

Solving Large-scale Eigenvalue Problems in SciDAC Applications

Chao Yang

Computational Research Division, Lawrence Berkeley National Lab, Berkeley, CA, 94597

E-mail: cyang@lbl.gov

Abstract. Large-scale eigenvalue problems arise in a number of DOE applications. This paper provides an overview of the recent development of eigenvalue computation in the context of two SciDAC applications. We emphasize the importance of Krylov subspace methods, and point out its limitations. We discuss the value of alternative approaches that are more amenable to the use of preconditioners, and report the progress on using the multi-level algebraic sub-structuring techniques to speed up eigenvalue calculation. In addition to methods for linear eigenvalue problems, we also examine new approaches to solving two types of non-linear eigenvalue problems arising from SciDAC applications.

1. Introduction

Large-scale eigenvalue problems arise in a number of DOE applications [15, 5, 11, 37]. In the last few years, rapid improvement in computer architecture and substantial advances in algorithmic research have enabled application scientists to tackle eigenvalue problems with tens of millions degrees of freedom. The purpose of this report is to summarize the latest development of large-scale eigenvalue computation research in the context of two SciDAC projects: 1) In the SciDAC accelerator modeling project, eigenvalue computation is required to facilitate the optimal cavity design for the next generation accelerator. 2) In the nanoscience initiative, eigenvalue computation plays an important role in characterizing the many-body electronic interaction for nanoscale materials.

The research accomplishments reported below result from the collaboration between the SciDAC TOPS team, the SciDAC nano-science team at LBNL and a number of SciDAC application centers. It is also important to recognize that our research benefited tremendously from the interaction between the DOE SciDAC projects and a much larger scientific community that contributed to some of the key ideas discussed below.

Although significant progress has been made in large-scale eigenvalue calculation research, a number of challenges still remain. This is particularly true in the area of non-linear eigenvalue problems. We will point out these challenges and discuss future research plans aimed at meeting these challenges.

2. Algorithms for Solving Large-scale Eigenvalue Problems

In this section, we provide an overview of the state-of-the-art algorithms for solving large-scale eigenvalue problems. Instead of providing all the computational details associated with

each algorithm, we will try to highlight the key features of each approach and the underlying principles that distinguish one algorithm from another. We emphasize the importance of Krylov subspace methods, and point out its limitations. We discuss the value of alternative approaches that are more amenable to the use of preconditioners, and report the progress on using multi-level algebraic sub-structuring techniques to speed up eigenvalue calculation. Many algorithms discussed here are developed for solving linear eigenvalue problems of the form,

$$Ax = \lambda x, \quad (1)$$

or

$$Kx = \lambda Mx, \quad (2)$$

where A and K are often symmetric, and M is symmetric positive definite. However, there is an increasing need to tackle non-linear eigenvalue problems in SciDAC applications. We will examine these problems and the current approaches for solving these problems in Section 2.5.

2.1. Krylov Subspace Methods

Krylov subspace methods (KSM) remain the most reliable and effective tools for solving large-scale eigenvalue problems. In a KSM, approximations to the desired eigenpairs of an n by n matrix A are extracted from a k -dimensional *Krylov* subspace

$$\mathcal{K}(v_0, A; k) = \text{span}\{v_0, Av_0, \dots, A^{(k-1)}v_0\},$$

where v_0 is often a randomly chosen starting vector and $k \ll n$. In practice, the retrieval of desired spectral information is accomplished by constructing an orthonormal basis $V_k \in \mathbb{R}^{n \times k}$ of $\mathcal{K}(v_0, A; k)$ and computing eigenvalues and eigenvectors of the k by k projected matrix $H_k = V_k^T A V_k$.

It is well known that dominant eigenvalues well separated from the rest of the spectrum of A converge rapidly in a standard KSM (e.g., the Lanczos or Arnoldi iteration). Furthermore, if the starting vector v_0 contains a linear combination of a few eigenvectors corresponding to the desired eigenvalues, then $\mathcal{K}(v_0, A; k)$ becomes invariant for a small value of k , and the eigenvalues of the projected matrix H_k are indeed the eigenvalues of A .

However, neither of the above two conditions are easily satisfied in real applications. For example, in both the SciDAC accelerator modeling project and the electronic structure calculation problem, the eigenvalues of interest are clustered near zero, and it is generally difficult to identify a starting vector that consists of only a few desired spectral components.

