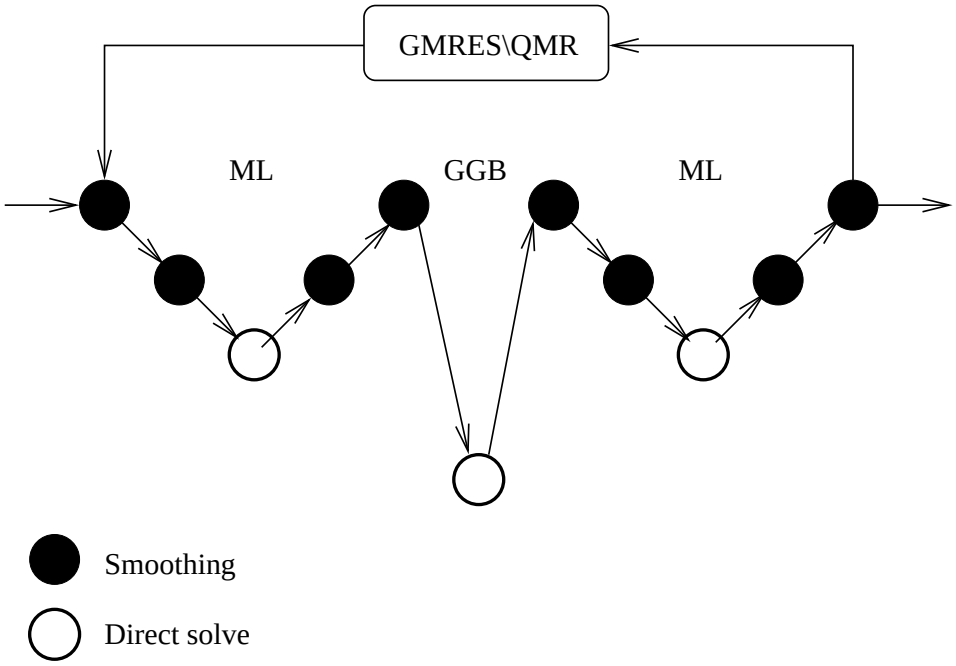

Figure 2. Generalized Global Basis (GGB) correction cycle

on Helmholtz problem. We show that the asymptotic convergence rate of GGB depends on the multilevel preconditioner, unlike the original Global-Basis method whose convergence characteristics are governed by a single level preconditioner employed for smoothing.

3. ANALYSIS OF GLOBAL BASIS TWO LEVEL METHOD

Consider a one-dimensional, Helmholtz model problem on the interval $\Omega = [0, 1]$ with homogeneous Dirichlet boundary conditions

$$\begin{cases} -u'' - k^2 u = f \\ u(x = 0) = u(x = 1) = 0 \end{cases} \tag{10}$$

. The domain is divided into $N + 1$ segments with equal length $h = \frac{1}{N+1}$. A central difference discretization is employed

$$-u''(x) = \frac{u(x + h) - 2u(x) + u(x - h)}{h^2} + O(h^2) \tag{11}$$

leading to the $N \times N$ linear system (1) with the following stencil

$$\frac{1}{h^2}[1 \quad -2 \quad 1] - k^2 I. \tag{12}$$

The problem is indefinite for $k^2 > \pi^2$ assuming that k^2 is not an eigenvalue of the negative Laplacian [14]. The eigenvalues and eigenvectors of the system matrix K are respectively

$$\lambda_j = \frac{2(1-\cos j\pi h)}{h^2} - k^2 = \frac{4}{h^2} s_j^2 - k^2 \qquad j = 1, \ldots, N \tag{13}$$

and

$$v_j = \sqrt{2h}\,[\sin ij\pi h]_{i=1}^{N} \qquad j = 1, \ldots, N, \tag{14}$$

were $s_j = \sin \frac{j\pi h}{2}$. We also define the matrix V as the span of all eigenvectors

$$V = span\,\{v_i\}_{i=1}^{N} = \begin{bmatrix} | & & | \\ v_1 & \ldots & v_N \\ | & & | \end{bmatrix}. \tag{15}$$

Damped Jacobi iteration is given by

$$u^{i+1} = u^i + \omega D^{-1}\left(f - Ku^i\right). \tag{16}$$

$D = \left(\frac{2}{h^2} - k^2\right) I$ is the diagonal of K, and ω is a damping factor. The relaxation scheme (16) is equivalent to the following iteration on the error

$$e^{i+1} = (I - \omega D^{-1} K) e^i = S e^i, \tag{17}$$

where S is the smoothing iteration matrix (see (3)). S shares the same eigenvectors as K. Specifically,

$$S v_j = \left(1 - \frac{\omega}{\frac{2}{h^2} - k^2}\lambda_j\right) v_j = \mu_j v_j, \tag{18}$$

where

$$\mu_j = 1 - \omega\left(\frac{4s_j^2 - (kh)^2}{2 - (kh)^2}\right) \qquad j = 1, \ldots, N. \tag{19}$$

We denote $\Lambda_{a,b}$ as a diagonal matrix of size $(b-a+1)\times(b-a+1)$, with all sorted eigenvalues μ_j such that $a \le j \le b$. It is apparent that for indefinite cases some eigenvalues of S are outside the unit circle, i.e. $|\mu_j| > 1$ and thus magnify the error of the corresponding eigenmodes. We are now ready to analyze convergence behavior of the global-basis two level method. Assuming there are m eigenvalues greater than one (the last μ_j's), the global-basis method constructs an optimal prolongation operator

$$Q_{gb} = span\,\{v_i\}_{i=N-m+1}^{N} = \begin{bmatrix} | & & | \\ v_{N-m+1} & \ldots & v_N \\ | & & | \end{bmatrix}. \tag{20}$$

The eigenvalue problem can be compactly written for the indefinite modes as

$$S Q_{gb} = Q_{gb} \Lambda_{N-m+1,N}. \tag{21}$$

Substituting (20) into (4) with relation (21) and the ortho-normality condition of the eigenvectors, gives the following coarse grid projector

$$\begin{aligned} F_{GB} &= I - Q_{gb}\left(Q_{gb}^T K Q_{gb}\right)^{-1} Q_{gb}^T K = I - Q_{gb}\left(Q_{gb}^T K Q_{gb}\right)^{-1}\left(K Q_{gb}\right)^T \\ &= I - Q_{gb} Q_{gb}^T. \end{aligned} \tag{22}$$

This is a rank m perturbation of the identity and the projector is orthogonal.
The two level yielding iteration matrix can now be analytically resolved using (15), (16) and (22)

$$\begin{aligned} SF_{GB}S &= S\left[I - Q_{gb}Q_{gb}^T\right] S = V\Lambda_{1,N}V^T\left[I - Q_{gb}Q_{gb}^T\right] V\Lambda_{1,N}V^T \\ &= V\Lambda_{1,N}^2 V^T - V\Lambda_{1,N}\begin{bmatrix} 0_{N-m\times N-m} & \\ & I_{m\times m}\end{bmatrix}\Lambda_{1,N}V^T \\ &= V\begin{bmatrix} \Lambda_{1,N-m}^2 & \\ & 0_{m\times m}\end{bmatrix}V^T. \end{aligned} \tag{23}$$

(23) illustrates that all the indefinite modes are eliminated from the two level iteration matrix and thus the convergence of the two-level cycle is guaranteed. It is also evident that the asymptotic rate of convergence, in eq. (5), is governed by the highest eigenvalue of the smoother

$$c = \mu_{N-m}^2 = \left[1 - \omega\left(\frac{4s_{N-m}^2 - (kh)^2}{2 - (kh)^2}\right)\right]^2 \tag{24}$$

that is inside the convergence region. This suggests that substitution of a single level smoother by a more effective method may lead to both faster rates of convergence and fewer eigen-modes outside the convergence region.

4. ANALYSIS OF GENERALIZED GLOBAL BASIS (GGB) METHOD

We apply a standard multilevel method to (1). We show that the multilevel preconditioner filtered by the GGB method is positive definite and its convergence is governed by spectral characteristics of the convergent modes of the unfiltered operator. Figure 3 shows a one-dimensional mesh. Superscript zero denotes the coarse grid variables. We assume an odd

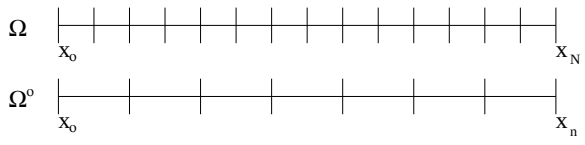

Figure 3. Fine and coarse meshes

number of fine grid points with $n = \frac{N-1}{2}$ being the coarse grid points. Linear interpolation is

used to transfer from coarse grid to fine grid (prolongation) $Q_{mg} : \mathbb{R}^n \to \mathbb{R}^N$,

$$Q_{mg} = \frac{1}{2}\begin{bmatrix} 1 & & & \\ 2 & & & \\ 1 & 1 & & \\ & 2 & \ddots & \\ & 1 & \ddots & 1 \\ & & \ddots & 2 \\ & & & 1 \end{bmatrix} \tag{25}$$

and Q_{mg}^T is used for the restriction operator (assuming periodic boundary conditions).
It is easy verified that the prolongation operator applied to a coarse grid eigenvector yields a linear combination of one smooth and one oscillatory mode of the fine grid [17]:

$$Q_{mg} v_j^0 = c_j^2 v_j - s_j^2 v_{N+1-j} \qquad j = 1, \ldots, N, \tag{26}$$

where $c_j = \cos\frac{j\pi h}{2}$ and s_j was given in (13). Similarly, Q_{mg}^T applied to a fine grid eigenvector yields

$$Q_{mg}^T v_j = \begin{cases} c_j^2 v_j^0 & j = 1, \ldots, n \\ 0 & j = n+1 \\ -c_j^2 v_{N+1-j}^0 & j = n+2, \ldots, N \end{cases} . \tag{27}$$

