High-Order Nyström Schemes for Efficient 3-D Capacitance Extraction

Sharad Kapur David E. Long
Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974

Abstract

Integral equation based approaches are popular for extracting the capacitance of integrated circuit structures. Typically, first-order collocation or Galerkin methods are used. The resulting dense system of equations is efficiently solved by combining matrix sparsification with an iterative solver. While the speed-up over direct factorization is substantial, the first-order methods still lead to large systems even for simple problems. In this paper we introduce a high-order Nyström scheme. For the same level of discretization, the high-order schemes can be an order of magnitude more accurate than the first-order approaches at the same computational cost. As a consequence, we obtain the same level of accuracy with a much smaller matrix.

1 Introduction

With decreasing feature sizes and increasing frequencies, accurate and efficient capacitance extraction has become critical for design. In recent years, capacitance extractors based on integral equations [3, 7] have become popular. Integral formulations have many advantages over finite-difference or finite element schemes, including good conditioning, reduction in dimensionality and ease in dealing with layered dielectrics [15]. While discretizing an integral equation leads to a linear system involving a dense matrix, modern matrix compression schemes combined with iterative solvers have made working with this dense matrix efficient. Previous extractors have discretized the integral equation using a first-order collocation [3, 7] or Galerkin [2] method. In these methods, the charge density is assumed to be piecewise constant. With this crude approximation, getting accurate answers mandates large discretizations even for simple problems.

In this paper we introduce an alternative discretization technique, the Nyström [5] method, which reduces the integral equation to a finite system by replacing the integral with a high-order quadrature. The primary advantage of this approach is that extrememly fast convergence can be achieved at a very small cost of constructing quadratures. Nyström approaches have been sucessfully used for one-dimensional and two-dimensional problems in the numerical analysis community [1, 4, 13]. However, there have not been any methods developed to handle three-dimensional extraction problems. In our approach, the matrix arising from the Nyström discretization is represented using IES^3 [3], a scheme for matrix compression based on the singular value decomposition (SVD). The compressed representation is then used to compute matrix-vector products during an iterative solve. For matrices of the same size, the high-order Nyström approach is essentially the same speed as the earlier methods based on first-order collocation. However, since the high-order Nyström scheme is rapidly convergent, the accuracy obtained is dramatically higher than before. As a consequence, in order to obtain the same accuracy, the high-order approach requires much smaller matrices.

2 Formulation of the Problem

The capacitance extraction problem can be reduced to solving the first-kind integral equation

$$\phi(r) = \int_R G(r, r')\sigma(r')\, dR, \tag{1}$$

where ϕ is the potential, σ is the surface charge density, R ranges over the surfaces of conductors, and G is the Green's function. In free-space, the Green's function $G(r, r')$ is simply $1/(4\pi\epsilon_0|r-r'|)$. For layered media, the Green's function can be precomputed and tabulated. To solve the integral equation numerically, we must first discretize the region R and reduce (1) to a matrix equation. The typical approach is to use first-order collocation or Galerkin methods [2, 3, 6].

2.1 First-order methods

In a Galerkin approach, the integral equation (1) is reduced to a matrix problem by projection onto the space spanned by an orthonormal set of basis functions $\{f_1, f_2, \ldots, f_n\}$. The approximate solution

$$\tilde{\sigma}(r) = \sum_{i=1}^{n} c_i f_i(r) \tag{2}$$

is obtained by solving the system of equations

$$\sum_{j=1}^{n} a_{ij} c_j = \phi_i, \qquad i = 1, \ldots, n \tag{3}$$

where

$$\phi_i = \int_R \phi(r) f_i(r)\, dr \tag{4}$$

and

$$a_{ij} = \int_R \int_{R'} G(r, r') f_i(r) f_j(r')\, dr\, dr'. \tag{5}$$

The quality of the approximation $\tilde{\sigma} \approx \sigma$ is dependent on the size of the differences

$$|G(r, r') - \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} f_i(r) f_j(r')|. \tag{6}$$

