# Direct linear time solvers for sparse matrices

Shivkumar Chandrasekaran* Ming Gu†

October 30, 2003

## 1 Introduction

In the last couple of years it has been realized that Gaussian elimination for sparse matrices arising from certain elliptic PDEs can be done in $O(n \log(1/\epsilon))$ flops, where $n$ is the number of unknowns and $\epsilon$ is a user-specified tolerance [Chandrasekaran and Gu]. The resulting solver will also be backward stable with an error of $O(\epsilon)$. The techniques used to achieve this speedup has some commonality with ideas from domain decomposition, multi-grid, incomplete factorizations, and other areas. However, the algorithms have also shown speedups on *random sparse* positive-definite matrices, making it clear that some novel ideas are at play.

It has also been known for sometime that one can use the ideas of the Fast Multi-pole Method (FMM) of Greengard and Rokhlin, to do Gaussian elimination of dense matrices that arise from the discretization of two-dimensional integral equations in $O(n^{1.5})$ flops, and from three-dimensional integral equations in $O(n^2)$ flops [Rokhlin and Starr]. This greatly improves the competitiveness of integral equation methods.

However, what has not been realized widely is that these same complexities can also be achieved quite simply by converting the dense matrices to sparse matrices and using standard sparse matrix solvers! This can be viewed as an extension of the **column stretching** techniques for sparse matrices [Grcar]. In brief this is how it is done. We first observe that the FMM is a technique to do fast matrix-vector multiplication, and that the algorithm consists of a set of **linear recursions** on a tree. We then label the intermediate values on the tree with new variables. Then the problem of solving the system of linear equations can be viewed as the problem of finding the intermediate variables **and** the unknown solution on the FMM tree. Since the FMM recursions are linear this gives rise to a sparse system of equations whose incidence graph is the associated FMM tree! This enables us to use the extensively developed techniques of sparse matrix solvers for dense integral equation methods. We will report on experiments with this method for electromagnetic scattering calculations.

Even more surprising, is that this idea in reverse is what is needed to speed-up Gaussian elimination for sparse matrices. More specifically, the fill-in that occurs during Gaussian elimination can be viewed as the discretization of an integral operator (associated with the Green's function of the PDE). There is nothing new about this idea, and it is well-known, especially in the domain-

*shiv@ece.ucsb.edu
†mgu@math.berkeley.edu

decomposition literature, and the preconditioning literature. However, till recently there has been no way to exploit the *FMM structure* in the fill-in efficiently.

In this talk we will outline our approach for exploiting the structure of the fill-in during Gaussian elimination to obtain a linear-time algorithm. At this point it is worth remarking that first forming the fill-in and then computing its FMM structure will not be sufficient to improve the complexity. In other words, we must compute the structure of the fill-in *without* forming the fill-in first! This is a non-trivial task.

## 2 Low-rank off-diagonal blocks

The FMM is based on a simple idea: Green's functions of many PDEs are smooth away from the diagonal singularity. It follows that the associated discrete matrices will have low-rank off-diagonal blocks. This is the property of the fill-in that we also wish to exploit. However, fill-ins are the result of a large number of calculations, and it is not clear how to construct a low-rank representation of a certain block without first forming the block.

Consider a simple model problem: a block tri-diagonal matrix $(\,C_{i-1} \quad A_i \quad B_i\,)$, with each of the $A_i$'s, $B_i$'s and $C_i$'s being tri-diagonal matrices themselves. (Such structures arise from the discretization of elliptic PDEs on rectangular grids using 9-point stencils.) Now the key equation during Gaussian elimination (without pivoting) of this sparse matrix is the recursion for the Schur complements:

$$S_{i+1} = A_{i+1} - C_i S_i^{-1} B_i,$$

with $S_0 = A_0$. It is clear that $S_i$ is dense for $i > 0$.

Now, we observe that each of the $A_i$'s, $B_i$'s and $C_i$'s has rank-1 off-diagonal blocks (since they are tri-diagonal). Now, we use an algebraic representation of matrices, called **hierarchically semi-separable** (HSS) representation. This representation uses the minimal number of parameters to represent a matrix which has low-rank off-diagonal blocks. Furthermore, it can be shown that the standard matrix operations in this representation, preserve the representation, and are linear in the length of the representation. For example, if we multiply two matrices using their HSS representation, we can find the HSS representation of their product in optimal (linear) time. Similarly, we can find the HSS representation of the inverse, $LU$ factorization, $QR$ factorization, etc., in optimal time. It follows that since the HSS representation of $A_i$, $B_i$ and $C_i$ can be found in linear time, we can find the HSS representation of $S_i$ in linear time, *without* forming them explicitly! Now, if the rank of the diagonal blocks of the $S_i$ remains small as $i$ gets bigger, then the resulting $LU$ factorization will take only linear time.

The reason the rank of the off-diagonal blocks of $S_i$ remains small is that the $S_i$ are related to the off-diagonal blocks of the Green's function, which is smooth. We can also prove a similar result for certain Toeplitz positive-definite sparse matrices which do not necessarily arise from PDEs.

So the phenomenon is provably true for elliptic PDEs, and seems to hold for a much larger class of sparse matrices.

We will also discuss applications to pseudo-spectral discretizations of PDEs.