Two types of strategies have been developed to accelerate the convergence of KSM. The implicitly restarting technique originally developed in [25] and implemented in ARPACK [18] is designed to gradually filter out the unwanted spectral components from v_0 by applying the implicitly shifted QR algorithm to the projected matrix H_k . The filter applied in ARPACK is a polynomial filter with roots set near the unwanted eigenvalues. This technique can be extended to include rational filters with poles set near the eigenvalues of interest [27, 33]. The use of implicit restart enables one to extract desired eigenpairs from a Krylov subspace of small dimension, thereby keeping the cost of orthogonalization low.

The second type of techniques for enhancing the convergence of KSM involves transforming the original eigenvalue problem into one that has a more favorable eigenvalue distribution. This technique is particularly helpful when the eigenvalues of interest are near a target shift σ in the interior of the spectrum. In this case, computing the the largest eigenvalues of the shifted inverse $(A - \sigma I)^{-1}$ is often more effective than computing those of A directly. Although rational transformation is the most commonly used spectral transformation, one may also use polynomial transformations [26] when it is prohibitively expensive to factor $A - \sigma I$ into a product of triangular matrices.

Spectral transformation is also useful for solving a generalized eigenvalue problem (2). When both K and M are sparse, it is often more effective to compute the dominant eigenvalues of

$$(K - \sigma M)^{-1} M x = \mu x. \quad (3)$$

Both the implicit restarting technique and spectral transformation prove to be important in solving generalized eigenvalue problems in the SciDAC accelerator modeling project.

In this application, the frequency domain representation of the Maxwell equation (written in terms of the electric field $\mathbf{E}(\mathbf{x})$)

$$\begin{aligned} \nabla \times \left(\frac{1}{\mu} \nabla \times \mathbf{E} \right) &= \frac{\omega^2}{c^2} \epsilon \mathbf{E}, & \mathbf{x} \in \Omega \\ n \times \mathbf{E} &= 0, & \mathbf{x} \in \text{electric boundary} \\ n \times \left(\frac{1}{\mu} \nabla \times \mathbf{E} \right) &= 0, & \mathbf{x} \in \text{magnetic boundary} \end{aligned}$$

is discretized using a hierarchical Nedelec basis [30] to yield a generalized eigenvalue problem in the form of (2). Due to the presence of the curl-curl operator and the particular choice of the finite element basis, the stiffness matrix K contains a null space of a relatively large dimension. The eigenvalues of interest are small but positive. Figure 1 shows the distribution of the eigenvalues at the low end of the spectrum.

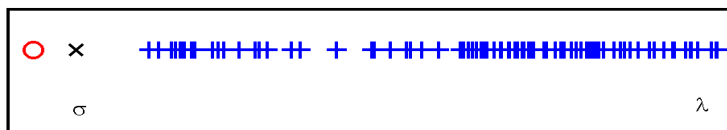


Figure 1. The low end of the spectrum associated with a 5-cell accelerator model. The pluses correspond to eigenvalue of interest. The zero eigenvalues are marked by a circle.

With the target shift σ placed slightly to the right of the origin (the black cross in Figure 1), the spectral transformation of the type (3) maps the eigenvalues of interest to the right end of the spectrum. However, because the transformation also turns the zero eigenvalues into negative eigenvalues (of the transformed problem) with a large magnitude, a simple Lanczos iteration applied to (3) would converge to null vectors of (K, M) first. The use of implicit restart eliminates this undesirable effect by constructing a polynomial filter that repeatedly removes the contribution of the null vectors from the starting vector v_0 .

2.2. The Limitation of Krylov Subspace Methods

The approximate eigenvector z produced by a standard KSM (such as the Arnoldi or Lanczos algorithm) can be expressed by

$$z = p(A)v_0,$$

where $p(\omega)$ is a polynomial that assumes a large value at the desired eigenvalue $\hat{\lambda}$. If z is a good approximation to the desired eigenvector, $p(\lambda)$ should be small for $\lambda \neq \hat{\lambda}$. When A has a large condition number and when $\hat{\lambda}$ is close to other eigenvalues of A , one would need to construct a high degree polynomial in order to produce an accurate approximation to the desired eigenvector. The use of implicit restart reduces the storage requirement and the cost of orthogonalization required to construct such a polynomial. However, it does not fundamentally reduce the degree of the polynomial. Therefore, a standard KSM may not be the most efficient way to solve such a problem due to the large number of matrix-vector multiplications required.