$\left(v_j^0, \lambda_j^0\right)$ is an eigenpair of the coarse grid matrix

$$Q_{mg}^T K Q_{mg} v_j^0 = \lambda_j^0 v_j^0 \tag{28}$$

given by $v_j^0 = [\sin 2ij\pi h]_{j=1}^n$ and

$$\lambda_j^0 = c_j^4 \lambda_j + s_j^4 \lambda_{N+1-j} = \frac{4}{h^2} s_j^2 c_j^2 - k^2 \left(s_j^4 + c_j^4\right). \tag{29}$$

Combining (26) - (27) one can show

$$T\left[v_j, v_{N+1-j}\right] = \left[v_j, v_{N+1-j}\right] \begin{bmatrix} 1 - c_j^4 \frac{\lambda_j}{\lambda_j^0} & s_j^2 c_j^2 \frac{\lambda_{N+1-j}}{\lambda_j^0} \\ s_j^2 c_j^2 \frac{\lambda_j}{\lambda_j^0} & 1 - s_j^4 \frac{\lambda_{N+1-j}}{\lambda_j^0} \end{bmatrix} \qquad j = 1, \ldots, n. \tag{30}$$

Recalling

$$S\left[v_j, v_{N+1-j}\right] = \left[v_j, v_{N+1-j}\right] \begin{bmatrix} \mu_j & \\ & \mu_{N+j-1} \end{bmatrix} \qquad j = 1, \ldots, n, \tag{31}$$

we can deduce

$$\begin{aligned} STS\left[v_j, v_{N+1-j}\right] &= \begin{bmatrix} \mu_j & \\ & \mu_{N+1-j} \end{bmatrix} \begin{bmatrix} 1 - c_j^4 \frac{\lambda_j}{\lambda_j^0} & s_j^2 c_j^2 \frac{\lambda_{N+1-j}}{\lambda_j^0} \\ s_j^2 c_j^2 \frac{\lambda_j}{\lambda_j^0} & 1 - s_j^4 \frac{\lambda_{N+1-j}}{\lambda_j^0} \end{bmatrix} \begin{bmatrix} \mu_j & \\ & \mu_{N+1-j} \end{bmatrix} \\ &= \left[v_j, v_{N+1-j}\right] \begin{bmatrix} \mu_j^2 \left(1 - c_j^4 \frac{\lambda_j}{\lambda_j^0}\right) & s_j^2 c_j^2 \frac{\lambda_{N+1-j}}{\lambda_j^0} \mu_j^2 \mu_{N+1-j} \\ s_j^2 c_j^2 \frac{\lambda_j}{\lambda_j^0} \mu_j^2 \mu_{N+1-j} & \mu_{N+1-j}^2 \left(1 - s_j^4 \frac{\lambda_{N+1-j}}{\lambda_j^0}\right) \end{bmatrix}. \end{aligned} \tag{32}$$

The above implies that the Fourier transform can be used to reduce the multigrid iteration operator to a block diagonal matrix, with 2×2 blocks. The eigenvectors of STS, ϕ_j, are a linear combination of $[v_j, v_{N+1-j}]$ with constants x_j and y_j

$$\phi_j = x_j v_j + y_j v_{N+1-j} \qquad j = 1, \ldots, n. \tag{33}$$

The eigenvalues are the union of the eigenvalues of the 2×2 matrices. Due to the projection, one of the eigenvalues of the 2×2 matrix is always zero and the other is

$$\theta_j = \mu_j^2 \left(1 - c_j^4 \frac{\lambda_j}{\lambda_j^0}\right) + \mu_{N+1-j}^2 \left(1 - s_j^4 \frac{\lambda_{N+1-j}}{\lambda_j^0}\right) \qquad j = 1, \ldots, n. \tag{34}$$

This implies that the rank of STS must be $n+1$ (the middle mode is left untouched). Figure 4 illustrates the first and second eigenvectors of STS.

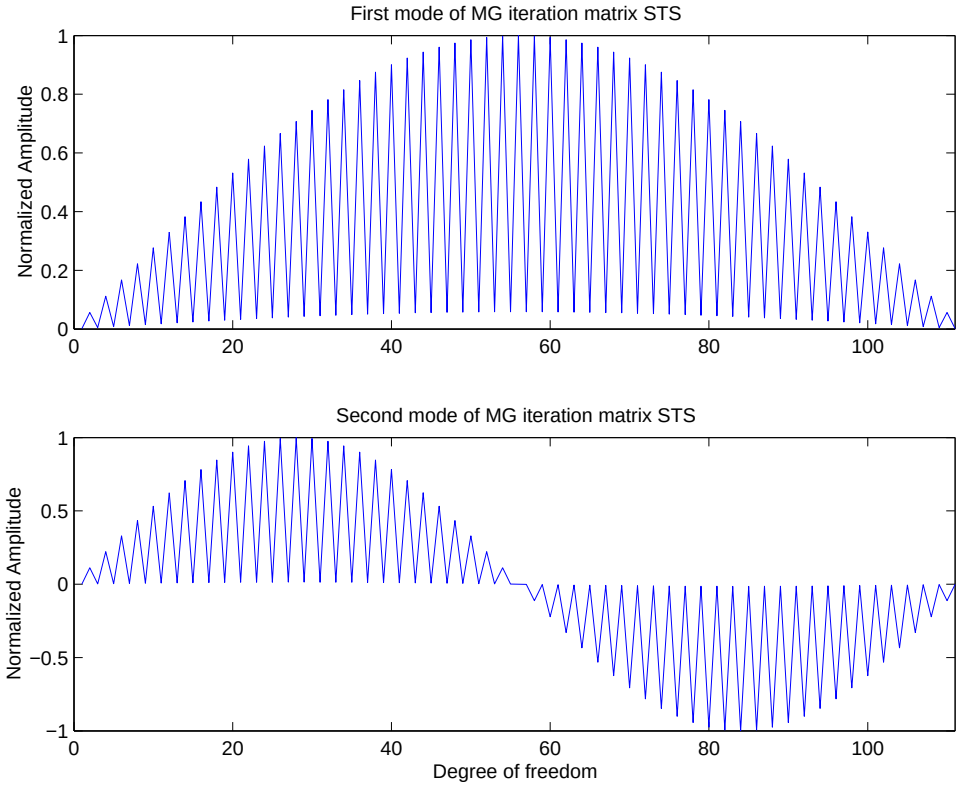

Figure 4. First and second eigenvectors of STS

We now examine how GGB corrects the multilevel scheme. We define the prolongation operator for the GGB scheme

$$Q_f = span\left\{\phi_i\right\}_{i=N-k+1}^{N} = \begin{bmatrix} | & & | \\ \phi_{N-k+1} & \ldots & \phi_N \\ | & & | \end{bmatrix}_{N \times k}. \tag{35}$$

The operator is spanned by all modes corresponding to eigenvalues $|\theta_j| > 1$, say k eigenvalues. In analogy to GB, the eigenvalue problem for STS can be written as

$$(STS)\,\Phi = \Phi\Theta, \tag{36}$$

where Φ is the complete set of eigenvectors and Θ is the eigenvalue matrix. There are n zero eigenvalues associated with STS, i.e. $dim\ \mathcal{N}(STS) = n$ ($dim\ \mathcal{N}$ is the dimension of the *nullspace*). We can now partition Φ in (36) into three subpaces: the null space, the convergent space and the indefinite space we want to filter out.

$$\begin{aligned} \text{null sapce} \qquad & \mathcal{N}(STS) = \{\Phi_0 : (STS)\Phi_0 = 0\} \\ \text{convergent space} \qquad & \mathcal{C}(STS) = \{\Phi_c : (STS)\Phi_c = \Phi_c\Theta_{n+1,N-k}\} \\ \text{indefinite space} \qquad & \mathcal{F}(STS) = \{Q_f : (STS)Q_f = Q_f\Theta_{N-k+1,N}\} \end{aligned} \tag{37}$$

We denote $\Theta_{a,b}$ as a diagonal matrix of size $(b-a+1)\times(b-a+1)$, with all eigenvalues θ_j on the diagonal such that $a \le j \le b$. The subspaces satisfy the following properties:

$$\begin{aligned} \mathcal{N} \oplus \mathcal{C} \oplus \mathcal{F} &= \Phi \\ \mathcal{N} \cap \mathcal{C} \cap \mathcal{F} &= 0 \\ \mathcal{N}\perp\mathcal{C}, \mathcal{N}\perp\mathcal{F}, \mathcal{C}\perp\mathcal{F}. & \end{aligned} \tag{38}$$

Rewriting (36), we have

$$(STS)[\Phi_0, \Phi_c, Q_f] = [\Phi_0, \Phi_c, Q_f] \begin{bmatrix} 0_{n\times n} & & \\ & \Theta_{n+1,N-k} & \\ & & \Theta_{N-k+1,N} \end{bmatrix}. \tag{39}$$

We now establish the convergence rate for the GGB method preconditioned by a standard multigrid. The eigenvalues of the GGB iteration matrix in equation (8) can now be analytically resolved. The eigenvalue problem reads

$$(STS)F_{GGB}(STS)\Psi = \Psi\Sigma. \tag{40}$$

Assuming that $\Psi = \Phi$ and substituting into (40) along with a similiarity transform, that is possible due to the properties given in (38), we get

$$\Sigma = \Phi^T(STS)F_{GGB}(STS)\Phi. \tag{41}$$

We form the following relation based on (36) and (38):

$$\Phi^T(STS) = \Phi^T(STS)\Phi\Phi^T = \Phi^T\Phi\Theta\Phi^T = \Theta\Phi^T. \tag{42}$$