In most implementations, the functions f_i are taken to be piecewise constant (a first-order method). Since a true Galerkin scheme requires accurate evaluation of the double integrals of the singular kernels of equation (5), first-order collocation schemes are more popular. In collocation schemes [3, 6], the domain R is subdivided into regions $\{T_1, T_2, \ldots, T_n\}$, a collocation point r_j is chosen in each region, and σ is assumed to be piecewise constant over each region. Then (1) reduces to the following set of equations:

$$\phi(r_i) = \sum_{j=1}^{n} \left(\int_{T_j} G(r_i, r')\, dr' \right) \sigma(r_j). \tag{7}$$

In both first-order collocation and Galerkin methods, the asymptotic error is at least $1/n$ (in a Galerkin method the constant associated with the error is smaller). The situation is even worse when the boundary has edges or corners with strong solution singularities. Hence, accurate solutions require extremely fine discretizations. A more efficient alternative is to use a high-order method.

2.2 High-order Nyström methods

The Nyström method represents the functions σ and ϕ in the integral equation

$$\phi(r) = \int_R G(r, r') \sigma(r')\, dr' \tag{8}$$

by their values at selected points $\{r_1, r_2, \ldots, r_n\}$ on R and replaces the integral by a *quadrature*. A quadrature is simply a formula of the form

$$Q(f) = \sum_{j=1}^{p} w_j f(r_j) \tag{9}$$

with the property that

$$\int_R f(r)\, dr = Q(f) + E(f). \tag{10}$$

The term $E(f)$ is called the quadrature or truncation error. The points r_j are called quadrature nodes and the w_j are called weights. The goal is to approximate the definite integral of a function f by evaluating f at a finite number of sample points [12].

By enforcing (8) at each r_j and replacing the integral by a quadrature, we obtain the linear system

$$\sum_{j=1}^{n} w_{ij} G(r_i, r_j) \sigma(r_j) = \phi(r_i), \qquad i = 1, \ldots, n. \tag{11}$$

The w_{ij} are quadrature weights that depend on the Green's function G. If we take $A_{ij} = w_{ij} G(r_i, r_j)$, then multiplying the vector of charge densities by the matrix A gives the vector of potentials.

The error in a Nyström scheme is proportional to the quadrature error

$$|\sum_{j=1}^{n} w_{ij} G(r_i, r_j) \sigma(r_j) - \int_R G(r_i, r') \sigma(r')\, dr'|, \tag{12}$$

where σ is the true solution [5]. The entire task in applying a Nyström scheme is the development of quadrature weights w_{ij} that are accurate for a set of basis functions broad enough to closely approximate the solution. In the following section, we introduce new quadrature methods for three-dimensional extraction problems. We also refer the reader to the literature for quadrature techniques for smooth and singular functions in one and two dimensions [1, 4, 8, 13].

3 Quadrature Construction

Structures for capacitance extraction problems are typically described in terms of a mesh of triangles (or, more generally, polygonal regions) T_k in R^3. R is just the union of all triangles. In the following discussion, we shall assume for notational simplicity that all the triangles T_k in the mesh lie in the x-y plane. (In practice, we make local transformations to barycentric coordinates.) For triangle T_k, we choose a set $\{\psi_1^k, \ldots, \psi_p^k\}$ of basis functions which have values

$$\{1, x, y, x^2, xy, y^2, \ldots\} \tag{13}$$

in T_k and which are zero elsewhere. Then given a point r, we must construct a quadrature rule for computing

$$\int_{T_k} G(r, r')\psi_i^k(r')\, dr' \tag{14}$$

for $i = 1, \ldots, p$. The task is nontrivial because the Green's function $G(r, r')$ in (14) has an integrable singularity at the point $r = r'$. Good convergence and accuracy depend on the correct treatment of this singularity.