Spectral transformation effectively reduces the degree of $p(\frac{1}{\omega-\sigma})$ required to produce an accurate approximation to the desired eigenvector. However, in order to apply KSM to the transformed problem $(A - \sigma I)^{-1}x = \mu x$, one must solve a sequence of large linear systems of the form

$$(A - \sigma I)w = v. \quad (4)$$

When the size of A becomes extremely large, or when the non-zero fills in its triangular factors are enormous, the cost of sparse matrix factorization can become prohibitively expensive in terms of both computational complexity and memory usage. This problem can be partially resolved by carrying out the sparse matrix operations in parallel on a distributed memory system. In the SciDAC accelerator modeling project, we were able to solve problems with 30 million degrees of freedom and 484 million non-zeros by distributing the sparse matrix computation on 1024 IBM SP processors at NERSC. The total memory usage in this case is 738 GB.

However, when the size of the problem reaches several hundreds of million, this approach is likely to be infeasible due to the current memory limitation and the poor scalability of sparse triangular solves.

Although it is natural to consider replacing a sparse direct solver with a preconditioned iterative solver, there are several difficulties associated with this approach. In particular, one must solve the linear system to a high accuracy to maintain all the desired properties of a Krylov subspace (even though the accuracy required in the approximate eigenpairs is well below machine precision.) This could potentially be very expensive.

2.3. Alternative Methods

The difficulty of introducing a preconditioner into a KSM in a straightforward fashion is fundamentally related to the fact that the eigenvectors of a preconditioned matrix $P^{-1}A$ is generally different from the eigenvectors of A . Thus, building a Krylov subspace in terms of $P^{-1}A$ does not readily provide approximation to the eigenvectors of A directly.

Two alternatives have been pursued in the last few years to overcome this problem. Both require treating an eigenvalue problem as a problem that can be solved without making use of the invariance property of an eigenvector.

2.3.1. Solving an Eigenvalue Problem as an Optimization Problem For problems in which A or K is symmetric (or Hermitian) and the eigenvalues of interest are the smallest (or the largest), one may solve the eigenvalue problem as a constrained optimization problem.

$$\min_{x^T x=1} \rho(x) = x^T A x \quad (5)$$

The use of a preconditioner in this formulation simply amounts to a change of variable.

When k eigenpairs are of interest, one can change the objective function in (5) to $\text{trace}(X^T A X)$, where X is an $n \times k$ matrix subject to the constraint $X^T X = I_k$. With a good preconditioner, a constrained optimization scheme such as the locally optimal block preconditioned conjugate gradient (LOBPCG) algorithm proposed in [14] tends to converge rapidly in the first few iterations. However, as the approximation (\hat{x}) become closer to the desired eigenvector, the reduction in the residual norm $\|A\hat{x} - \rho(\hat{x})\hat{x}\|$ becomes miniscule. The slow down in convergence is mainly due to the fact that $\rho(\hat{x})$ is much more accurate than \hat{x} . Thus as \hat{x} become close to the exact eigenvector, minimizing the $\rho(x)$ is not productive.

A remedy for this type of stagnation is to refine the eigenvector approximation while fixing the approximate eigenvalue that is already very accurate. That is, instead of minimizing the Rayleigh quotient $\rho(x)$, one can solve the following optimization problem instead:

$$\begin{aligned} \min \quad & \|A\hat{x} - \theta\hat{x}\| \\ \text{subject to} \quad & \hat{x} \in V, \|\hat{x}\| = 1, \end{aligned}$$

where θ is a highly accurate approximation to the desired eigenvalue and V is an orthonormal basis for a subspace that contains \hat{x} . This type of eigenvector refinement scheme was originally proposed in [32] in the context of electronic structure calculation. It is later fully analyzed in [12, 29].

The constrained optimization approach has been effective for solving large-scale eigenvalue problems arising from the self-consistent field iteration used in electronic structure calculation. To apply this type of algorithm to the generalized eigenvalue problem arising from the accelerator modeling project, one must deflate the null space associated with the stiffness matrix first. Because deflation essentially turns the mass matrix M into a dense matrix (with a slightly smaller dimension), the cost of matrix-vector multiplication associated with the deflated problem tends to be higher in comparison with the cost of working with the original K and M matrices.