Substituting the relations (36), (37), (39) and (42) into (41), yields

$$\begin{aligned} \Sigma &= \Phi^T(STS)F_{GGB}(STS)\Phi = \Theta\Phi^T F_{GGB}\Phi\Theta \\ &= \Theta\Phi^T\left[I - Q_f(Q_f^*KQ_f)^{-1}Q_f^*K\right]\Phi\Theta \\ &= \Theta\left[\Phi_0, \Phi_c, Q_f\right]^T\left[I - Q_f(Q_f^*KQ_f)^{-1}Q_f^*K\right]\left[\Phi_0, \Phi_c, Q_f\right]\Theta. \end{aligned} \tag{43}$$

Using the orthogonality properties (38), we arive at

$$\Sigma = \Theta\left[I, I, 0\right]\Theta = \begin{bmatrix} 0_{n\times n} & & \\ & \Theta^2_{n+1,N-k} & \\ & & 0_{k\times k} \end{bmatrix}. \tag{44}$$

This completes the proof. It also implies that the rate of convergence is governed by $c = \rho(\Sigma)$ or the highest eigenvalue within the convergent space $\mathcal{C}$, which is given by

$$\theta_k^2 = \left[\mu_{k+1}^2 \left(1 - c_{k+1}^4 \frac{\lambda_{k+1}}{\lambda_{k+1}^0}\right) + \mu_{N-k}^2 \left(1 - s_{N-k}^4 \frac{\lambda_{N-k}}{\lambda_{k+1}^0}\right)\right]^2 . \tag{45}$$

5. NUMERICAL RESULTS

We apply the GGB scheme to three classes of problems: a one-dimensional Helmholtz equation, a two-dimensional, elastic wave propagation problem, and a three-dimensional thermal-convection flow application. For the Helmholtz problem we use the standard multilevel method described in section 4. In the wave propagation example we apply Generalized Aggregation Multilevel (GAM) [7], [18] accelerated by the GGB method. Since this problem is time dependent, many linear systems are solved, and thus the cost of the eigenvectors computations is minimized (only done at the setup phase) and the eigenspace information is reused. This case is similar to problems with multiple right hand sides. For the third example we use a stabilized finite element code (MPsalsa) [24], [25] to obtain the algebraic nonsymmetric system. We apply the smoothed aggregation method [19], [20] from the ML package [21], as a multilevel preconditioner. The problems were run on an Intel(R) Xeon(TM) i686 processor with speed of 1700 MHz and a Linux operating system.
The solvers considered are: unpreconditioned GMRES, two-levels Global-Basis (GB), Generalized Global-Basis (GGB), GMRES preconditioned by GGB and GMRES preconditioned by the appropriate multilevel method. We use GMRES(m) to restart GMRES every m iterations [22], and the following right preconditioning

$$KP^{-1}(Pu) = f, \tag{46}$$

where KP^{-1} is the new preconditioned system that contains one or two sweeps of a multilevel cycle [11]. Right preconditioning guarantees that the norm of the preconditioned residual is the same as the unpreconditioned one [22]. For all examples we used the following stopping criteria

$$\frac{||r^i||_2}{||r^0||_2} < 10^{-6}, \tag{47}$$

where $r^i = f - Ku^i$ is the residual at iteration i. Finally, we discuss scalability and computational cost of GGB.

5.1. 1D Helmholtz preconditioned by standard multilevel

Figure 5 presents the convergence behavior of the solvers applied to a one-dimensional Helmholtz problem given in equations (10) and (12) for varying wave number k. The governing equations and boundary conditions are given in Sections 3 and 4. As shown in Table I, the indefiniteness of the system can be controlled by varying the value of k. Highly indefinite systems are obtained for large wave numbers. In the present study GMRES is not restarted. We use two cycles of the standard multilevel method to precondition GMRES so that this can be compared with GGB which uses two multilevel cycles per iteration. As a smoother for the two-grid method we use a simple damped Jacobi smoother with damping factor $\omega = \frac{2}{3}$. The

system consists of 411 equations.
Neither the unpreconditioned GMRES nor the Global-Basis method (GB) performs well on this problem. Unpreconditioned GMRES converges only after 411 (system size) iterations and therefore is not presented. On the other hand, excellent performance has been observed for both the two-grid preconditioned GMRES and the GGB methods. However, when k is large, GMRES preconditioned by GGB outperforms all methods (see Figure 5). Table I shows the number of eigenvectors used for the GB and GGB methods versus the number of negative eigenvalues of the system. It corresponds to the number of eigenvalues greater than 0.95 in absolute value. It is clear that not is the GGB convergence rate faster than the GB rate, but it requires less eigenvectors.

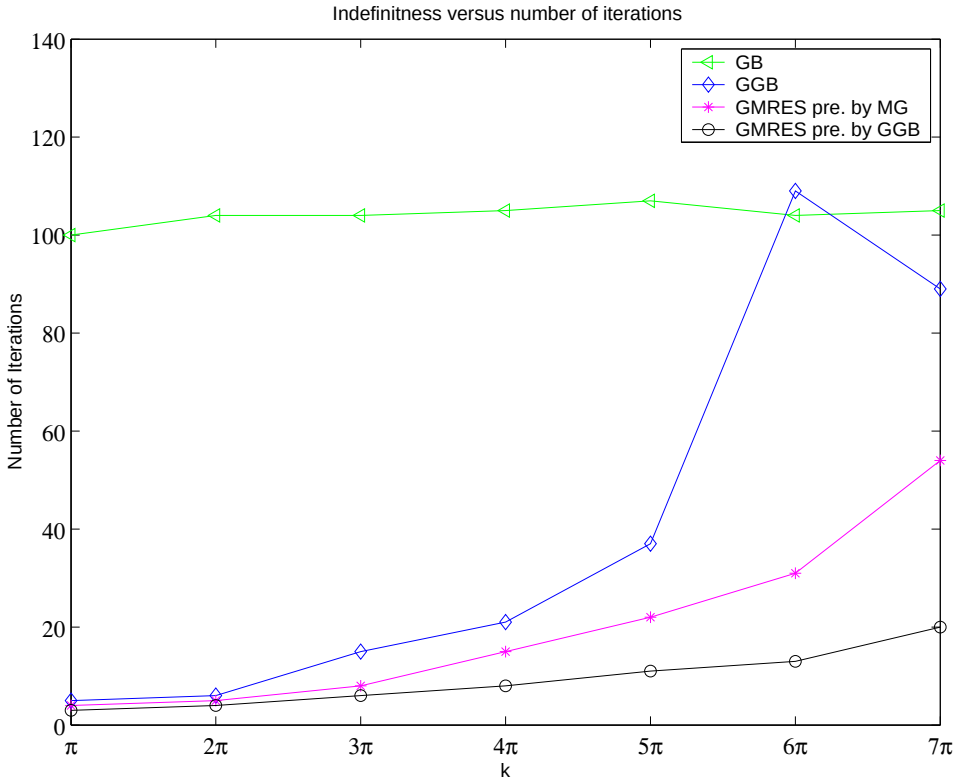

Figure 5. Number of Iterations required by the solvers to converge as the wave number (indefinitness) of the system increases

Figure 6 demonstrates the convergence of the solvers for a highly indefinite system ($k = 130\pi$) with 130 negative eigenvalues of the shifted Laplacian. It can be seen that GMRES without restarting converges only in the last iteration, i.e. in 411 iterations. GGB converges slightly better than GB but remarkably requires only $\frac{1}{6}$ of the eigenvectors. The performance of GMRES preconditioned by multigrid stalls in the first iterations and eventually converges after 54 iterations. GGB accelerated by GMRES does not stall and converges in 20 iterations. By incorporating the "bad" modes in an additional coarse level, GMRES can be significantly

Table I. Number of eigenvectors used for convergence studies, compared to the number of indefinite eigenvalues of the system

	GB	GGB	Negative eigenvalues of K
$k = 10\pi$	52	1	10
$k = 30\pi$	58	1	30
$k = 50\pi$	70	2	50
$k = 70\pi$	85	5	70
$k = 90\pi$	102	8	90
$k = 110\pi$	121	13	110
$k = 130\pi$	167	27	130

accelerated.