The integrals over T_k fall into two classes, as shown in Figure 1. If r is far from T_k ($|r - r'| > c$), we can use standard Gaussian quadrature [14]. To apply Gaussian quadrature, the points $r_1^k, \ldots, r_p^k$ inside of T_k where ϕ and σ are tabulated must be Gaussian nodes. Then the far-field quadrature weights are simply standard Gaussian weights $w_1^k, \ldots, w_p^k$ [14].

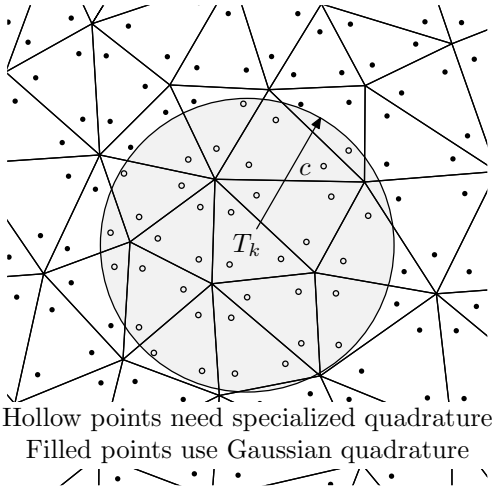

Figure 1: Integration regimes for the triangle T_k

Unfortunately, Gaussian quadrature only works well in regions where the Green's function $G(r, r')$ is smooth. When the singular point r is close to T_k ($|r - r'| \le c$), the singularity is strong and needs special treatment. In this regime, a special quadrature is constructed for evaluating the integral (14). We set up a system of equations which must be satisfied by the unknown weights $v_1^k, v_2^k, \ldots, v_p^k$. Let

$$a_{ij} = G(r, r_j^k)\psi_i(r_j^k) \tag{15}$$

for $i, j = 1, \ldots, p$. Then the requirement that that weights correctly integrate each basis function is equivalent to the weights satisfying the system

$$\sum_{j=1}^{p} a_{ij} v_j^k = \int_{T_k} G(r, r')\psi_i^k(r')\, dr'. \tag{16}$$

We solve for the weights v_j^k by LU factoring the $p \times p$ matrix a_{ij}. The weights will exactly integrate any linear combination of the basis functions convolved with $G(r, r')$. Note that the weights v_j^k depend both on the particular triangle T_k and on the singular point r. This is in contrast to the Gaussian weights, which depend on the triangle but not on the point. Fortunately, the near-field for any given triangle encompasses only a small number of other triangles. Hence, the number of weights which must be explicitly computed is small. In free space problems, we use analytic formulas to evaluate the singular and near-singular integrals needed on the right-hand side of (16) [9]. For layered media Green's functions, we analytically remove the singularity and use adaptive Gaussian quadrature to evaluate the remaining smooth part.

The integral over all of R to the point r is given by the sums over individual triangles:

$$\sum_{\text{near } T_k} \sum_{i=1}^{p} v_i^k G(r, r_i^k) + \sum_{\text{far } T_k} \sum_{i=1}^{p} w_i^k G(r, r_i^k). \tag{17}$$

The combination of all such quadrature rules as r ranges over the nodes r_i^k defines the matrix A in equation (11).

The distance at which the Gaussian weights cease to be adequate depends on the order of the quadrature, the accuracy required and the Green's function. Distance is measured from the triangle's centroid and scaled so that the distance to the farthest vertex of the triangle is one unit. To obtain 0.1% accuracy, using the free-space Green's function and a third-order rule, the Gaussian weights are sufficient for points at a distance of *only* 2. For typical discretizations, almost all the points are farther away than this. Hence the number of specialized quadratures is very small.