2.3.2. Solving an Eigenvalue Problem as a System of Nonlinear Equations An eigenvalue problem can also be viewed as a set of nonlinear equations

$$Ax = (x^T Ax)x, \quad x^T x = 1. \quad (6)$$

By treating an eigenvalue problem as a nonlinear system, one can exploit the possibility of using Newton's method to compute the desired eigenvalues and eigenvectors. If one has an approximate eigenpair (u, θ) , solving (6) is equivalent to finding a correction pair (z, γ) such that

$$A(u + z) = (\theta + \gamma)(u + z), \quad \text{and} \quad u^T z = 0. \quad (7)$$

This approach leads to the Jacobi-Davidson (JD) [24] algorithm, which is an extension of the Davidson [7] and Olsen [21] algorithms. The method can be described as an inner-outer iteration. Approximations to the desired eigenvalues and eigenvectors are computed in the outer iteration by projecting A into a subspace V . The basis vectors of this subspace are constructed by a sequence of inner iterations that produce an approximate solution to the following projected linear system

$$(I - uu^T)(A - \theta I)(I - uu^T)z = -r, \quad u^T z = 0, \quad (8)$$

where $r = Au - \theta u$.

The use of a preconditioner in JD is mainly aimed at reducing the number of inner iterations required to produce an effective correction vector z . It has been observed that one does not need to solve the correction equation (8) to full accuracy in order to produce a reasonable subspace for eigenvalue approximation. However, the issue of how to control the accuracy of the inner iteration and thus balance the work between the inner and outer iterations requires further investigation. In some cases, the residual norm of the approximate eigenpair can be monitored in each inner iteration by exploiting the relationship between the eigen-residual $\|Au - \theta u\|$ and the linear system residual associated with the correction equation (8) [20, 28]. This relationship allows one to develop a stopping criterion for the inner iteration that is optimal in terms of the total matrix vector multiplications used in the JD algorithm.

Unlike the LOBPCG algorithm, extending JD to a block algorithm appears to be difficult. Hence, eigenpairs are typically computed one at a time in a JD procedure. To avoid recomputing the eigenpair that have converged earlier, one must apply deflation explicitly in the inner iteration.

In the SciDAC accelerator modeling project, an effective preconditioner P is constructed implicitly from a hierarchical finite element basis. The solution to a linear system representing a linear finite element approximation to the differential operator is interpolated onto a subspace associated with a higher order basis. The interpolated solution then serves as the solution to the preconditioning equation. Using this preconditioner in a JD type of algorithm allows us to

compute 8 eigenpairs of the H60VG3 accelerator model in 420 minutes on 128 IBM SP CPUs maintained at NERSC. The H60VG3 model has 93 million degrees of freedom. The total amount of memory used in this calculation is 704 GB, which is significantly less than the 2.5 TB memory that would be required to carry out a shift-invert Lanczos iteration.

2.4. Multi-level and Algebraic Sub-structuring

There has been some progress in developing effective multi-level methods for large-scale eigenvalue calculations since the late 70's [1, 19]. However, most of these algorithms were designed to compute a single eigenpair, and their applicability beyond elliptic problems requires further investigation.

Some recent work [35, 2] has been done on combining a multi-grid solver with the JD, LOBPCG and shift-invert Lanczos algorithms to compute the interior eigenvalues of an elliptic differential operator. Strictly speaking, these approaches are not multi-level eigensolvers because the multi-grid solver used for the inner iteration simply serves as a preconditioner.

Another multi-level technique that has recently gained a lot of attention is the Algebraic Multi-level Sub-structuring (AMLS) method originally developed by Bennighof in the context of vibrational analysis in structure engineering [4]. The method is a multi-level extension of the component mode synthesis (CMS) [6] method developed in the 60's. It is based on a domain decomposition concept, i.e., instead of solving an eigenvalue problem on the entire computational domain (structure), one reduces a large-scale problem to a set of smaller problems defined on several sub-domains (sub-structures). The solutions to these smaller problems are used to construct a subspace from which approximation to the eigenpairs of the original problem is drawn. Because solving problems on each sub-structure requires far less computational power, sub-structuring can lead to a significant reduction in the computational time required in a large-scale engineering simulation for complex structures.