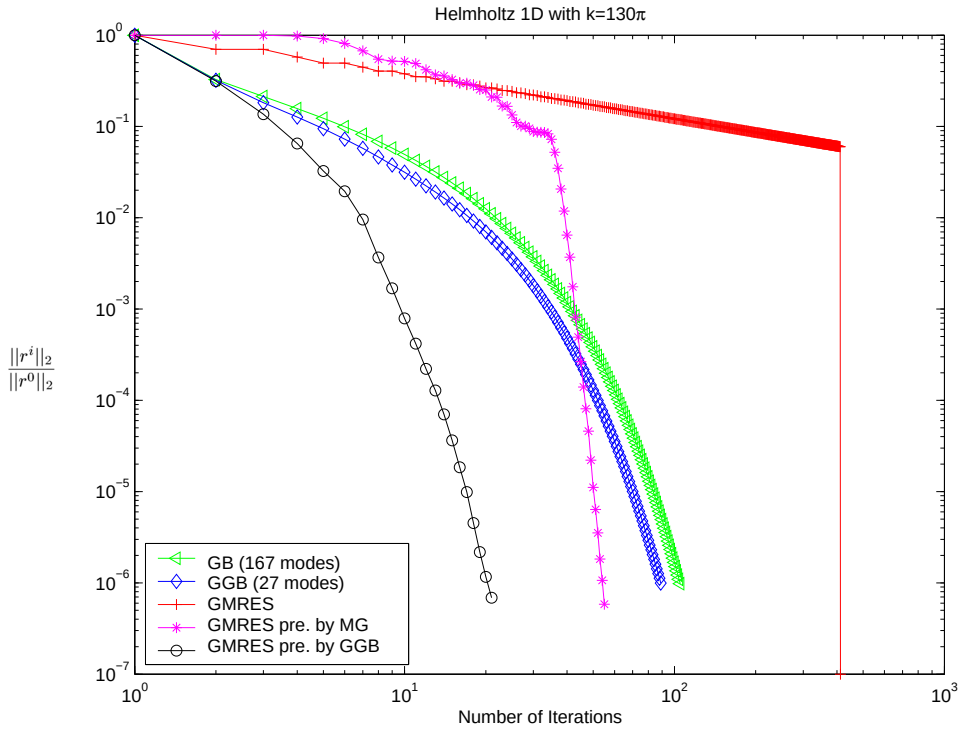

Figure 6. Convergence behavior of the solvers for 1D Helmholtz equation with $k = 130\pi$

5.2. 2D Elastic wave propagation preconditioned by Generalized Aggregation Multilevel (GAM)

We next study performance of various solvers for a 2D elastic wave propagation problem in a cracked membrane with softening zones at the tips of the crack. The governing equation for transverse displacement u in the domain $(x, y) \in \Omega$ with boundary Γ is:

$$\rho(x,y)\frac{\partial^2 u}{\partial t^2} = \nabla \cdot (\mu(x,y)\nabla u) + f \qquad \text{in } \Omega, \tag{48}$$

where t is time,ρ is the density, μ is the material constant (usually greater than zero) and f is the forcing function. We neglect structural damping. The boundary conditions for the problem are given as

$$\begin{aligned} u &= 0 && \text{on } \Gamma \\ \frac{\partial u}{\partial n} &= 0 && \text{on } \ s^+, s^- \end{aligned} \tag{49}$$

s^+, s^- are the two sides of the cracks boundaries. The initial conditions are set to zero for both the displacement and the displacement velocity

$$u(t=0) = \dot{u}(t=0) = 0. \tag{50}$$

A standard Galerkin finite element semi-discretization in space produces a system of ordinary differential equations of the form

$$M\ddot{u} + Ku = f, \tag{51}$$

where M is the mass matrix, K is the stiffness matrix and f is the force vector obtained by the assembly operation over the local element domains. Figure 7 shows the finite element meshes considered. The membrane is constrained all around and a point force is applied at a single node. Details of the finite element mesh at the tip of the crack are shown in Figure 8. The red dashed line illustrates the crack interface. The circles around the crack tips define the damage regions for which μ is negative. This causes the problem to become indefinite.

An Implicit Newmark predictor-corrector time integration method is used with parameters $\beta = \frac{1}{4}$ and $\gamma = \frac{1}{2}$ that define the average acceleration method or the trapezoidal rule (for more details see [23]). The integration scheme leads to the system of linear equations (1) at each time step. The parameters ρ and μ are chosen to be constants within the element.

The multilevel method used as preconditioner is the Generalized Aggregation Method (GAM) [18]. We use two GAM cycles to precondition GMRES, similar to two cycles in GGB. The method constructs aggregates using element connectivity information only. Aggregates are defined as a group of connected elements that do not touch Dirichlet boundary conditions. Once the aggregation process is complete (details can be found in [7]) aggregates stiffness matrices K_A are assembled. We then solve the aggregate eigenvalue problem $K_A\Psi_A = \Psi_A\Sigma_A$ for the eigenpair (Ψ_A, Σ_A) and span the prolongation operator Q_{GAM} by the eigenvectors corresponding to the lowest eigenvalues. The number of aggregate eigenvectors used must be equal or greater than the size of the *nullspace* of the aggregate, i.e. greater or equal to the rigid body modes. Figure 9 shows the formation of such large aggregates.

The results are obtained for normalized variables $\rho_A = 2,\ \rho_B = 1$ where subscripts A and B indicate material type (see Figure 8),$\mu = 5$ outside the damaged region and $\mu = -5$ inside. The damaged region is set to have radius of 3 units. We use 5 time steps of $\Delta t = 5[sec]$ to

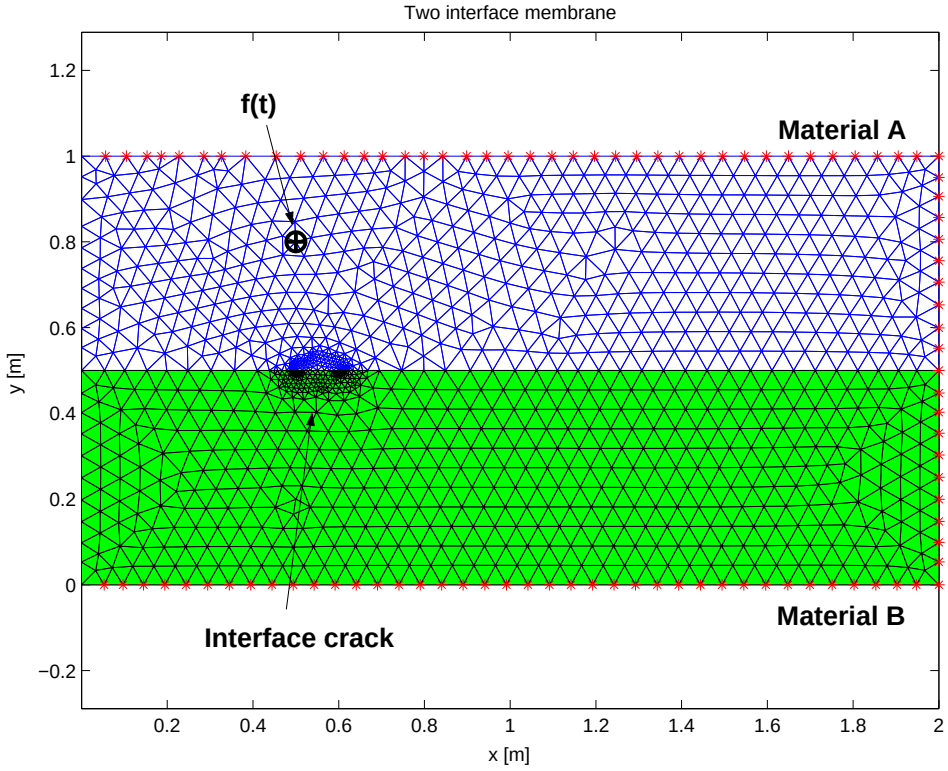

Figure 7. Two materials meshed membrane with a crack between them. The problem contains 1113 unknowns.

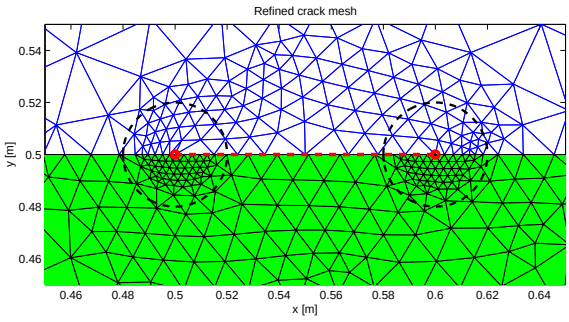

Figure 8. Zoom of the crack with damage zones

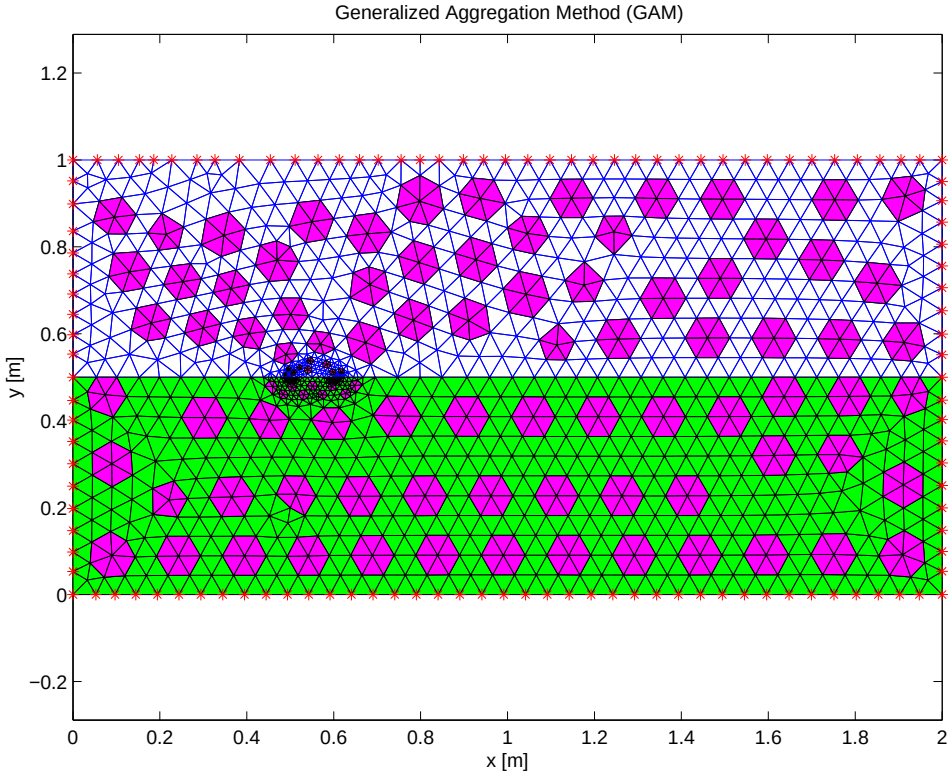

Figure 9. Formation of large aggregates used to span the prolongation operator

march with the Newmark scheme. Figure 10 presents the overall number of iterations versus the step number of the various solvers, and Figure 11 illustrates the convergence at the first time step.

