

Progress Report
DOE Grant No. DE-FG02-91ER25095
June 1992 to Feb 1993

Sparse Matrix Algorithms on Distributed Memory Multiprocessors

Alex Pothén

We have been able to make significant progress in the creation of algorithms and software for large-scale sparse matrix computations on advanced distributed-memory parallel machines during the past year. Together with my students and colleagues, I am continuing to pursue several research issues on these topics.

1. **Large-scale linear systems.** In this area, we focused on three problems: algorithms for solving sparse triangular systems on highly parallel machines like the Connection Machine CM-2, the computation of spectral nested dissection orderings for solving sparse systems of equations, and algorithms and software for factoring sparse matrices on distributed-memory multiprocessors.

1.1. **Highly parallel triangular solution.** On massively parallel machines such as the Connection Machine, a bottleneck in the parallel solution of linear systems is the triangular solution part, since $O(n^2)$ floating point operations are performed on $O(n^2)$ elements. In the situation when the system involves multiple right-hand side vectors, a *product-form inverse* (PFI) approach can be used to significantly improve the parallelism by computing the solution by means of a sequence of matrix-vector multiplications. By minimizing the number of steps of matrix-vector multiplications, we obtain an algorithm for solving the triangular system efficiently on massively parallel machines.

In [14], we had designed a fast linear-time algorithm to reduce the number of matrix-vector multiplication steps when the input matrix is symmetric positive definite. This was accomplished by minimizing the number of multiplication steps over all symmetric permutations of the given triangular matrix such that the permuted matrix is also triangular. This algorithm was faster by more than a hundred fold on a collection of problems over a previous algorithm designed for nonsymmetric or symmetric indefinite matrices [1]; it has an even greater edge in terms of auxiliary storage. This program is now being used in a software package called the Sparse Matrix Manipulation System (SMMS) created by Professor Alvarado (Wisconsin).

With Barry Peyton (Oak Ridge National Labs) and a graduate student Xiaoqing Yuan [11, 12], I was able to generalize the above problem to reduce the number of matrix-vector multiplications even further. Given a Cholesky factor L of a symmetric positive-definite matrix A , we were able to minimize the number of matrix-vector multiplication steps over all symmetric permutations such that the structure of the symmetric *filled matrix* $L + L^T$ does not change under the permutation. The reordering corresponding to the permutation has to be applied to the original matrix A *before* the Cholesky factorization. This work has necessitated the study of chordless paths and the structure of the vertex separators in chordal

graphs. The paper [11] describes a high-level scheme to solve this problem together with a simple linear-time algorithm that implements the scheme. A more efficient implementation of the above scheme is possible by making use of a compact data structure called the clique tree; this will be described in [12].

An important numerical issue is the stability of the PFI approach to triangular solution. We have identified a number which measures the growth of elements in the product-form inverse such that when this 'growth factor' is small, the method is normwise both forward and backward stable [8]. The growth factor is guaranteed to be small when the matrix is well-conditioned, and can also be bounded independent of the condition in many circumstances.

We have surveyed this work in [2], as part of a publication that discusses the state of the art in sparse matrix computations. I talked about this work at the 1992 annual meeting of the International Linear Algebra Society (ILAS) at Lisbon (Portugal) in August, and will also talk on this topic at the SIAM Parallel Processing Conference at Norfolk in March. Barry Peyton will be presenting this work at the ILAS 1993 annual meeting at Pensacola, FL in a session on sparse matrix algorithms that I have organized.

1.2. Spectral nested dissection orderings. We have developed an algebraic approach for ordering sparse matrices for parallel factorization called *spectral nested dissection*. The ideas here can also be used for domain decomposition of irregular domains and for mapping the computations in a parallel algorithm onto processors. This is joint work with my Ph.D. student Lie Wang and Horst Simon (NASA Ames) [3].

In this approach, we use the adjacency graph of the matrix to form a matrix called the Laplacian matrix, and then use information about a particular eigenvector to compute a separator in the graph. This approach is then recursively employed to compute spectral nested dissection orderings. Our results on very large problems (with tens of thousands of unknowns) show that this approach is very successful in computing orderings that have better parallelism than the currently available methods such as minimum-degree and earlier variants of nested dissection. Currently we are working on an efficient implementation of the spectral nested dissection algorithm.