Instead of relying on a geometric partitioning of the computational domain, sub-structuring can be done in a purely algebraic fashion by making use of matrix partitioning techniques such as the nested dissection (ND) algorithm. For a generalized eigenvalue problem (2), a single-level ND partition yields

$$K = \begin{matrix} & \begin{matrix} n_1 & n_2 & n_3 \end{matrix} \\ \begin{matrix} n_1 \\ n_2 \\ n_3 \end{matrix} & \begin{pmatrix} K_{11} & & K_{13} \\ & K_{22} & K_{23} \\ K_{13}^T & K_{23}^T & K_{33} \end{pmatrix} \end{matrix} \quad \text{and} \quad M = \begin{matrix} & \begin{matrix} n_1 & n_2 & n_3 \end{matrix} \\ \begin{matrix} n_1 \\ n_2 \\ n_3 \end{matrix} & \begin{pmatrix} M_{11} & & M_{13} \\ & M_{22} & M_{23} \\ M_{13}^T & M_{23}^T & M_{33} \end{pmatrix} \end{matrix}, \quad (9)$$

where the labels n_1 , n_2 and n_3 denote the dimensions of each sub-matrix block. The pencils (K_{11}, M_{11}) and (K_{22}, M_{22}) now define two algebraic sub-structures that are connected by the third block rows and columns of K and M which we will refer to as the *interface* block. In AMLS, a congruence transformation is applied to (K, M) to turn K into a block diagonal matrix. The transformation results in the modification of the third block rows and columns of K and M . In particular, it turns K_{33} into \hat{K}_{33} and M_{33} into \hat{M}_{33} . The approximation to the eigenpairs of the original problem are then extracted from the subspace

$$S = \begin{matrix} & \begin{matrix} k_1 & k_2 & n_3 \end{matrix} \\ \begin{matrix} n_1 \\ n_2 \\ n_3 \end{matrix} & \begin{pmatrix} S_1 & & \\ & S_2 & \\ & & S_3 \end{pmatrix} \end{matrix} \quad (10)$$

where S_1 and S_2 consist of k_1 and k_2 selected eigenvectors of (K_{11}, M_{11}) and (K_{22}, M_{22}) respectively, and S_3 consists of a subset of the eigenvectors of $(\hat{K}_{33}, \hat{M}_{33})$.

One of the key decisions one has to make in AMLS is the number of eigenpairs to compute on each sub-structure (and to include in S_i , $i = 1, 2, 3$). These eigenpairs are referred to as sub-structure modes. Selecting too many modes increases the size of the final projected problem, and consequently the cost of the overall computation. Limiting the the selection to too few modes can result in a lack of accuracy in the final approximation to the original problem.

An error analysis for a single-level algebraic sub-structuring is carried out in [34]. The analysis measures the the contribution of each sub-structure mode to the final approximation of the desired eigenvector. An error bound for the approximation to the smallest eigenvalue is established. The error bound is expressed in terms of the cumulative contribution of the sub-structure modes that are excluded from the approximation eigenvector.

It is observed in [34] that, for many problems (including the ones arising from the SciDAC accelerator modeling), most of the sub-structure modes do not make a significant contribution to the approximation of the desired eigenvector. Hence they do not need to be computed or included in S . A heuristic is developed in [34] for choosing sub-structuring modes based on the desired accuracy of the approximate eigenvector. This heuristic is justified independently in [8].

The AMLS algorithm has been used successfully in structure analysis applications. The timing results reported in [13, 17] indicate that AMLS is significantly faster than conventional Lanczos-based approaches [18, 10]. However, our recent performance evaluation of the algorithm indicate that the performance of the algorithm depends on a number of factors such as the number of partitioning levels, the number of modes selected from each substructure, the choice of method for solving the final projected problem [9]. Because performing a congruence transformation is almost as expensive as factoring $K - \sigma M$, AMLS is not the most efficient method to use when only a few eigenpairs are needed. However, because AMLS does not carry out basis orthogonalization or solve any triangular systems, it is attractive when a large number of eigenvalues are needed.

A sequential version of the AMLS algorithm has been developed and used in the SciDAC accelerator project. The implementation has the capability to deflate the null space in the sub-structuring calculation.

2.5. Non-linear Eigenvalue Problems

Although significant progress has been made in solving large-scale linear eigenvalue problems of the form (1) and (2), there is an increasing need for developing efficient algorithms for solving non-linear eigenvalue problems.