We use GMRES without restarting. Unpreconditioned GMRES is not shown due to poor performance. It is evident that the iterations count increase linearly over the time integration as only the right hand side varies. We use an Implicit Restarted Arnoldi method [15] implemented in ARPACK [16] for eigenvectors calculations. Again, the fastest method to converge is GMRES preconditioned by GGB. In terms of *CPU* time, GGB requires the additional projection step within each iteration and the time to compute the eigenvectors. The additional projection is inexpensive compared to the multigrid iterations. The eigenvector calculation time is also negligible as it is only performed in the first step and then reused throughout the time integration. This is in contrast to GMRES which builds information during the computation of a linear solve but then normally throws this information away for subsequent linear solve. Restarting the GMRES iteration makes it even harder and the computational cost increases tremendously, where sometimes convergence is not guarantied.

This example illustrates the main advantage of the GGB method when the matrix is constant but the right hand side varies. The GGB method computes eigenspace information that it is able to reuse over many linear solves. Examples of multiple right hand side solves come from

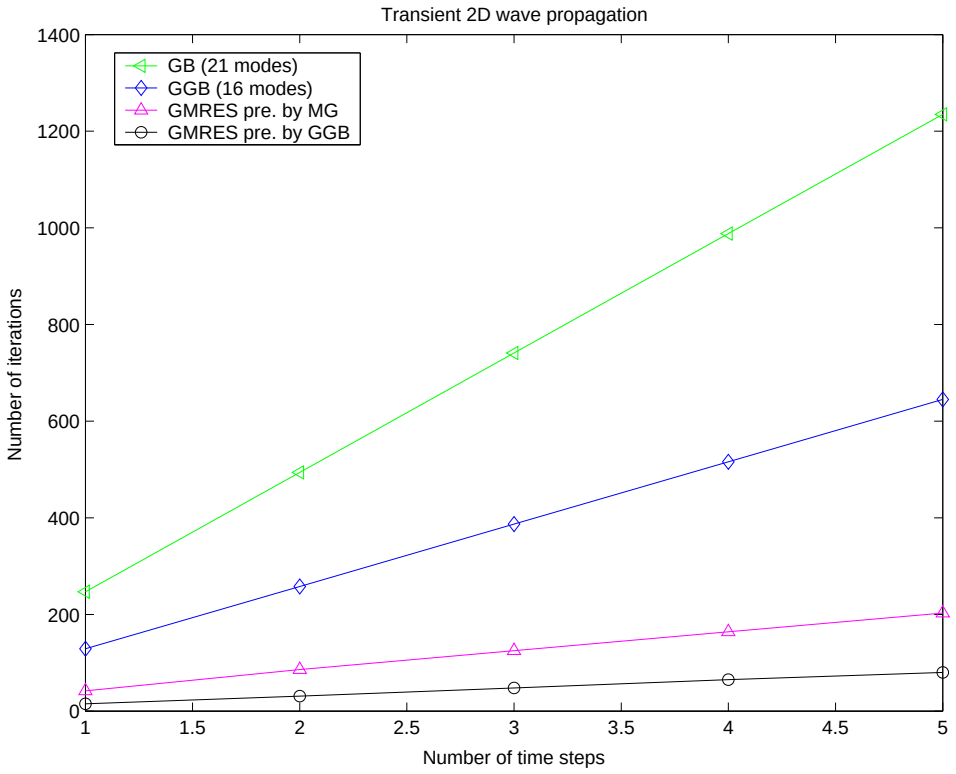

Figure 10. Number of iterations until convergence of the solvers for a transient 2D elastic wave propagation

linear transient problems, shift and invert eigenvalue calculations using an iterative solve for the "invert" phase, nonlinear problems when one uses a modified Newton scheme, Monte-Carlo simulations, etc.

5.3. Thermal-convection, flow preconditioned by Smooth Aggregation

In this section we demonstrate the performance of GGB method applied to steady, thermal-convection flow. The governing PDEs are the following Navier-Stokes with thermal energy equations

$$\text{Momentum} \qquad \rho\mathbf{u}\cdot\nabla\mathbf{u} - \nabla\cdot\mathsf{T} - \rho\mathbf{g} = 0 \tag{52}$$

$$\text{Total mass} \qquad \nabla\cdot(\rho\mathbf{u}) = 0 \tag{53}$$

$$\text{Thermal energy} \qquad \rho\hat{C}_p\mathbf{u}\cdot\nabla T + \nabla\cdot\mathbf{q} = 0 \tag{54}$$

The unknown quantities are $\mathbf{u}$ the fluid velocity vector, P the hydrodynamic pressure and T the temperature. ρ, $\mathbf{g}$, and $\hat{C}_p$ are respectively, the density, the gravity vector and the specific heat at constant pressure. The Boussinesq approximation is used for representing the body force term.

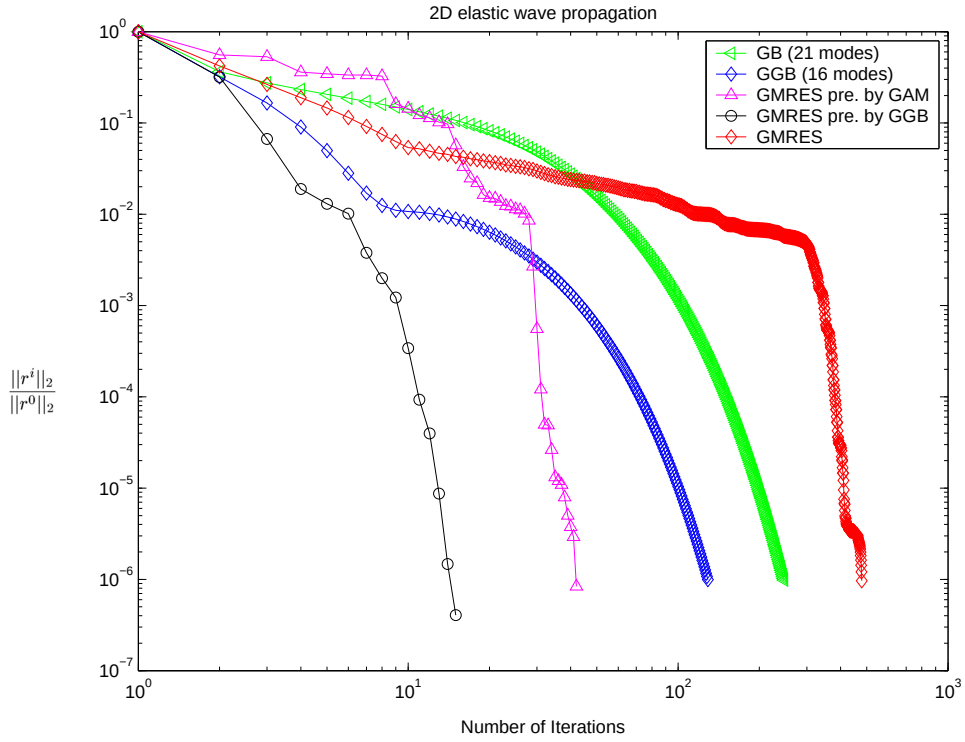

Figure 11. Number of iterations for convergence of the solvers at the first time step

The necessary constitutive equations for T and $\mathbf{q}$ are

$$\begin{aligned} &\text{Stress tensor} && \mathsf{T} = -P\mathbf{I} + \Upsilon = -P\mathbf{I} + \mu(\nabla\mathbf{u} + \nabla\mathbf{u}^T) \\ &\text{Heat flux} && \mathbf{q} = -\kappa\nabla T \end{aligned} \tag{55}$$

where μ is the viscosity and κ is the thermal diffusivity. (52)-(54) are approximated by a Galerkin Least Squares formulation. The resulting nonlinear system of equations gives rise to a system of coupled, nonlinear and nonsymmetric (definite) algebraic equations. We employ MPSalsa [24], [25] to generate the system of equations, and a smooth aggregation multilevel method [19] , [20] implemented in ML package [21] as preconditioner to GMRES. Due to nonsymmetry of the problem our experience shows that best performance is obtained when piecewise constants are used as grid interpolants (unsmoothed aggregation). The solvers are applied to the following cases:
(i) 3D flow between two finite length concentric cylinders.
(ii) Large-scale 3D flow in a cube.

5.3.1. 3D flow between two finite length concentric cylinders In this case the fluid is confined between two cylinders with a outer to inner radius ratio of $\frac{8}{3}$ and a ratio of length to outer diameter of 1. A no-slip condition is enforced on all boundaries. A hot temperature is set to the inner cylinder and a cold temperature is set to the outer cylinder. The ends of the annular

region are insulated. The momentum transport, total mass conservation and energy transport are given in equations (52)-(54). We set the Prandtl number to be 1.0 and the Rayleigh number to be 4×10^3, which is relatively small but demonstrates the behavior. The mesh described in Figure 12 consists of 768 elements with 5400 unknowns. The results are presented for the Jacobian arising in the last Newton step.
Figure 13 presents the eigenvalues spectrum of the Jacobian and the multilevel iteration matrix, respectively. Figure 14 and Table II illustrates the convergence behavior of the various solvers.

