A Nearly-Linear Time Algorithm for Approximately Solving Linear Systems in a Symmetric M-Matrix*

Samuel I. Daitch[1], Daniel A. Spielman[2],

[1] Yale University, Department of Computer Science
PO Box 208285, New Haven, CT 06520-8285, USA
`samuel.daitch@yale.edu`
[2] Yale University, Program in Applied Mathematics
and Department of Computer Science
PO Box 208285, New Haven, CT 06520-8285, USA
`spielman@cs.yale.edu`

Abstract. We present an algorithm for solving a linear system in a symmetric M-matrix. In particular, for $n \times n$ symmetric M-matrix M, we show how to find a diagonal matrix D such that DMD is diagonally-dominant. To compute D, the algorithm must solve $\mathcal{O}(\log n)$ linear systems in diagonally-dominant matrices. If we solve these diagonally-dominant systems approximately using the Spielman-Teng nearly-linear time solver [1], then we obtain an algorithm for approximately solving linear systems in symmetric M-matrices, for which the expected running time is also nearly-linear.

Keywords. M-matrix, diagonally-dominant matrix, linear system solver, iterative algorithm, randomized algorithm, network flow, gain graph

1 Extended Abstract

1.1 Introduction

A **symmetric M-matrix** may be defined as a positive-definite matrix with non-positive off-diagonal entries. Matrices of this sort have many notable properties. For example, for every symmetric M-matrix M, there is a diagonal matrix D such that DMD is diagonally-dominant. In this paper, we present an iterative algorithm for computing this diagonal matrix. In particular, given an $n \times n$ symmetric M-matric M with m non-zero entries, we show how to find a diagonal matrix D that makes DMD diagonally dominant, by solving $\mathcal{O}(\log n)$ linear systems in diagonally-dominant matrices with at most m non-zeros.

* This material is based upon work supported by the National Science Foundation under Grant Nos. CCF-0707522 and CCF-0634957. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Dagstuhl Seminar Proceedings 09061
Combinatorial Scientific Computing
http://drops.dagstuhl.de/opus/volltexte/2009/2080

Our algorithm relies on another property of symmetric M-matrices, namely that all such matrices have a **width-2** factorization [2]. This means that if M is a symmetric M-matrix, then it can be expressed as $M = AA^T$ for some matrix A that has at most 2 non-zero entries per column. Our algorithm assumes that we are given a width-2 factorization of the symmetric M-matrix.

Now, suppose that we wish to find the solution $\boldsymbol{x}$ for $M\boldsymbol{x} = \boldsymbol{b}$, where M is a symmetric M-matrix. Note that $\boldsymbol{x} = D\boldsymbol{x}'$, where $\boldsymbol{x}'$ is the solution to $(DMD)\boldsymbol{x}' = D\boldsymbol{b}$. Thus, by using our algorithm to find a diagonal matrix D that makes DMD diagonally-dominant, and then solving $(DMD)\boldsymbol{x}' = D\boldsymbol{b}$ for $\boldsymbol{x}'$, we are able to solve $M\boldsymbol{x} = \boldsymbol{b}$ by solving $\mathcal{O}(\log n)$ linear systems in diagonally-dominant matrices.

If we use the Spielman-Teng low-stretch support graph preconditioning algorithm [1] to approximately solve the linear systems in diagonally-dominant matrices in nearly-linear time, then our algorithm yields a nearly-linear time algorithm for solving linear systems in symmetric M-matrices:

Theorem 1. *For symmetric M-matrix M with m nonzeros, where a width-2 factorization of M is given, our algorithm solves $M\boldsymbol{x} = \boldsymbol{b}$ to within relative error ϵ in expected time $\mathcal{O}\left(m \log^{\mathcal{O}(1)} m \log \frac{\kappa}{\epsilon}\right)$, where κ is the condition number of M. The error is measured in the matrix norm $\|\boldsymbol{x}\|_M = \sqrt{\boldsymbol{x}^T M \boldsymbol{x}}$.*

1.2 Algorithm Overview

The key to finding a diagonal matrix D that makes DMD diagonally-dominant is the observation that a random diagonal matrix D will already make a constant fraction of the diagonal entries of DMD dominate their rows. (We say that a diagonal entry dominates its row if it is greater in absolute value than the sum of the absolute values of all the other entries in the row.)