A nonlinear eigenvalue problem arises in the SciDAC accelerator modeling project when one takes into account the external coupling of waveguides with an open cavity. In this case, one must determine the wave function x that satisfies

$$Kx + i\sqrt{k^2 - k_c^2}Wx = k^2Mx, \quad (11)$$

where k is an unknown cavity resonant frequency, K is the stiffness matrix, M is the mass matrix, W is a known damping matrix and k_c is a fixed reference frequency. If we let $\lambda = \sqrt{k^2 - k_c^2}$, then (11) can be formulated as a quadratic eigenvalue problem (QEP)

$$\lambda^2Mx - i\lambda Wx + (k_c^2M - K)x = 0. \quad (12)$$

The standard approach for solving (12) is to linearize it first. The linearization would produce a generalized (linear) eigenvalue problem twice as large in dimension. The larger linear eigenvalue can then be solved by using, for example, a KSM method.

In a recent paper [3], Bai and Su proposed a second order Arnoldi (SOAR) algorithm for solving a quadratic eigenvalue problem without linearization. The new algorithm constructs a *second order Krylov subspace* in terms of $A = iM^{-1}W$ and $B = M^{-1}K - k_c^2I$ [3]. Because

the algorithm projects the quadratic eigenvalue problem directly into the second order Krylov subspace, it has the advantage of preserving all the essential structure and properties of the QEP.

SOAR has been used successfully in computing the cavity modes of a 9-cell superconducting model (which is part of the international linear collider project) coupled with one external input. The dimension of the discretized model is $n = 3.2 \times 10^6$ in this case. Using 768 IBM SP processors, the SOAR computation takes less than an hour. This is much faster than applying an Arnoldi iteration to the linearized problem. The computed eigenvalues are more accurate than those obtained in the Arnoldi iteration also.

A more difficult nonlinear eigenvalue problem arises in electronic structure calculation which is part of the SciDAC nanoscience project. One of the major problems in electronic structure calculation is to minimize the Kohn-Sham (KS) total energy functional associated with an atomistic system with respect to its electron orbitals. Once discretized, the total energy can be expressed by

$$E_{total}(X) = E_{kinetic}(X) + E_{ion}(X) + E_H(X) + E_{XC}(X),$$

where $X = (x_1, x_2, \dots, x_k)$ represent a set of electron wavefunctions corresponding to k occupied states, and $E_{kinetic}$, E_{ion} , E_H and E_{XC} represent contributions from the kinetic energy, the ionic, Hartree and exchange-correlation potentials.

The total energy must be minimized under the orthonormality constraint $X^T X = I_k$. The Karush-Kuhn-Tucker (KKT) condition for the constrained optimization problem yields the following nonlinear eigenvalue problem

$$H(X)X = X\Lambda_k, \quad X^T X = I_k, \quad (13)$$

where the $H(X)$ is the Kohn-Sham Hamiltonian that depends on X .

Currently, the most widely used algorithm for solving this energy minimization problem is the so-called *Self Consistent Field* (SCF) iteration algorithm. The algorithm is essentially a fixed point iterative scheme applied to the Kohn-Sham equation (13). Given an initial guess of X , say $X^{(0)}$, one forms the discrete Hamiltonian $H(X^{(0)})$ and computes eigenvectors ($X^{(1)}$) associated with the k smallest eigenvalues of $H(X_0)$. If the difference between $H(X^{(i)})$ and $H(X^{(i+1)})$ remains large, the eigenvectors associated with the k smallest eigenvalues of $H(X^{(i+1)})$ are computed, and this process continues until the difference between $H(X^{(i)})$ and $H(X^{(i+1)})$ becomes negligibly small. When SCF converges, $X^{(i)}$ consists of a set of wave functions that are self-consistent with respect to the KS equation (13).

Although SCF is widely used, its convergence property is still poorly understood. It is well known that SCF iterations often fail to converge. Several techniques such as the Direct Inversion of Iterative Subspace (DIIS) [23] and charge mixing [16] have been developed to prevent the SCF iteration from diverging. However, these techniques are not well understood either. Although they help in many cases, they can still fail.

A constrained minimization algorithm has recently been developed to minimize the total energy directly [36]. This method is an extension of the LOBPCG algorithm for solving a linear eigenvalue problem. It differs from some of the earlier attempts to minimize the total energy [22, 31] in the way the search direction and step length are chosen. The new algorithm essentially reduces a large constrained minimization problem to a sequence of much smaller constrained minimization problem. Such a reduction allows one to exploit more sophisticated optimization algorithms for finding solutions to the nonlinear eigenvalue problem. Furthermore, it eliminates the need to solve a sequence of large-scale linear eigenvalue problem in the SCF iteration.