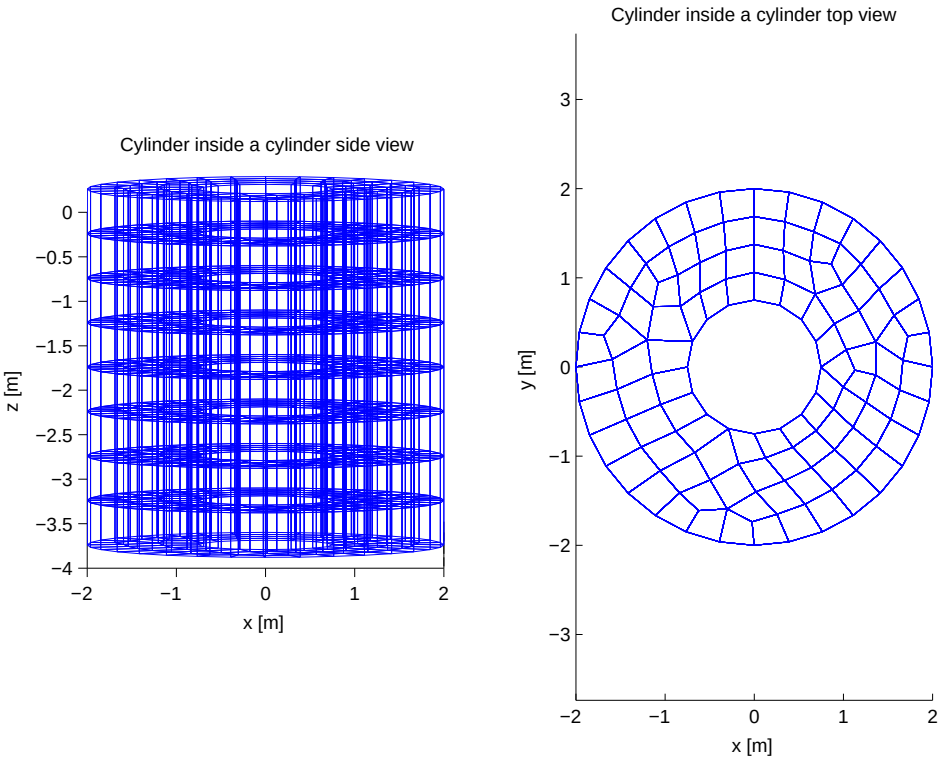

Figure 12. Concentric Cylinders meshed with 768 elements. Side and top views.

Two multilevel sweeps are applied to precondition GMRES, with a two-level cycle each. The smoother used in the multilevel aggregation method is a damped Jacobi with $\omega = \frac{2}{3}$. GMRES is restarted every 100 iterations in the case of the multilevel preconditioner and every 20 iterations in the case of GGB preconditioner. In this way GMRES(100) without GGB uses approximately the same storage as GMRES(20) with GGB using 80 modes.
The results clearly show that GGB accelerated by GMRES has the best performance, converging in just 51 iterations. GGB alone converges faster than GMRES preconditioned by aggregation. We use 80 largest eigenvectors as the additional coarse grid correction for both GB and GGB. We also note that GMRES(100) preconditioned by aggregation did not converge in a reasonable time and therefore the restarting was increased to 150. To this end it is important to mention that GGB plays an important complementary role to GMRES.

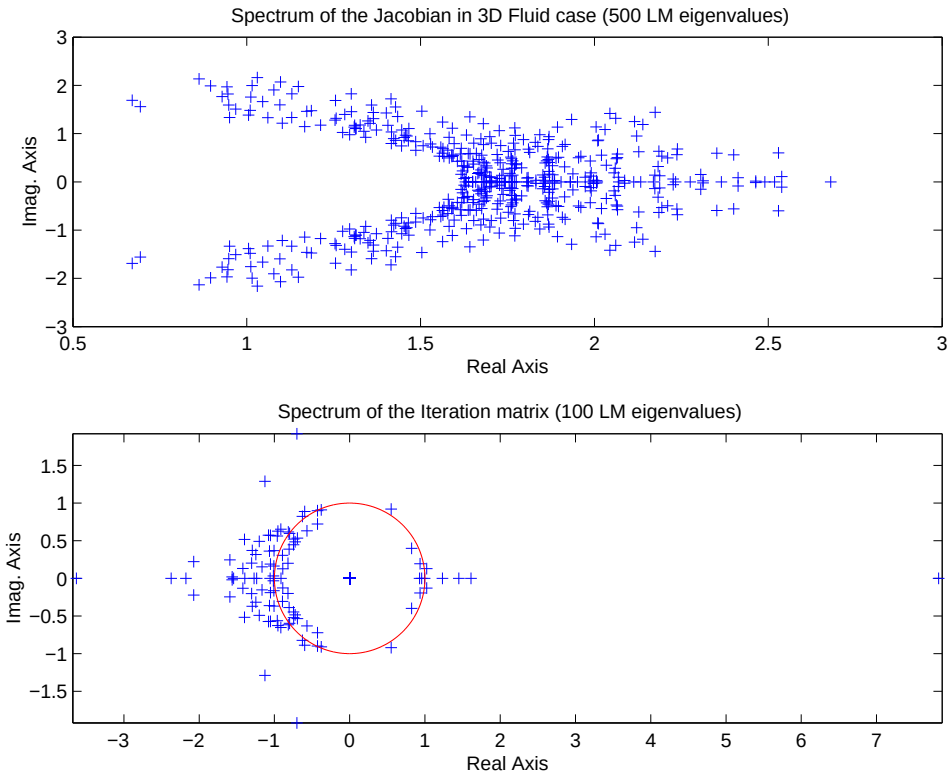

Figure 13. The spectrum of the Jacobian and multilevel iteration matrices, respectively, at the last Newton step.

Restarting GMRES is crucial for convergence. However, once GMRES is restarted, all the prior information is lost, causing temporal stall (see Figure 14) until it reconstructs a sufficient space for further reduction of the error. When GGB is used, the information about the indefinite modes which span the most difficult space is stored (the space that is slow to converge), which enables GMRES to recover rapidly and to converge faster.

Table II. Total number of iterations until convergence for 3D cylinder-inside-cylinder thermal-convection problem.

	GB	GGB	GMRES(150) pre. by ML	GMRES(20) pre. by GGB
Total iterations	440	76	441	51

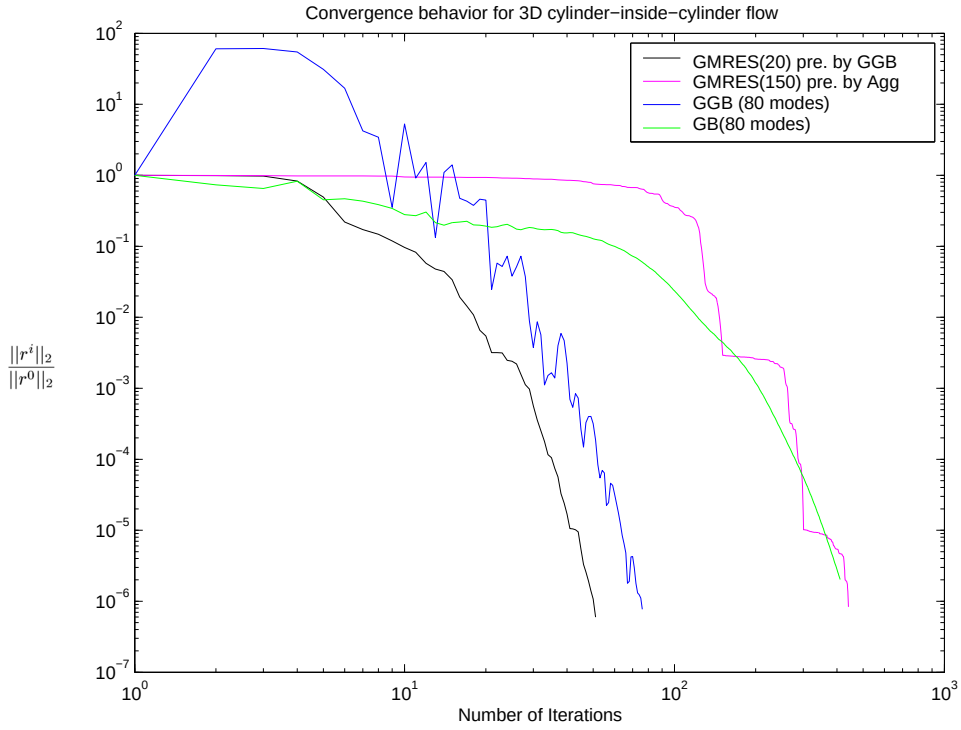

Figure 14. Comparison of solvers for 3D cylinder-inside-cylinder thermal-convection problem.

5.3.2. Large-scale 3D flow in a cube Finally we compare the performance of the solvers applied to a 3D thermal-convection flow in a cube. A no-slip condition is enforced on all surfaces. A hot temperature is set to one face of the cube and a cold temperature is set to the parallel. The governing equations are given in (52)-(54). We set Rayleigh number to 2.4×10^5 and the Prandtl number to 1.0. $ilu(0)$ is applied as a smoother to the aggregation method at each level excluding the coarse one. We consider cubes consisting of $16 \times 16 \times 16$ and $32 \times 32 \times 32$ mesh elements that yield systems with $24,565$ and $179,685$ number of unknowns, respectively. A 3-level aggregation method is applied to the $16 \times 16 \times 16$ cube and a 4-level to the $32 \times 32 \times 32$ cube. All methods are applied as preconditioners to GMRES(30). We also introduce a modified GGB scheme (MGGB) where the eigenspace is not always recomputed for each linear solve of Newton's method. In the current cube examples, the eigenspace is computed only once at the first Jacobian and the GGB cycle (see Figure 2) is performed with the same prolongation operator for all the remaining linear solves. This idea stems from the assumption that the Jacobian does not change too much during the Newton iterations. Actually, in this example the second Jacobian is significantly different from from the first Jacobian. However, the GGB method still performs quite well.