Note that the choice of basis functions $\{1, x, y, x^2, xy, y^2, \ldots\}$ we use is arbitrary. We know that if the solution is smooth then this choice

is a good one. For singular solutions, as found in problem domains with edges and corners, this choice is not optimal. Instead, it is probably better to use more sophisticated basis functions (at edges and corners) which incorporate the behaviour of the solution singularity [1]. We have not yet incorporated such specialized basis functions.

4 Rapid Matrix Solution

Solution of the linear system $A\sigma = \phi$ via direct factorization would be prohibitive. However, the matrix turns out to be extremely well-conditioned, and hence Krylov-subspace iterative schemes such as GMRES [10] can be used. Iterative solvers require application of the matrix A to a sequence of recursively generated vectors. The dominant costs become the $O(n^2)$ time and space required for constructing and storing the matrix and the $O(n^2)$ time required for each matrix-vector product. While this is already an improvement over direct factorization, the storage and computational cost is still excessive. Each of these costs can be reduced to $O(n \log n)$ for typical problems using the fast integral equation solver IES^3 [3].

The key idea behind IES^3 is to exploit the fact that typical Green's functions vary smoothly with distance. Consequently, large parts of the matrix A are numerically low-rank. These low-rank regions are represented as sparse outer products using the singular value decomposition (SVD). The SVD is an extremely effective tool for the compression of rank deficient matrices. Based on this observation, Kapur and Long [3] describe a scheme for recursively partitioning, sampling and compressing the matrix. Figure 2 is a "rank map" of a matrix for a typical problem. The rank map shows the partitioning of the matrix into submatrices and the rank of each submatrix. Although there are some strong off-diagonal interactions, the rank map is typical in problems where the time and memory requirements drop from $O(n^2)$ to $O(n \log n)$.

5 Examples

In this section we compare high-order Nyström methods to first-order collocation schemes. All matrices were solved using IES^3 and the experiments were run on a Sun Ultra Enterprise (248 MHz UltraSPARC-II CPU).

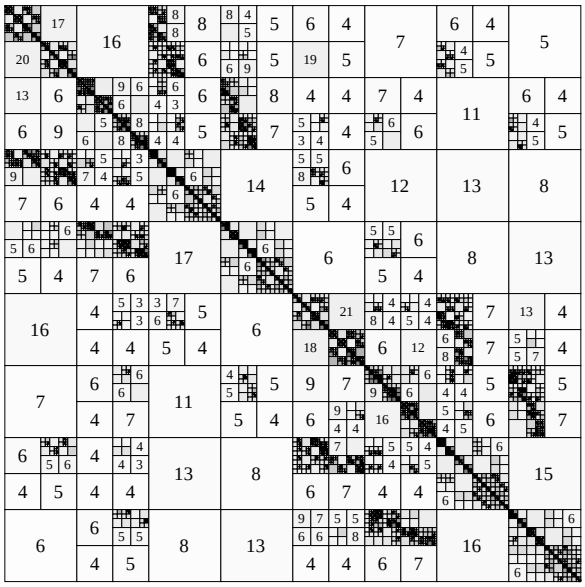

Figure 2: Rank map of a typical compressed matrix

5.1 Convergence analysis

The first example we consider is a 3x3 bus crossing. The bus lines were of a size typically found in a CMOS process. In the first example, all the triangles were approximately the same size, i.e., no special treatement was used at corners and edges where charge distribution is expected to vary rapidly. The simulator was run at various levels of discretization for four schemes: a first-order collocation method (one point per triangle), a second-order Nyström scheme (three points per triangle), a third-order scheme (six points per triangle) and a mixed-order scheme. The mixed-order scheme used six points for corner and edge triangles and three points for interior triangles. Comparisons between the methods are shown in Figures 3 and 4. The x-axis in both graphs gives the number of points (or matrix size). In Figure 3, the y-axis gives the percent error. The error is computed by running the third order Nyström method at a discretization of 68,000 points and computing $||C - \tilde{C}||/||C||$ where C is the capcitance matrix of the accurate solution and $\tilde{C}$ is the low-discretization capacitance matrix.