The challenge here is to compute the required eigenvector of the large, sparse matrices fast. A new 'multi-level' approach is being developed in which we repeatedly form smaller matrices from the given matrix, compute the eigenvector of the smallest matrix in the sequence by the Lanczos algorithm, and then use Rayleigh quotient iteration to compute the eigenvector of the original matrix. Many important theoretical and practical issues need to be resolved before a fast algorithm may be obtained to compute spectral orderings. Once that is done, we will focus on the parallel computation of the orderings.

Professor Bojan Mohar of Ljubljana of Slovenia (formerly Yugoslavia) and I have used the spectral approach to design and analyze the performance of an algorithm for reducing the envelope size of a sparse matrix. This problem is important in several structural engineering codes, where envelope methods are used to solve large systems of equations. We showed that the Laplacian matrix could be used to greatly reduce the size of the envelope, and thereby the storage and arithmetic work required for the solution. This work [10] is being written up.

I talked about this work at the IMACS conference on partial differential equations at

Rutgers in May and also at Supercomputing '92 at Minneapolis in November. I also plan to present this work at a Workshop on Algebraic Graph Theory organized by the International Center for Mathematical Sciences at Edinburgh in July.

1.3. Parallel Multifrontal factorization. The multifrontal method is known to be an efficient method for computing the Cholesky factorization of sparse matrices on vector and parallel computational environments. My Ph.D. student Chunguang Sun (now a postdoc at the Advanced Computing Research Institute, Cornell University) and I investigated several issues in producing an efficient implementation of the multifrontal method on the iPSC/2 and iPSC/860 hypercubes [16]. We used a data structure called the clique tree (which we had previously studied—see [9, 15]) to organize the computation using efficient dense matrix kernels, and designed a *proportional mapping* algorithm to map computational subtasks to the processors. We reported the first set of results on parallel execution times for irregular sparse systems for the hypercube machines, and efficiencies were comparable to the results obtained for the model regular grid problem. During the past year, work was performed on our code to make it high quality software for parallel multifrontal factorization on the iPSC/860 hypercube [17]. This software is now available for public use and we have received several requests for it.

2. Structure of orthogonal factors. A direct method for the solution of least-squares problems requires the computation of the orthogonal factors of the given sparse matrix. To do so efficiently, we require data structures that store only the nonzeros in the factors before the numerical factorization is computed. Last year Hare, Johnson, Olesky, and van den Driessche [7] showed how the structures of the orthogonal factors could be predicted. They proved that given the position of a nonzero in the predicted factor, there exists a matrix whose factor has a nonzero in that position. In [13], I extended this work to show that the structures predicted were the best possible: the orthogonal factors of almost all matrices have nonzeros in *every* nonzero position in the predicted data structures. Hence if a matrix has elements that are reasonably 'random', then the structures of its factors are exactly equal to the predicted structures. We also developed algorithms for efficiently computing the predicted data structures.

In current work, we are studying the problem of predicting the structure of the Householder matrix, an important data structure for representing the orthogonal factor. This is a first step towards the design of efficient algorithms for orthogonal factorization with pivoting for rank-deficient and ill-conditioned problems.

I described this work at a Workshop on computational and graph-theoretic aspects of linear algebra at the University of Essen, Germany in July.

3. Sparse Bases for the Range Space and the Null Space. A central problem in the solution of large-scale numerical optimization problems is computing a sparse basis for the null space of a large, sparse, underdetermined matrix. A theoretical study of the sparse null space basis problem was made in [5], and then algorithms for computing null space bases were designed and implemented in [6]. A fundamental open problem associated with computing a sparse null space basis is identifying a condition on the zero-nonzero structure which would guarantee the linear independence of the computed null vectors.