Lemma 1. *For symmetric M-matrix M with diagonal entries m_{ii}, construct a random diagonal matrix D by independently choosing the ith diagonal entry uniformly at random from the interval $(0, m_{ii}^{-1/2})$. Then with positive constant probability, a constant fraction of the diagonal entries of DMD dominate their rows.*

Once we have DMD for which a constant fraction of the diagonal entries dominate their rows, we give a procedure to find a new diagonal matrix D' such that the number of diagonal entries of $D'MD'$ that do not dominate their rows is fewer than in DMD by at least a constant factor. This procedure requires the solution of a constant number of linear systems in diagonally-dominant matrices. By repeating this procedure $\mathcal{O}(\log n)$ times, we obtain a DMD that is entirely diagonally-dominant, and we require altogether the solution of $\mathcal{O}(\log n)$ linear systems in diagonally-dominant matrices, as claimed.

Let us briefly describe this procedure for decreasing the number of rows not dominated by their diagonal entries. We partition the rows (and columns) of M into two groups, writing $M = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}$ and $D = \begin{pmatrix} D_1 & \\ & D_2 \end{pmatrix}$ such that the diagonal entries in the upper part of the matrix DMD dominate their rows, while those in the lower part do not. (Without loss of generality, we may assume that the rows and colums of M are permuted such that the diagonal entries that dominate their rows come first.) We will take advantage of the fact that we are already able to solve linear systems in M_{11} by solving equivalent linear systems in the diagonally-dominant matrix $D_1 M_{11} D_1$.

We wish to find a D' such that all the diagonal entries in the upper part of the matrix $D'MD'$ still dominate their rows, and additionally a constant fraction of the diagonal entries in the lower part of $D'MD'$ also dominate their rows. It turns out that we can obtain such a D' by applying Lemma 1 to the Schur complement of M, namely $M_S = M_{22} - M_{21} M_{11}^{-1} M_{12}$.

To apply Lemma 1 to M_S, we need to compute the diagonal entries of M_S. We may accomplish this using the width-2 factorization $M = AA^T$, with the rows of $A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}$ partitioned into the same two groups as the rows of M. It so happens that the diagonal entries of M_S may be obtained by projecting the rows of A_2 onto the null space of A_1. To compute these projections exactly would be too computationally intensive. However, we may compute these values approximately using Johnson-Lindenstrauss [3] style random projections, by solving only $\mathcal{O}(1)$ linear systems in M_{11}. As noted above, using the current value of D_1 we are already able to reduce solving linear systems in M_{11} to solving systems in the diagonally-dominant matrix $D_1 M_{11} D_1$.

For proofs and further details of the algorithm, please refer to [4].

1.3 Related Work

An interesting example of where symmetric M-matrices occur is in solving generalized network flow problems. These are network flow problems where each edge is assigned a gain (or loss) factor that defines the ratio between the flow into and out of the edge. When solving generalized network flow problems using interior-point methods, each iteration of the interior-point method reduces to solving a linear system in a symmetric M-matrix. (Compare the case of standard network flow problems where all gain factors are 1, and the interior-point iterations reduce to solving a linear system in a diagonally-dominant matrix.) In [4], we show that using our M-matrix algorithm as part of an interior-point method yields new faster approximation algorithms for the generalized max flow and min-cost flow problems, in the lossy case where no edge has more flowing out than flowing in.

References

1. Spielman, D.A., Teng, S.H.: Nearly-linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. <http://www.arxiv.org/abs/cs.NA/0607105> (2006)
2. Boman, E.G., Chen, D., Parekh, O., Toledo, S.: On factor width and symmetric h-matrices. Linear Algebra and its Applications **405** (2005) 239–248

3. Johnson, W., Lindenstrauss, J.: Extensions of lipschitz mappings into a hilbert space. Contemp. Math. **26** (1984) 189–206
4. Daitch, S.I., Spielman, D.A.: Faster approximate lossy generalized flow via interior point algorithms. In: Proceedings of the 40th ACM Symposium on the Theory of Computing. (2008) 451–460 <http://arxiv.org/abs/0803.0988>.