The following observations can be made from Figure 3. At the same number of points, a high-order Nyström method is more accurate than the low-order methods and the slopes indicate that the high-order schemes exhibit faster convergence. For the same number of points, the mixed-order method gives greater accuracy than the third-order method.

In Figure 4, the time for each of the methods is presented. At low levels of discretization, the low-

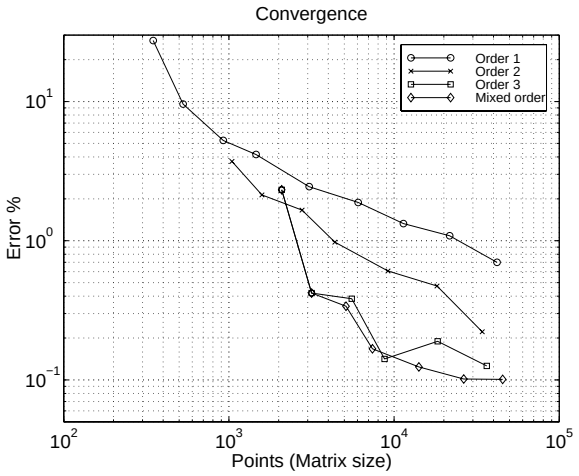

Figure 3: Comparison of convergence rates for a regular mesh

order schemes are slightly faster than the high-order schemes. This is because the number of singular and near-singular weights is relatively large. Since the number of special weights is approximately a constant times the number of triangles, at higher levels of discretization, the time for spent computing these weights becomes small compared to the total time. This is clear from Figure 4 where, for high discretizations, all the schemes take roughly the same amount of time.

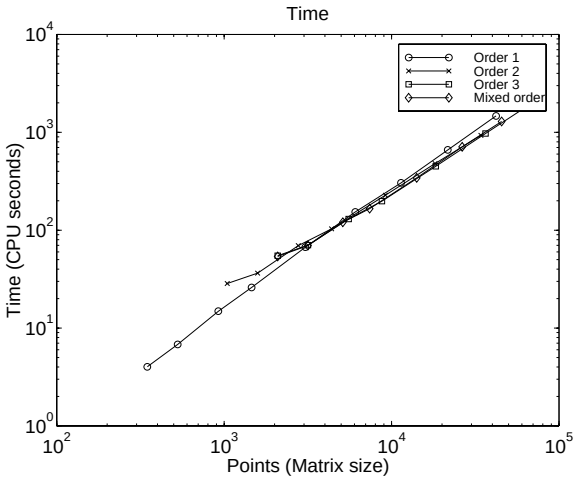

Figure 4: Comparison of timings

If the mesh is refined to have more triangles in regions where there is rapid variation in charge (typically near corners and edges) the matrix size can be further reduced. Even in this case, it turns out that the high-order Nyström methods maintain their advantage over the low-order schemes. In Figure 5 we show the convergence of solution with problem size. At around 20,000 points, the error using the regular mesh is twice that of the refined mesh.