We present results for $16 \times 16 \times 16$ cube in Figure 15 and Table III and for $32 \times 32 \times 32$ in Figure 16 and Table IV. The results reported include total CPU time, eigen solver time, the pracentage of the eigensolver time from the overall time and iterations count of the linear

solver. It is evident that GGB and MGGB outperformed aggregation multilevel method both in CPU time and in iterations count. For both cubes, MGGB preformed the best in terms of overall time due to the inexpensive cost of the eigensolutions compared to GGB as the projection is computed only once. The eigen solve time slightly decreases as the problem size increases. In terms of iterations count, GB and GGB had about the same count for the $16 \times 16 \times 16$ cube but for the $32 \times 32 \times 32$ cube GGB outperformed all the other methods. In general, it is best to recompute the eigenspace if the Jacobian changes significantly. In fact, in this example the best run times are obtained by recomputing the eigenspace after the second and third Jacobians (not shown in this paper), as the Jacobian changes in the first couple of iterations.

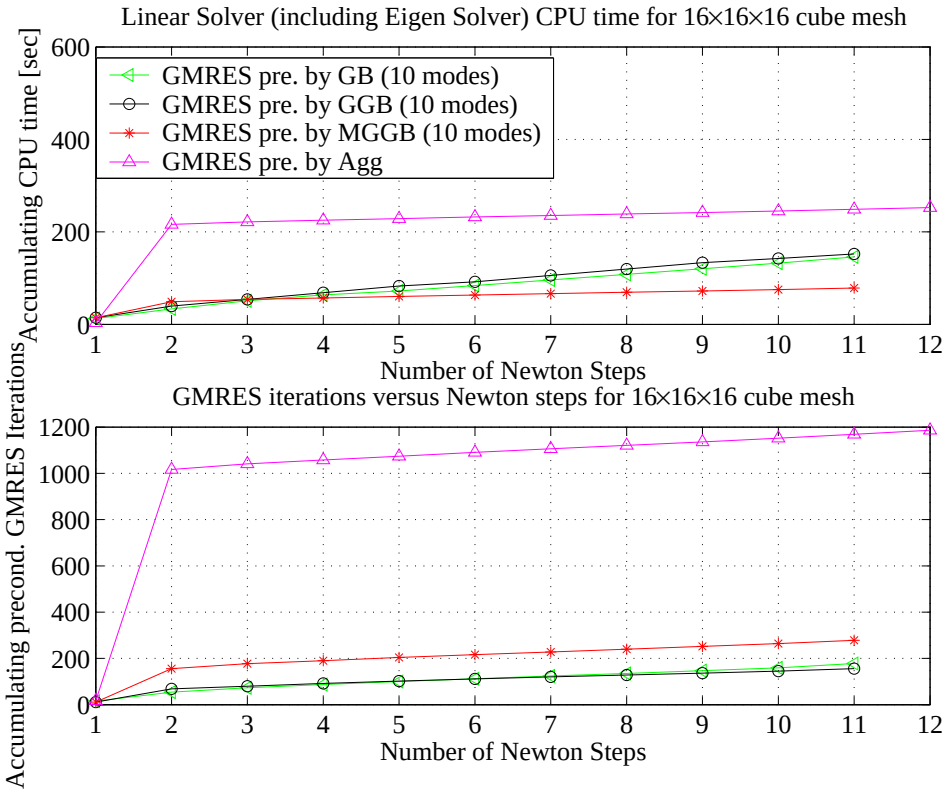

Figure 15. CPU time and linear solver iteration count of various solvers for thermal-convection flow in $16 \times 16 \times 16$ cube with $24,565$ unknowns. 3-levels aggregation method is applied.

5.4. Scalability and computational cost

We focus on studying the scalability properties of the GGB filter. The computational efficiency of the GGB filter is tightly linked to the performance of the eigen solver which could dominate the entire computational cost. Clearly, the filter is most attractive in applications with multiple right hand sides, such as linear transient problems or nonlinear problems where the projector

Table III. *CPU* time and iteration summary for thermal-convection flow in $16 \times 16 \times 16$ cube with $24,565$ unknowns. 3-levels aggregation method is applied.

	GMRES(30) pre. by ML	GMRES(30) pre. by GB	GMRES(30) pre. by GGB	GMRES(30) pre. by MGGB
Total *CPU* time [sec]	481.49	352.29	367.01	295.99
Total Eigen time [sec]	-	112.10	112.66	11.65
Percentage of Eigen work	-	31.82 %	30.70 %	3.94 %
Total linear solves iter.	1186	178	156	279
Aver. num. of linear solves	107.81	16.18	14.18	25.36

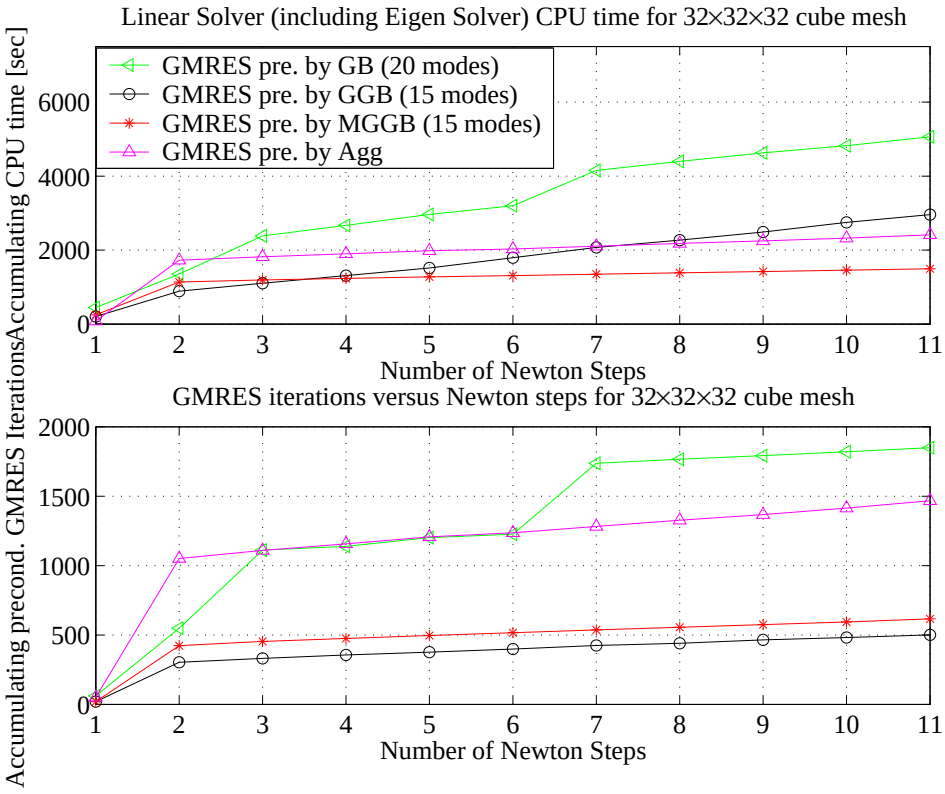

Figure 16. *CPU* time and linear solver iteration count of various solvers for thermal-convection flow in $32 \times 32 \times 32$ cube with $179,685$ unknowns. 4-levels aggregation method is applied.

Table IV. *CPU* time and iteration summary for thermal-convection flow in $32 \times 32 \times 32$ cube with $179,685$ unknowns. 4-levels aggregation method is applied.

	GMRES(30) pre. by MG	GMRES(30) pre. by GB	GMRES(30) pre. by GGB	GMRES(30) pre. by MGGB
Total *CPU* time [sec]	6998.40	9191.40	7183.63	5817.99
Total Eigen time [sec]	-	2306.10	2033.50	206.9
Percentage of Eigen work	-	25%	28.31%	3.56%
Total linear solves iter.	1521	1849	501	616
Aver. num. of linear solves	126.75	168.09	45.54	56.00

can be kept fixed over a certain number of iterations. However, the GGB filter offers significant computational advantages even for problems with a single right hand. Further, when a fixed number of modes is used, the cost of using the GGB filter is scalable. Table V illustrates the scalability properties of the GGB preconditioner with 10 modes using GMRES(80) on several 2D thermal-convection flows. The Rayleigh number is set to 1.0×10^5. The results presented

Table V. Scalability of GGB applied to several 2D flow in a box. The number of modes is set to 10

	Unknowns	Levels	Total CPU time [sec]	Ave. No. of iter.	Eig. CPU time [sec]	Percentage of Eigen work
10×10	484	2	2.09	5.90	1.00	47.85%
32×32	4356	3	13.71	10.50	5.50	40.12%
64×64	16900	3	90.12	16.10	49.98	55.46%
100×100	40804	4	214.45	26.10	97.35	45.40%
128×128	66564	4	334.41	35.50	109.55	32.76%
256×256	264196	5	1783.30	63.60	404.85	22.70%

in Table V indicate a decrease in the cost of the eigensolver relative to the total computational cost. That is, the setup cost associated with GGB becomes less significant for larger problems. Of course, the number of iterations increases if the GGB modes are kept fixed as the mesh is refined. However, this is consistent with numerous observations [11] for nonsymmetric and highly indefinite systems. The lack of scalability of multilevel methods for highly indefinite nonsymmetric systems can be explained by observing the spectrum of the multilevel iteration matrix (the highest 300 eigenvalues shown in Figure 17). The spectrum is depicted for the first Newton iteration. It can be seen that as the problem increases more eigenvalues appear to be outside the unit circle and there is a significant clustering of eigenvalues close to the unit circle. In principal, one can increase the number of global basis modes selected with increase in problem size to preserve the scalability of the multilevel method. Unfortunately, such a strategy has been found to be suboptimal in terms of the total computational cost. The important point is that though mesh independent convergence on indefinite problems is not attained in a computationally attractive fashion, a fixed number of GGB modes does significantly accelerate convergence of an existing scheme.

