FINAL PROGRESS REPORT - JANUARY 16, 2014

Grant number	**DE-FG 08ER 25841**
Institution	University of Minnesota
Principal Invesitagor	Yousef Saad

Robust parallel iterative solvers for linear and least-squares problems

Campus address:	*4-192 Keller Hall*
	200 Union st. SE
	Minneapolis, MN 55455
Phone number	*612-624-7804*
Fax	*612-625-0572*
e-mail	*saad@cs.umn.edu*
URL	*www.cs.umn.edu/∼saad*

1 Objectives and highlighted accomplishments

The primary goal of this project is to study and develop robust iterative methods for solving linear systems of equations and least squares systems. This is a collaborative project with Masha Sosonkina from Ames lab., IA. The focus of the Minnesota team is on algorithms development, robustness issues, and on tests and validation of the methods on realistic problems. The Ames team focuses on parallel implementations, and issues related to scalability on massively parallel architectures, and on software integration. For the Minnesota team the actual project *started effectively on September 2008.*

In the following we list the highlights of accomplishments for the duration of the project. Section 2 presents details on selected topics.

1. The project begun with an investigation on how to practically update a preconditioner obtained from an ILU-type factorization, when the coefficient matrix changes. This work has been published in [2].

2. Along with Masha Sosonkina and (later) Eugene Vecharynsky, we investigated strategies to improve robustness in parallel preconditioners in a specific case of a PDE with discontinuous coefficients. This work was published in [15].

3. We explored ways to adapt standard preconditioners for solving linear systems arising from the Helmholtz equation. These are often difficult linear systems to solve by iterative methods. This is discussed in the article [10].

4. We have also worked on purely theoretical issues related to the analysis of Krylov subspace methods for linear systems, see [1].

5. Jointly with Scott MacLachlan (Rutgers), and Daniel Osei-Kuffuor (student), we developed an effective strategy for performing ILU factorizations for the case when the matrix is highly indefinite. The strategy uses shifting in some optimal way. The method was extended to the solution of Helmholtz equations by using complex shifts, yielding very good results in many cases. This work appeared in [8].

6. In an effort with Ruipeng Li (student) we addressed the difficult problem of preconditioning sparse systems of equations on GPUs [5]. This resulted in a collaborative article which appeared in two papers [6, 12].

7. A by-product of the above work is a software package consisting of an iterative solver library for GPUs based on CUDA. This was made publicly available, see `http://www-users.cs.umn.edu/~saad/software/`. It was the first such library that offers complete iterative solvers for GPUs.

8. In the final two years of the project, we considered another form of ILU which blends coarsening techniques from Multigrid with algebraic multilevel methods. A paper was submitted and has been conditionally accepted for publication [9].

9. We have released a new version on our parallel solver - called pARMS [new version is version 3]. As part of this we have tested the code in complex settings - including the solution of Maxwell and Helmholtz equations and for a problem of crystal growth [along with Jeff Derby from Chemical Engineering at the U. of M.]

10. As an application of polynomial preconditioning we considered the problem of evaluating $f(A)v$ which arises in statistical sampling. This is collaborative work with Miha Anitescu from Argonne and was published in [3].

11. As an application to the methods we developed, we tackled the problem of computing the diagonal of the inverse of a matrix. This arises in statistical applications as well as in many applications in physics. We explored probing methods [14] as well as domain-decomposition type methods [13].

12. A collaboration with researchers from Toulouse, France, considered the important problem of computing the Schur complement in a domain-decomposition approach- One paper was published [4].

13. In the final year of the project, we explored new ways of preconditioning linear systems, based on low-rank approximations. Part of this work overlapped with the new grant from NSF on the same topic. The ideas were conceived during the last year of this grant and were completed under the new (NSF) grant. One article was published [7].

2 Research contributions

Updating preconditioners. This work was a collaboration with Jean-Paul Chehab (Univ. Amiens, France) and Caterina Calgaro (Univ. of Lilles, France). It examined a number of techniques for computing incremental ILU factorizations, i.e., ways to update factorizations when the coefficient matrix varies slowly. This problem arises in many important applications. For example, a common situation in computational fluid dynamics, is when the equations change only slightly, possibly in some parts of the physical domain. In such situations it is wastful to recompute entirely any LU or ILU factorizations computed for the previous coefficient matrix. Techniques based on approximate inverses as well as alternating techniques for updating the factors L and U of the factorization, were proposed. The work is published in [2].