Preliminary results reported in [36] show that the direct constrained minimization is more effective than SCF on a simple test problem. However, more extensive tests are required to assess the convergence behavior of the algorithm for larger atomistic systems. The key to make

this algorithm more efficient and robust is to identify the best scheme for solving the reduced minimization problem. This remains to be a research problem.

3. Concluding Remarks

As part of the SciDAC Tera-scale PDE Simulation (TOPS) and nano-science projects, we have been working with other SciDAC application centers on developing and deploying efficient and reliable algorithms for solving large-scale eigenvalue problems in SciDAC applications. While Krylov subspace methods (KSM) remain the most reliable and effective tool for solving linear eigenvalue problems, it has some limitations. In particular, it is not easy to make the best use of preconditioners in KSM. This limitation requires us to explore other alternatives. Although preliminary studies indicate that multi-level algebraic sub-structuring techniques can be extremely helpful in speeding up the calculation of a large number of eigenpairs, further studies are required to improve the accuracy of this approach. Significant progress has been made on the development of efficient and structure preserving algorithms for solving quadratic eigenvalue problems. However, much more needs to be done in the development of reliable and efficient algorithms for solving non-linear eigenvalue problems arising from electronic structure calculation.

4. Acknowledgement

The author would like to thank Weiguo Gao, Xiaoye Li, Parry Husbands, Esmond Ng, Juan Meza Lin-wang Wang (LBNL), Lie-quan Lee (SLAC) and Zhaojun Bai (UC-Davis) for helpful discussions.

5. References

- [1] S. F. McCormick A. Brandt and J. Ruge. Multigrid methods for differential eigenproblems. *SIAM J. Sci. Stat. Comput.*, 4(2):244–260, 1992.
- [2] P. Arbenz, U. L. Hetmaniuk, R. B. Lehoucq, and R. S. Tuminaro. A comparison of eigensolvers for large-scale 3D modal analysis using AMG-preconditioned iterative methods. Technical report 2005-0282J, Sandia National Laboratories, 2005.
- [3] Z. Bai and Y. Su. SOAR: A second-order Arnoldi method for the solution of the quadratic eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 26(3):640–659, 2005.
- [4] J. K. Bennighof and C. K. Kim. An adaptive multi-level substructuring method for efficient modeling of complex structures. In *Proceedings of the AIAA 33rd SDM Conference*, Dallas, Texas, 1992.
- [5] E.A. Burroughs, L.A. Romero, R. B. Lehoucq, and A.J. Salinger. Linear stability of flow in a differentially heated cavity via large-scale eigenvalue calculations. *International J. Num. Meth. Heat and Fluid Flows*, 14(6):803–822, 2004.
- [6] R. R. Craig and M. C. C. Bampton. Coupling of substructures for dynamic analysis. *AIAA Journal*, 6:1313–1319, 1968.
- [7] E. R. Davidson. The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real symmetric matrices. *Journal of Computational Physics*, 17:87–94, 1975.
- [8] K. Elssel and H. Voss. An a priori bound for automated multi-level substructuring. Report 81, Arbeitsbereich Mathematik, TU Hamburg-Harburg, 2004.
- [9] W. Gao, X. S. Li, C. Yang, and Z. Bai. Implementation and evaluation of an algebraic sub-structuring method for sparse eigenvalue problems. Technical report, Lawrence Berkeley National Lab, Berkeley, CA, 2005.
- [10] R. G. Grimes, J. G. Lewis, and H. D. Simon. A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems. *SIAM Journal on Matrix Analysis and Applications*, 15(1):228–272, January 1994.
- [11] M. Head-Gordon. Quantum chemistry and molecular processes. *J. Phys. Chem*, 100:13213, 1996.
- [12] Z. Jia. Refined iterative algorithm based on Arnoldi’s process for large unsymmetric eigenproblems. *Linear Algebra and Its Applications*, 259:1–23, 1997.
- [13] M. F. Kaplan. *Implementation of Automated Multilevel Substructuring for Frequency Response Analysis of Structures*. PhD thesis, University of Texas at Austin, Austin, TX, December 2001.
- [14] A. V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM J. Sci. Comput.*, 23:517–541, 2001.

