DEPARTMENT OF COMPUTER SCIENCE
COLLEGE OF SCIENCES
OLD DOMINION UNIVERSITY
NORFOLK, VA 23529

WAVEFRONT MINIMIZATION OF SPARSE MATRICES

By
Dr. Alex Pothen, Principal Investigator
Department of Computer Science

FINAL REPORT
For the period ending June 14, 1998

February 1999

DISCLAIMER

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Final Report
DOE grant DE-FG05-94ER25216
PI: Alex Pothen

1 Summary of Results

1.1 Wavefront minimization of Sparse Matrices

Gary Kumfert and I have improved the quality and run time of two ordering algorithms for
minimizing the wavefront and envelope size of sparse matrices and graphs. These algorithms
compute orderings for irregular data structures (e.g., unstructured meshes) that reduce the
number of cache misses on modern workstation architectures.

The first algorithm is an improved implementation of a combinatorial algorithm due to
Sloan. We described an algorithm whose running time is bounded by the number of edges
(nonzeros) in the sparse graph (matrix). Consequently the new implementation is an order
of magnitude faster for large problems. We also improved the quality of the algorithm by
identifying two classes of problems that have differing asymptotic behaviors of their wave-
front parameters, and by choosing parameters appropriately for each class. This algorithm
currently provides good quality with low execution times, and is the best combinatorial
algorithm for the problem at this time.

We have also improved an algebraic (spectral) algorithm for reducing the wavefront by
using a post-processing phase involving the Sloan algorithm. The hybrid algorithm further
reduces the wavefronts over the spectral and Sloan algorithms but requires a factor of six
greater running time. Examples show that these improved orderings lead to frontal solvers
that are faster by more than an order of magnitude on advanced architectures.

The C and C++ software that we have developed is available with three interfaces:
Matlab, PETSc (the Portable, Extensible Toolkit for Scientific Computing), and as a stand-
alone code, where it forms part of an ordering and partitioning package called SPINDLE.

An important application of this work is in computing incomplete factorization precondi-
tioners for iterative solvers for linear systems of equations. Another is in organizing irregular
data structures to reduce cache misses on a computer with hierarchical memory. These
applications are being currently investigated. Our code is currently used by the Applied Nu-
merical Aerodynamics group at NASA Ames (Aftosmis), the Advanced Structural Integrity
program at NASA Langley (James), and in multigrid strategies for viscous flow solvers on
unstructured anisotropic meshes at ICASE (Mavriplis). Our work has led Drs. John Reid
and Jennifer Scott at Rutherford Appleton Lab (U.K.) to replace the current Sloan code in

the Harwell software library (MC40) by new routine that incorporates our improvements to the Sloan algorithm and code (MC60).

1.2 Parallel Sparse Solvers

We have completed the implementation of a parallel solver for sparse, symmetric indefinite systems for distributed memory computers such as the IBM SP-2. The indefiniteness requires us to incorporate block pivoting (2 by 2 blocks) in the algorithm, thus demanding dynamic, parallel data structures. This is the first reported parallel solver for the indefinite problem.

Several features are included in the parallel algorithm to enhance parallel performance. The algorithm employs a new pivoting strategy that reduces the serial bottleneck of earlier pivot selection methods. The pivot strategy also reduces the chance of pivot failures by exhaustively searching all available elements to be pivot candidates. The pivot threshold is set loosely to increase parallel performance by reducing pivot failures. We employ a few steps of iterative refinement to obtain full accuracy in the solution.

The algorithm makes use of the multifrontal algorithm to perform the computation. This algorithm has regular dense matrix kernels in the numerical factorization, and also limits the communication necessary in the algorithm. BLAS-3 operations are used to make the numerical factorization efficient.

The code is written in C, and uses the MPI message passing library for portability. We have tested the indefinite code on a collection of problems from structural analysis (Boeing-Harwell, NASA, some structural analysis vendors), from CFD (Boeing-Harwell, Matrix Market), and linear programming problems (Argonne National Labs). We have solved all the problems tested to high accuracy relative to the conditioning of the problems. We are collaborating with Prof. Hafiz Atassi from Notre Dame on solving Helmholtz problems on an NSF-supported multidisciplinary challenges project.

1.3 Object Oriented Design of Sparse Solvers

Direct methods for solving systems of linear equations employ sophisticated combinatorial and algebraic algorithms that contribute to software complexity, and hence it is natural to consider object-oriented design (OOD) in this context. We have continued to create software for solving sparse systems of linear equations by direct methods employing OOD.

OOD manages complexity by means of decomposition and abstraction. We decompose our software into two main types of objects: structural objects corresponding to data structures, and algorithmic objects corresponding to algorithms. This design decouples data

structures from algorithms, permitting a user to experiment with different algorithms and different data structures, and if necessary develop new algorithms and data structures. We have implemented a family of minimum degree ordering algorithms using this design paradigm, and thus provided a laboratory for users to experiment with recent enhancements and algorithmic variants. We have also implemented a direct solver for symmetric positive definite and indefinite problems. The increased flexibility in OOD could come at the cost of some loss in efficiency. We have made careful trade-offs in our software to achieve the benefits of OOD without sacrificing efficiency. The running times of our C++ code for the symmetric positive definite solver compare quite favorably with existing Fortran 77 codes. We have reported our results at the SCITOOLS98 (Oslo) and ISCOPE98 (Santa Fe) conferences this year. This is the first object-oriented implementation of sparse direct methods that we know of.