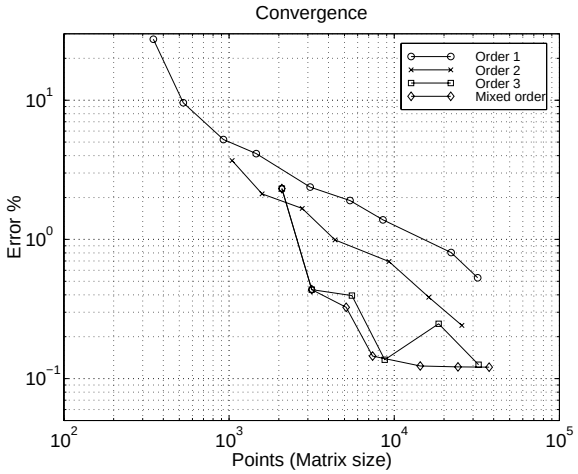

Figure 5: Comparison of convergence rates for an edge and corner-refined mesh

5.2 Simulating a memory cell

In this subsection, we consider the capacitance extraction of a memory cell array. Figure 6 is a coarse discretization of the 14 conductor array. Two IES^3-based simulations were run. The first simulation used first-order collocation while the second used the mixed-order Nyström scheme. Comparisons between the methods are shown in Figures 7 and 8. The error is computed by running the mixed-order Nyström method at a discretization of 112,000 points. At this discretization the total time required to compute the capacitance matrix was 4649 seconds. It took 1812 seconds to compress the matrix and 541 seconds to generate the mixed-order corrections. Approximately 164 seconds were required for each of the 14 solves (with around 25 iterations per solve). The advantage of using the Nyström method is clear from Figures 7 and 8.

5.3 Simulating an interdigitated capacitor

For structures with many fine features, getting accurate results requires extremely large discretizations. In this subsection, we consider is an interdigitated capacitor fabricated in CMOS. Figure 9 is a mesh of the capacitor. Two sets of IES^3-based simulations were run. The first set used first-order collocation while the second used the third-order Nyström scheme.

Since we could not run the first-order scheme to convergence on the full capacitor, we studied a ver-