- [15] K. Ko, N. Folwell, L. Ge, A. Guetz, V. Ivanov, L. Lee, Z. Li, I. Malik, W. Mi, C. Ng, and M. Wolf. Electromagnetic systems simulation - from simulation to fabrication. SciDAC report, Stanford Linear Accelerator Center, Menlo Park, CA, 2003.
- [16] G. Kresse and J. Furthmüller. Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set. *Physical Review B*, 54(16):11169 – 11185, 1996.
- [17] A. Kropp and D. Heiserer. Efficient broadband vibro-acoustic analysis of passenger car bodies using an FE-based component mode synthesis approach. *J. Comput. Acoustics*, 11:139–157, 2003.
- [18] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK USERS GUIDE: Solution of Large Scale Eigenvalue Problems by Implicitly Restarted Arnoldi Methods*. SIAM, Philadelphia, PA, 1998.
- [19] S. F. McCormick. Multi-level adaptive methods for elliptic eigenproblems: A two-level convergence theory. *SIAM Journal on Numerical Analysis*, 31(6):1731–1745, 1994.
- [20] Y. Notay. Combination of Jacobi-Davidson and conjugate gradient for the partial symmetric eigenproblem. *Numer. Lin. Alg. Appl.*, 9:21–44, 2002.
- [21] J. Olsen, P. Jorgensen, and J. Simons. Passing the one-billion limit in full configuration-interaction (fci) calculations. *Chemical Physics Letters*, 169:463–472, 1990.
- [22] M. C. Payne, M. P. Teter, D. C. Allen, T. A. Arias, and J. D. Joannopoulos. Iterative minimization techniques for *ab initio* total energy calculation: molecular dynamics and conjugate gradients. *Reviews of Modern Physics*, 64(4):1045–1097, 1992.
- [23] P. Pulay. Convergence acceleration of iterative sequences: The case of SCF iteration. *Chemical Physics Letters*, 73(2):393–398, 1980.
- [24] G. L. G. Sleijpen and H. A. Van der Vorst. A Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 17(2):401–425, April 1996.
- [25] D. C. Sorensen. Implicit application of polynomial filters in a k-step Arnoldi method. *SIAM Journal on Matrix Analysis and Applications*, 13(1):357–385, January 1992.
- [26] D. C. Sorensen and C. Yang. Accelerating the Lanczos algorithm via polynomial spectral transformations. Technical report TR 97-29, Department of Computational & Applied Mathematics, Rice University, Houston, TX, 1997.
- [27] D. C. Sorensen and C. Yang. A truncated RQ-iteration for large scale eigenvalue calculations. *SIAM J. Matrix Anal. Appl.*, 19(4):1045–1073, 1998.
- [28] A. Stathopoulos. Nearly optimal preconditioned methods for hermitian eigenproblems under limited memory. part i: Seeking one eigenvalue. Technical report Submitted, College of William and Mary, Williamsburg, VA, 2005.
- [29] G. W. Stewart. *Matrix Algorithms, Vol II: Eigensystems*. SIAM, Philadelphia, PA, 2001.
- [30] Din-Kow Sun, Jin-Fa Lee, and Zoltan Cendes. Construction of nearly orthogonal Nedelec bases for rapid convergence with multilevel preconditioned solvers. *SIAM J. on Sci. Comput.*, 23(4):1053–1076, 2001.
- [31] T. Van Voorhis and M. Head-Gordon. A geometric approach to direct minimization. *Molecular Physics*, 100(11):1713–1721, 2002.
- [32] D. M. Wood and A. Zunger. A new method for diagonalising large matrices. *J. Phys. A: Math. Gen.*, 18:1343–1359, 1985.
- [33] C. Yang. Convergence analysis of an inexact truncated RQ-iteration. *Electronic Transaction on Numerical Analysis*, 7(4):40–55, 1998.
- [34] C. Yang, W. Gao, Z. Bai, X. Li, L. Lee, P. Husbands, and E. Ng. An algebraic sub-structuring method for large-scale eigenvalue calculation. Technical report LBNL-55050, Lawrence Berkeley National Lab, 2004, To appear in SIAM J. Sci. Comput.
- [35] C. Yang and W. F. Lawkins. A multi-level method for large-scale eigenvalue calculation. Technical report ORNL TM-2000/43, Computer Science & Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, 2000.
- [36] C. Yang, J. Meza, and L. W. Wang. A constrained optimization algorithm for total energy minimization in electronic structure calculation. LBNL Report 57434, Lawrence Berkeley National Laboratory, 2005.
- [37] C. Yang, B. W. Peyton, D. W. Noid, B. G. Sumpter, and R. E. Tuzun. Large-scale normal coordinate analysis for molecular structures. *SIAM J. Sci. Comp.*, 23(2):563–582, 2001.