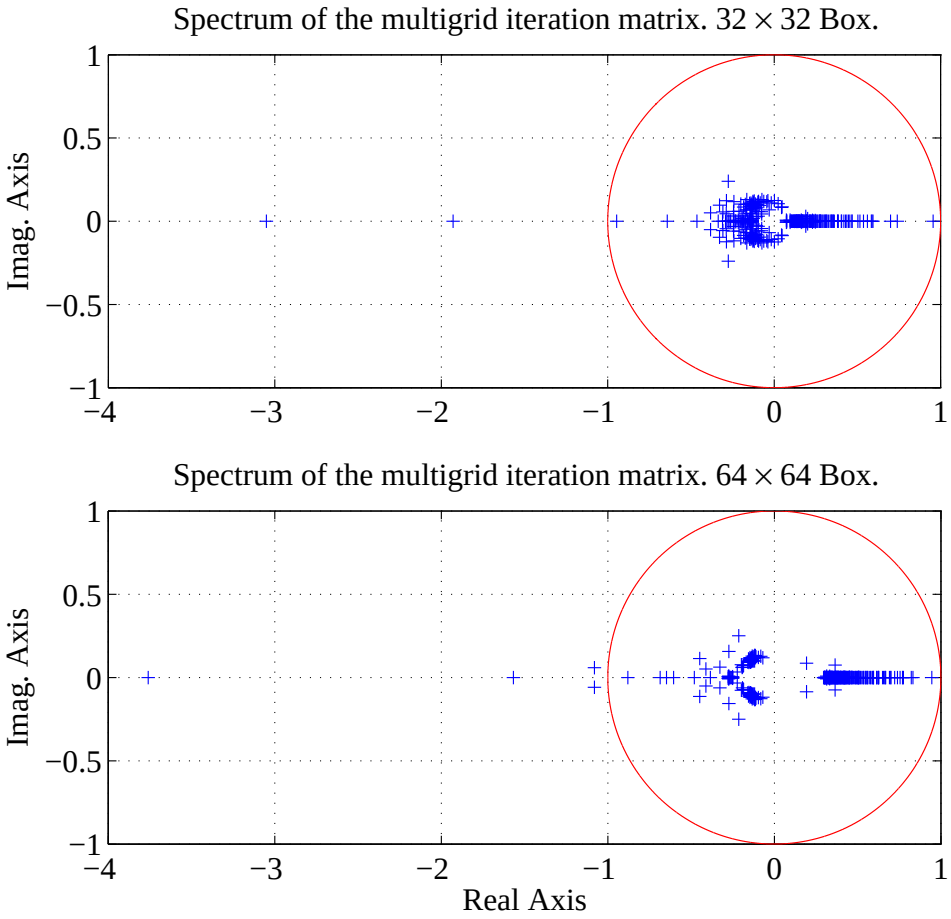

Figure 17. The spectrum of the multilevels iteration matrix, for problem sizes 32×32 and 64×64, respectively, at the first Newton step.

6. CONCLUSIONS

In this manuscript we introduce and study the Generalized Global Basis (GGB) method. We show that GGB is robust for variety of problems considered and can be used to accelerate any multilevel method. If sufficient number of modes is taken, convergence is ensured for indefinite and/or nonsymmetric systems. Theoretical convergence estimates have been obtained for a 1D Helmholtz problem. Numerical experiments have been conducted for a 1D Helmholz equation, 2D wave propagation in a membrane and a 3D thermal-convection flow. In all examples considered GMRES preconditioned by GGB has been found to have the best performance. The GGB is most attractive for problems with multiple right hand sides such as linear transient problems, or nonlinear schemes where the Jacobian varies only slightly, since the indefinite eigenspace information is stored and reused. This also enhances restarted GMRES strategies.

ACKNOWLEDGEMENTS

This work was supported by the ASCI program at Sandia National Laboratories under contract number DE-ACD4-94AL85000.
The authors wish to thank Rich Lehoucq for helpful suggestions on employing the Arnoldi method and help with ARPACK.

REFERENCES

1. Fish J, Qu Y. Global-basis two-level method for indefinite systems. Part 1:convergence studies. *International Journal for Numerical Methods in Engineering* 2000; **49**:439–460.
2. Qu Y, Fish J. Global-basis two-level method for indefinite systems. Part 2:computational issues. *International Journal for Numerical Methods in Engineering* 2000; **49**:461–478.
3. Brandt A, Taásan S. Multigrid methods for nearly singular and slightly indefinite problems. In *Multigrid Methods II,* Hackbusch W., Trottenberg U. In *Lecture Notes in Mathematics,* Springer-Verlag: Berlin, 1986; 99–121.
4. Bramble JH, Kwak DY, Pasciak JE. Uniform convergence of multigrid V-cycle iterations for indefinite and nonsymmetric problems *SIAM Journal of Numerical Analysis* 1994; **31**(6):1746–1763.
5. Shapira Y. Multigrid techniques for highly indefinite equations. In *proceedings 1995 Copper Mountain Conference on Multigrid Methods,* 1995.
6. Fish J, Qu Y, Suvorov A. Towards robust two-level methods for indefinite systems. *International Journal for Numerical Methods in Engineering* 1999; **45**:1433–1456.
7. Giddings TE, Fish J. An algebraic two-level preconditioner for asymmetric, positive-definite systems. *International Journal for Numerical Methods in Engineering* 2001; **52**:1443–1463.
8. Tuminaro RS, Tong CH, Shadid J, Devine KD, Day DM. On a multilevel preconditioning module for unstructured mesh Krylov solvers: Two level Schwartz. *Communications in Numerical Methods in Engineering* 2002; **18**:363–389.
9. Oosterlee CW, Washio T. On the use of multigrid as preconditioner In *Proceedings of the 9th int. conf. On domain decomposition methods,* P.E. Bojorstad et al: Norway 1996; 441-448.
10. Zavorin I, Oĺeary DP, Elman H. *Stagnation of GMRES* 2001; UMCP-CSD; CS-TR-4296.
11. Oosterlee CW, Washio T. An evaluation of parallel multigrid as a solver and as a preconditioner for singularly perturbated problems. *Journal on scientific computing* 1998; **19**:87–110.
12. Shroff GM, Keller HB. Stabilization of unstable procedures: The Recursive Projection Method. *SIAM Journal on numerical analysis* 1993; **30**:1099–1120.
13. Carpentieri B., Duff IS, Giraud L. A class of spectral two-level preconditioners. *Technical report TR/PA/02/55* CERFACS, Toulouse, France 2002.
14. Elman HC, Ernst OG, Oĺeary DP. A multigrid method enhanced by Krylov subspace iteration for discrete Helmholtz equations. *SIAM Journal on scientific computing* 2001; **23**:1921–1315.
15. Sorensen DC. Implicitly restarted Arnoldi/Lanczos method for large scale eigenvalue calculations. In *Parallel Numerical Algorithms,* Keyes DE, Sameh A, Venkatakrishnan V. eds, Kluwer: Dordrecht 1996.
16. Lehoucq R, Sorensen DC, Yang C. *ARPACK User's Guide: Solution of large scale eigenvalue problems with Implicitly Restarted Arnoldi methods* SIAM: April 1998;
17. Briggs WL. *A multigrid tutorial* SIAM: Philadelphia, 1987;
18. Fish J, Belsky V. Generalized Aggregation Multilevel Solver. *International Journal for Numerical Methods in Engineering* 1997; **40**:4341–4361.
19. Vanek P, Mandel J, Brezina M. Algebraic multigrid by the smoothed aggregation for second and fourth order elliptic problems. *Computing* 1996; **56**:179–196.
20. Tuminaro R, Tong C. Parallel smoothed aggregation multigrid: Aggregation strategies on massively parallel machines. In *SupperComputing 2000 Proceedings,* Donnelley J. 2000.
21. Tong C, Tuminaro RS. ML 2.0 smoothed aggregation users guide. *Technical report SAND2001-8028* Sandia National Laboratories, Dec. 2000.
22. Saad Y. *Iterative Methods for sparse linear systems* (2nd edn): 2000.
23. Hughes TJR. *The finite element method-linear static and dynamic finite element analysis* Prentice Hall: 2000.
24. Shadid J, Moffat HK, Hutchinson SA, Hennigan GL, Devine KD, Salinger AG. MPSalsa-A finite element computer program for reacting flow problems. Part 1-Theoretical development *Technical report SAND95-2752* Sandia National Laboratories, May 1996.

25. Salinger A, Devine K, Hennigan G, Moffat H, Hutchinson S, Shadid J. MPSalsa-A finite element computer program for reacting flow problems. Part 2: User's guide. *Technical report SAND96-2331* Sandia National Laboratories, Sep. 1996.