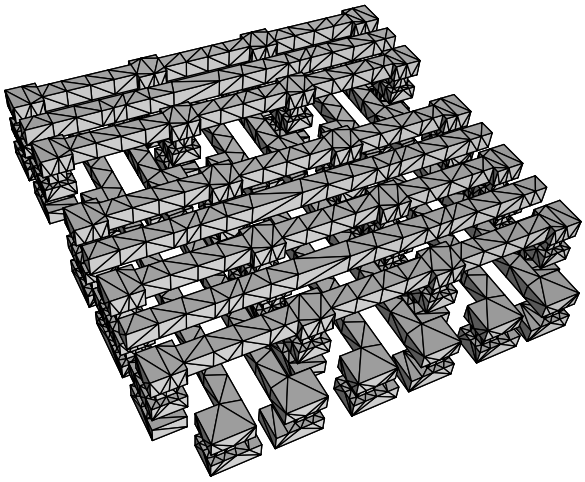

Figure 6: Mesh for a memory cell array

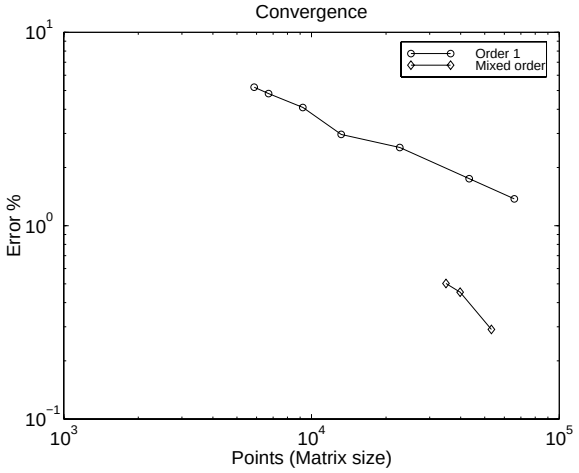

Figure 7: Convergence for the memory cell

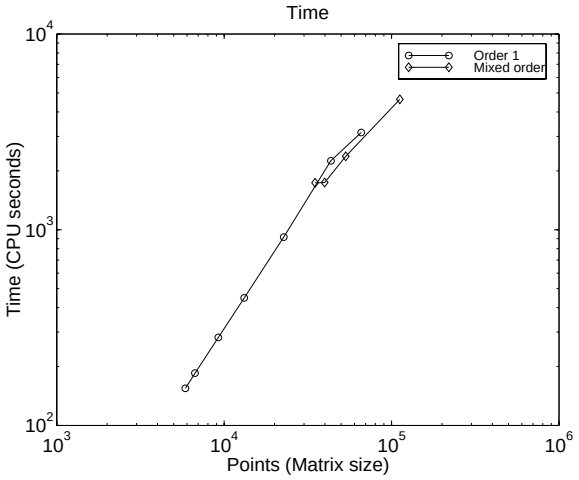

Figure 8: Timing for the memory cell

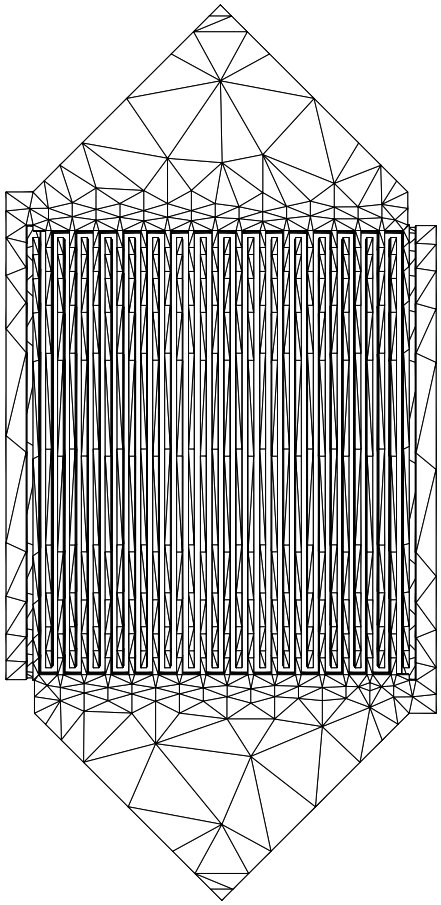

Figure 9: Interdigitated capacitor

sion with only a few fingers. Convergence of the mutual capacitance under both schemes is shown in Figure 12. The Nyström scheme shows a slight increase in capacitance with increasing discretization, presumably due to the fact that edge and corner solution singularities are not reflected in the quadrature. However, the Nyström approach at the lowest discretization gives much better accuracy than the collocation scheme at the highest discretization. Further, the efficiency of the two schemes is similar: at 10,000 points, the Nyström version takes 150 seconds while the collocation scheme requires 160. Thus, the superior convergence available with the high-order scheme comes at very little cost. The scaling of time and memory requirements with problem size is shown in Figures 10 and 11. For both collocation and the Nyström method, the resource requirements grow slightly faster than linearly.

For the full capacitor, the Nyström-based extractor using 38,142 points took just over an hour of CPU time (using layered Green's functions [15]) and predicted a capacitance of 252 fF. The average measured

value for this capacitor (over a population of eight) was 241 fF, with a standard deviation of 10 fF. At about 100,000 points the first order scheme had not come close to converging.

The timing for this example is significantly higher than similar sized examples in the previous subsection. In order to restrict the size of the mesh, elongated triangles were used in the fingers of the capacitor. Hence a relatively large number of special quadratures was required.