Adapting partitioners for discontinuous problems. When solving sparse linear systems of equations in parallel one must partition the problem carefully at the outset to achieve good convergence of the iterative process. Standard graph partitioners aim at balancing the number of unknowns and reducing communication volume, based on the nonzero pattern of the matrix. As is well-known this objective is not robust for realistic practical problems. This is the case for example, when solving discretized Partial Differential Equation (PDE) with discontinuous coefficients or when dealing with complex multi-physics phenomena. It is therefore desirable that the partitioner take into account information beyond the adjacency graph of the matrix. The technical report [11] shows how the flexibility of hypergraph partitioning can be exploited to develop heuristics for incorporating numerical information in partitioning tasks. A modification of a standard hypergraph model is proposed along with several weighting schemes to build partitionings which will more likely lead to good convergence of the process. In a set of numerical experiments we show comparisons with standard approaches,

on simple two- and three-dimensional elliptic problems with discontinuous coefficients on rectangular meshes. We have revisited and continued this work in the last year of the project with Eugene Vecharynski. Here we used a criterion based on the so-called CBS constant to find good ways to partition the graph. This work was recently accepted for publication [15].

Robust preconditioners. Linear systems which originate from the simulation of wave propagation phenomena can be very difficult to solve by iterative methods. These systems are typically complex valued and they tend to be highly indefinite, which renders standard ILU-based preconditioners ineffective. The paper [10] presents a study of ways to enhance standard preconditioners by altering the diagonal by imaginary shifts. It is already known that modifying the diagonal entries during the incomplete factorization process, by adding to it purely imaginary values can improve the quality of the preconditioner in a substantial way. We proposed to use simple algebraic heuristics to perform the shifting and tested these techniques with the ARMS and ILUT preconditioners. Comparisons made with applications stemming from the diffraction of an acoustic wave incident on a bounded obstacle (governed by the Helmholtz Wave Equation), show that this is a quite effective method.

Toward the end of the project, we worked on two topics related to robust preconditioners. First, we developed a preconditioning method which blends ideas from coarsening techniques in Multigrid with algebraic multilevel methods. A paper was submitted and has recently been conditionally accepted for publication [9]. Second, we explored new ways of preconditioning linear systems, based on low-rank approximations. One article was published [7]. In both cases the work overlapped with the new grant from NSF on the same topic. The ideas were conceived during the last year of this grant and were completed under the new (NSF) grant. In both papers we acknowledged the two grants.

Convergence study of Krylov methods. Krylov subspace methods are strongly related to polynomial spaces and their convergence analysis can often be naturally derived from approximation theory. Convergence can then be analyzed from different angles, using either a discrete min-max approach or a discrete least-square approach. The paper [1] examines the relationships between these two approaches. Specifically, it shows that for normal matrices, the Karush-Kuhn-Tucker (KKT) optimality conditions derived from a convex maximization problem related to the second viewpoint are identical to the properties that characterize the polynomial of best approximation on a finite set of points. Therefore, these two approaches are mathematically equivalent. This is joint work with Mohammed Bellalij (Univ. Valenciennes, France), and Hassane Sadok (Univ. of Calais, France).

Sparse matrix computations on GPUs. Upon returning from a summer internship with a major oil company, Ruipeng Li, a student supported by this project, gained an interest in the solution of sparse linear systems on manycore computers and GPUs. The issue of developing 'robust' and parallel iterative solvers is a very difficult one for such platforms. This is because the performance of GPUs can be disastrously poor for irregular computations. For example an NVIDIA Tesla C1060 can deliver close to 1 Teraflop in single precision for certain types of computations, but will yield only about 20 GFLOPS for sparse matrix techniques. If a standard ILU preconditioner is used, performance can be unacceptable, e.g., delivering a few tens to 100 MFLOPS, or 4 to 5 orders of magnitude below the peak rate. We considered several alternatives. First we looked only at the performance of matrix-vector products and found it helpful to look back at techniques used in the 1990s for vector computers. In particular, a format which mixes diagonal storage with general sparse storage generally gives the excellent results. In terms of preconditioners, we considered block diagonal techniques but found that using polynomial preconditioners with high degree (up to 100 sometimes) gave the best results. The technical report [5] describes our first contributions on this topic. It also serves as a documentation to the library which we developed (see Section 3 for details). Later we collaborated with specialists on oil recovery, on the use of these techniques for reservoir simulation. This particular work appeared in two papers [6, 12].