With Professors Richard Brualdi (Wisconsin) and Shmuel Friedland (Illinois) [4], I have been able to characterize the structure of sparsest bases of dense underdetermined matrices in terms of a condition on the zero-nonzero structure of the basis. This problem turned out to be surprisingly difficult, and we had to employ techniques from multilinear algebra to solve the problem. We are currently trying to extend these results to the sparse case using some results from algebraic geometry. A solution to this problem will make it possible for us to begin work on computing null space bases in parallel.

REFERENCES

- [1] F. Alvarado and R. S. Schreiber. Optimal parallel solution of sparse triangular systems. *SIAM J. Sci. Stat. Comput.*, to appear, 1993.
- [2] F. L. Alvarado, A. Pothén, and R. S. Schreiber. Highly parallel sparse triangular solution. In J. A. George, J. W. H. Liu, and J. R. Gilbert, editors, *Sparse Matrix Algorithms: Graph Theory Issues and Algorithms*. Springer Verlag, 1992. (an IMA volume in Mathematics and its Applications.)
- [3] S. Barnard, A. Pothén, H. D. Simon, and L. Wang. Towards a fast implementation of spectral nested dissection. In *Proceedings of Supercomputing '92*, pages 42–51, Los Alamitos, CA, 1992. IEEE Computer Society Press.
- [4] R. A. Brualdi, S. Friedland, and A. Pothén. Sparse bases, elementary vectors, and nonzero minors of compound matrices. Technical Report CS-92-05, Computer Science, Penn State, April 1992. Submitted to *SIAM Journal on Matrix Analysis and its Applications* (under revision).
- [5] T. F. Coleman and A. Pothén. The null space problem I: Complexity. *SIAM J. Alg. & Disc. Meth.*, 7:527–537, 1986.
- [6] T. F. Coleman and A. Pothén. The null space problem II: Algorithms. *SIAM J. Alg. & Disc. Meth.*, 8:544–563, 1987.
- [7] D. R. Hare, C. R. Johnson, D. D. Olesky, and P. van den Driessche. Sparsity analysis for sparse orthogonal factorization. *SIAM Journal on Matrix Analysis and its Applications*, 1993. To appear.
- [8] N. J. Higham and A. Pothén. The stability of the partitioned inverse approach to sparse triangular solution. *SIAM Journal on Scientific Computing*, 1993. To appear.
- [9] J. G. Lewis, B. W. Peyton, and A. Pothén. A fast algorithm for reordering sparse matrices for parallel factorization. *SIAM J. Sci. Stat. Comput.*, 10:1146–1173, 1989. (Special section on Sparse Matrix Algorithms for Supercomputers).
- [10] B. Mohar and A. Pothén. A spectral approach to envelope reduction of sparse matrices. In preparation., 1993.
- [11] B. W. Peyton, A. Pothén, and X. Yuan. Partitioning a chordal graph into transitively closed subgraphs for parallel sparse triangular solution. Technical Report CS-92-55, Computer Science, University of Waterloo, Oct 1992. Submitted to *Linear Algebra and its Applications*.
- [12] B. W. Peyton, A. Pothén, and X. Yuan. A clique tree algorithm for partitioning a chordal graph into transitively closed subgraphs. In preparation, 1993.
- [13] A. Pothén. Predicting the structure of sparse orthogonal factors. *Linear Algebra and its Applications*, 1993. To appear.
- [14] A. Pothén and F. Alvarado. A fast reordering algorithm for parallel sparse triangular solution. *SIAM J. Sci. Stat. Comput.*, 13:645–653, March 1992.
- [15] A. Pothén and C. Sun. Compact clique tree data structures in sparse matrix factorizations. In *Large-Scale Numerical Optimization, Proceedings in Applied Mathematics*, pages 180–204. Society for Industrial and Applied Mathematics, 1990.
- [16] A. Pothén and C. Sun. A mapping algorithm for parallel sparse Cholesky factorization. *SIAM Journal on Scientific Computing*, 1993. To appear.
- [17] C. Sun. Efficient parallel solutions of large sparse SPD systems on distributed-memory multiprocessors. Technical Report TR 102, Theory Center, Cornell University, 1992. Submitted to *SIAM Journal on Scientific and Statistical Computing*.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

END

**DATE
FILMED**

5 / 10 / 93