We are continuing to create parallel and out-of-core solvers using OOD. Our current serial code is available with an interface to the PETSc toolkit from Argonne National Labs, and as a stand-alone code, and is called OBLIO.

1.4 Fast Algorithms for Robust Incomplete Factor Preconditioners

Fast computation of robust preconditioners is a priority for solving large systems of equations on unstructured grids and in other applications. We have developed new algorithms and software that can compute incomplete factorization preconditioners for high level fill in time proportional to the number of floating point operations and memory accesses.

We have developed a structure theory based on paths in the adjacency graph of the matrix to predict where zero elements become nonzeros in incomplete factorization (fill elements). A level function is used in incomplete factorization to control the number of fill elements, and we relate the level of fill to lengths of appropriately defined paths in the adjacency graph. This result permits us to search in the neighborhood of a vertex in the graph to predict all fill elements associated with that vertex. We have designed two variants of these algorithms and have proved that they have a smaller running time complexity than currently used algorithms for computing incomplete factorizations. The more efficient algorithms make use of the concept of transitive reduction of directed graphs (symmetric problems) and symmetric reduction of directed graphs (unsymmetric problems) in order to search for paths in smaller graphs. Our implementation in C shows that the new algorithms are faster than implementations available earlier.

We are investigating parallel implementations of incomplete factorizations. Our serial

code is available with an interface to the PETSc toolkit from Argonne; it adds functionality in the area of preconditioners to PETSc.

1.5   Publications

S. T. Barnard, A. Pothen, and H. D. Simon, A spectral algorithm for envelope reduction of sparse matrices, *Numerical Linear Algebra with Applications*, 2, pp. 317–334, 1995.

I. Stoica, H. Abdel-Wahab, and A. Pothen, A microeconomic scheduler for parallel computers, In Load-balancing and Job Scheduling for Parallel Computers, D. G. Feitelson and L. Rudolph (eds.), *Lecture Notes in Computer Science*, Vol. 949, Springer-Verlag, pp. 200–218, 1995.

I. Stoica and A. Pothen, A robust and flexible microeconomic scheduler for parallel computers, Proceedings of the Third International High Performance Computing Conference, Thiruvananthapuram, India, IEEE Press, 1997. (12 pages)

A. George and A. Pothen, An analysis of spectral envelope-reduction via quadratic assignment problems, SIAM Journal of Matrix Analysis and its Applications, 18, pp. 706–732, 1997.

A. Pothen, Graph partitioning algorithms with applications to scientific computing, Parallel Numerical Algorithms, D. E. Keyes, A. H. Sameh and V. Venkatakrishnan (eds.), Kluwer Academic Press, pp. 323–368, 1996.

G. Kumfert and A. Pothen, Two improved algorithms for reducing the envelope size and wavefront of sparse matrices, BIT, 18, pp. 559–590, 1997.

D. Hysom and A. Pothen, Symmetric reduction redux: incomplete factorizations, in preparation, 1998.

Gary Kumfert and Alex Pothen, An object-oriented collection of minimum degree algorithms: design, implementation, and experiences, In Computing in Object-oriented Parallel Environments, Lecture Notes in Computer Science 1505, D. Caromel et al (eds.), pp. 95–106, Springer Verlag, 1998. (Proceedings of the Second International Symposium on Computing in Object-oriented Parallel Environments, Santa Fe, 1998.)

Florin Dobrian, Gary Kumfert, and Alex Pothen, Object-oriented design of a sparse symmetric solver, In Computing in Object-oriented Parallel Environments, Lecture Notes

in Computer Science 1505, D. Caromel et al (eds.), pp. 207–214, Springer Verlag, 1998. (Proceedings of the Second International Symposium on Computing in Object-oriented Parallel Environments, Santa Fe, 1998.)

Florin Dobrian, Gary Kumfert, and Alex Pothen, Object-oriented design for sparse direct solvers, In Modern Software Tools in Scientific Computing, H. P. Langtangen et al. (eds.), Birkhauser Press, 40 pp., 1999 (to appear).

1.6 Human and Educational Impact

Dr. Yogin Campbell, who joined our group after receiving his PhD from the University of Florida in 1995, worked on the symmetric indefinite solver for one year. He completed his postdoctoral research associateship, and has joined AT& T. Dr. Lois C. McInnes worked on a primitive variable Euler formulation of the external aerodynamics problem in three dimensions as described above. She was our postdoctoral associate, and has been active in the Portable, Extensible Toolkit for Scientific Computation (PETSc) effort at Argonne National Lab. She has now accepted a permanent position at Argonne as a Computer Scientist.

Gary Kumfert is a Graduate Research Assistant working on the graph partitioning and variable ordering problem. He will complete his PhD in Summer '99, and currently is recruiting at several DOE labs. His thesis work resulted in an object-oriented software package for ordering and partitioning called SPINDLE. Florin Dobrian is working on the object-oriented design of parallel and out-of-core solvers. David Hysom is working on fast algorithms for computing robust preconditioners. Both are expected to complete their PhD theses by Summer 2000.

The PI has engaged in several outreach activities to the applied computational community. In the past three years, he has co-organized the several conferences and workshops, organized sessions at them centered around the themes of this project, or been invited to participate to share their expertise. Among these are the Ninth International Conference on Domain Decomposition Methods, Workshop on Parallel Unstructured Grid Computations at Argonne, ICASE-LaRC Industry Roundtable, and the Petaflops workshop. In addition, he has delivered nine invited departmental seminars at universities, two invited seminars at national laboratories, one invited seminar in industry, and six talks at national meetings, (SIAM conferences, the Copper Mountain conference, and IMA Workshops).