Our work on GPUs for sparse linear systems indicated that additional advances may be necessary on the hardware side before performance of sparse linear solvers can match that achieved for dense solvers. To quote our recent tech report on this issue (see of the paper conclusion) "Current GPUs

provide a much lower performance advantage for irregular (sparse) computations than they can for more regular (dense) computations..." We decided to limit our exposure to this line of work for this reason. However, we developed and made available a software library for solving linear systems by iterative method with the language CUDA, see Section 3 below.

Application: computing $f(A)v$. The report [3] considers the problem of computing $f(A)v$ for a certain function f and a vector v. The method uses explicitly least-squares polynomials. The work was in collaboration with Mihai Anitescu from Argonne National Lab and Jie Chen, a student. The main application considered corresponds to the situation when $f(t) = \sqrt{t}$ and A is a sparse, symmetric positive-definite matrix which arises in sampling from a Gaussian process distribution. The covariance matrix of the distribution is defined by using a covariance function with compact support, at a very large number of sites located on a regular or irregular grid. Note that the class of polynomials used for this application is of the same type as that used for preconditioning in the context of GPUs, and discussed above. Due to the application of the paper to an important problem in data mining, the student working on this specific topic was supported by this grant and another (by NSF) related to the application. Both grants are acknowledged.

Application: Diagonal of matrix inverse. The computation of some entries of the inverse of a matrix is important for several relevant applications in practice. Such applications range from Density functional theory in electronic structure calculations, to statistics, and Dynamic Mean Field Thoeory in highly correlated systems. The paper [14] considers a method for performing these calculations. The idea works for the case when the inverse exhibits a decay of its entries away from the diagonal which is a common practical situation. Two methods were studied, one presented is based on 'probing'. Iterative method are used for solving linear systems corresponding to probing vectors which determine the pattern of the inverse. Due to the application of the paper to an important problem in Dynamical Mean Field Theory (DMFT), the post-doc working on this specific topic was supported by this grant and another (by NSF) related to the application. Both grants are acknowledged.

3 Contributions to software

Parallel Algebraic Recursive Multilevel Solver (pARMS). Daniel Osei-Kuffuor (student) has updated and tested thoroughly a new version of pARMS which was released in the second year of the project:

`http://www-users.cs.umn.edu/~saad/software/pARMS/index.html`

A major effort was put to develop an interface to PETSc, which was deployed with the version 3 that is now posted. PETSc is a widely used libray developed by Department of Energy researchers and it is important to make available an interface to allow calling pARMS from pERSC. Some effort has been put into an interface with Trilinos. The interface has been implemented but not released.

The code has been tested on a number of applications including linear systems arising form Helmholtz and Maxwell equations, and from a challenging application related to crystal growth (in a collaboration with Jeff Derby from Chemical Engineering and Materials Science at the university of Minnesota). The goal was to disseminate a solver that is fairly robust and that has been thoroughly tested on key applications. Exposure to these applications also enabled us to develop more effective and user-friendly software.

CUDA_ITSOL As was mentioned above, one of the by products of our work on the use of GPUs for iterative solution techniques, was the development of a software package. Our CUDA Iterative Solver package called CUDA_ITSOL, was made publicly available in May 2011. This is a package for performing various sparse matrix operations and, more importantly, for solving sparse linear systems of equations. It has been developed under the CUDA environment primarily by Ruipeng Li [PhD Student]. The technical report [5] serves as documentation for the package, which can be found at the following site:

`http://www-users.cs.umn.edu/~saad/software/`

4 Contributions to human resources

Three students and two post-docs were supported by this grant. Daniel Osei-Kuffuor (PhD student) was supported from this grant from its start. Daniel completed his PhD in Sept. 2011 and is now working at Lawrence Livermore National Lab. Jie Chen (PhD student) was supported from this grant for a short period (about a semester). He defended his thesis in June 2011, and has since worked for Argonne National Lab. Ruipeng Li was partially supported by this year (last year of project). He is due to complete his PhD within in 2014-2015. Eugene Vecharynski (Post-doc) was supported by this for one year. He is now working at the Lawrence Berkeley Lab. Finally, Jok Tang was supported part-time from this project during the academic year 2009-2010. He worked with Daniel Osei-Kuffuor on new ideas for our solver pARMS as well as on the application of robust solvers for computing the diagonal of the inverse of a matrix. He is currently working for a numerical software company in the Netherlands.

5 Publications

Technical reports and articles published under this grant were cited in Section 1 and are listed in the references. All papers are available online from

`http://www.cs.umn.edu/~saad/reports.html`

References

[1] M. Bellalij, Y. Saad, and H. Sadok. Analysis of some Krylov subspace methods for normal matrices via approximation theory and convex optimization. *Electronic Transactions on Numerical Analysis*, 33:17–30, 2008.

[2] Caterina Calgaro, Jean-Paul Chehab, and Yousef Saad. Incremental incomplete lu factorizations with applications. *Numerical Linear Algebra with Applications*, 17(5):811–837, 2010.

[3] Jie Chen, Mihai Anitescu, and Yousef Saad. Computing f(A)b via least squares polynomial approximations. *SIAM Journal on Scientific Computing*, 33(1):195–222, 2011.

[4] L. Giraud, A. Haidar, and Y. Saad. Sparse approximations of the Schur complement for parallel algebraic hybrid solvers in 3D. *Numerical Mathematics: Theory, Methods and Applications*, 3(3):276–294, 2010.

[5] R. Li and Y. Saad. GPU-accelerated preconditioned iterative linear solvers. Technical Report umsi-2010-112, Minnesota Supercomputer Institute, University of Minnesota, Minneapolis, MN, 2010.

[6] Ruipeng Li, Hector Klie, Hari Sudan, and Yousef Saad. Towards realistic reservoir simulations on many-core platforms. *SPE Journal*, pages 1–23, 2010.

[7] Ruipeng Li and Yousef Saad. Divide and conquer low-rank preconditioning techniques. *SIAM Journal on Scientific Computing*, -:–, 2013. To Appear.

[8] S. MacLachlan, D. Osei-Kuffuor, and Yousef Saad. Modification and compensation strategies for threshold-based incomplete factorizations. *SIAM Journal on Scientific Computing*, 34(1):A48–A75, 2012.

[9] Daniel Osei-Kuffuor, Ruipend Li, and Yousef Saad. Matrix reordering using multilevel graph coarsening for ilu preconditioning. Technical Report ys-2013-4, Dept. Computer Science and Engineering, University of Minnesota, Minneapolis, MN, 2013.

[10] Daniel Osei-Kuffuor and Yousef Saad. Preconditioning Helmholtz linear systems. *Appl. Numer. Math.*, 60:420–431, April 2010.

[11] Masha Sosonkina and Yousef Saad. Hypergraph partitioning for sparse linear systems: A case study with a simple discontinuous PDE. Technical Report umsi-2009-29, Minnesota Supercomputer Institute, University of Minnesota, Minneapolis, MN, 2009. Submitted.

[12] Hari Sudan, Hector Klie, Ruipeng Li, and Yousef Saad. High performance manycore solvers for reservoir simulation. In *ECMOR 12th European Conference on the Mathematics of Oil Recovery, 6-9 September 2010, Oxford, UK*, 2010.

[13] J. Tang and Y. Saad. Domain-decomposition-type methods for computing the diagonal of a matrix inverse. *SIAM Journal on Scientific Computing*, 33(5):2823–2847, 2011. Accepted Apr. 2011.

[14] Jok M. Tang and Yousef Saad. A probing method for computing the diagonal of a matrix inverse. *Numerical Linear Algebra with Applications*, 19(3):485–501, 2011.

[15] Eugene Vecharynski, Yousef Saad, and Masha Sosonkina. Graph partitioning with matrix coefficients for symmetric positive definite linear systems. *SIAM Journal on Scientific Computing*, -, 2014. To appear.