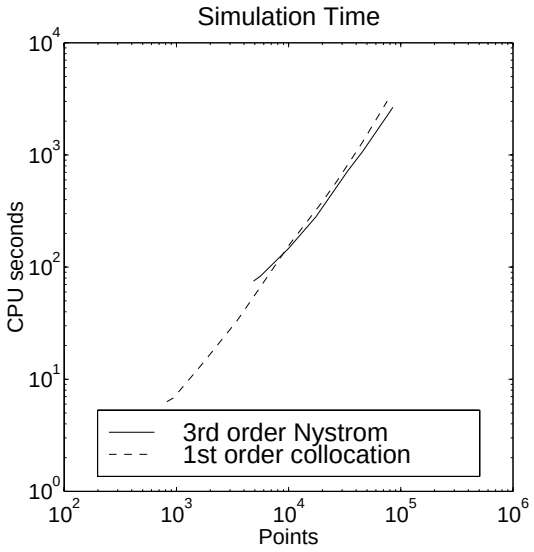

Figure 10: Time requirements for the interdigitated capacitor

6 Conclusions

In this paper we described a high-order Nyström method for efficient capacitance extraction in three dimensions. The method is based on combining Gaussian quadratures for far-field interactions with special purpose quadratures for the near-field interactions. These quadratures give an accurate discrete approximation to the integral operator. The resulting dense matrix is efficiently solved by combining the matrix sparsification of IES^3 with an iterative method. For the same level of discretization, the high-order schemes were an order of magnitude more accurate than the first-order approaches at the same computational cost. Hence, in order to get the same level of accuracy as a first-order method the high-order Nyström method requires a much smaller matrix. The Nyström method was found to maintain its advantage even for refined meshes. The high-order methods were run on a number of examples and were

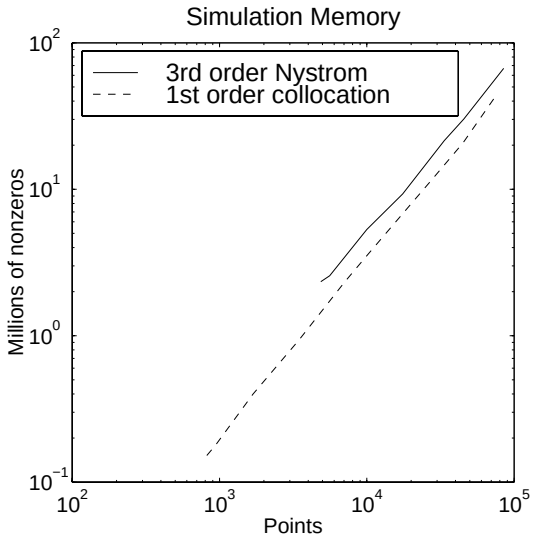

Figure 11: Memory requirements for the interdigitated capacitor

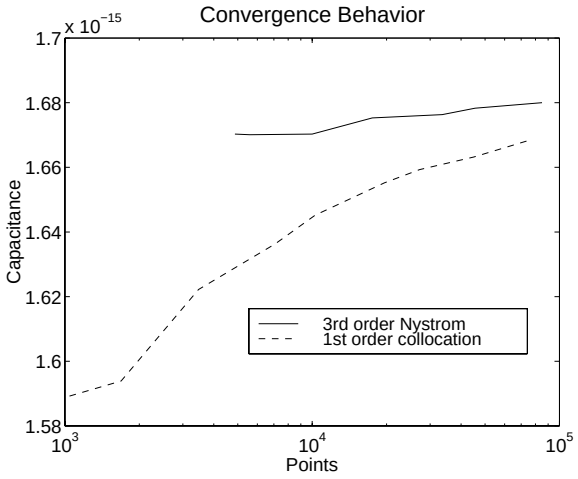

Figure 12: Convergence of mutual capacitance for the interdigitated capacitor example

found to be far superior to the first-order methods in all cases.

Acknowledgements

Peter Kinget designed and measured the interdigitated capacitor. Rohini Gupta and Ken Haruta provided the memory cell example.

References

[1] B. K. Alpert. High order quadratures for integral operators with singular kernels. *Journal of Computational and Applied Mathematics*, 60(3):367–378, 1995.

[2] X. Cai, K. Nabors, and J. White. Efficient Galerkin techniques for multipole-accelerated capacitance extraction of 3-D structures with multiple dielectrics. In *Proceedings of the 16th Conference on Advanced Research in VLSI, Chapel Hill, NC*, March 1995.

[3] S. Kapur and D. E. Long. IES^3: A fast integral equation solver for efficient 3-dimensional extraction. In *37th International Conference on Computer Aided Design*, Nov 1997.

[4] S. Kapur and V. Rokhlin. High-order corrected trapezoidal rules for singular functions. *SIAM Journal of Numerical Analysis*, 34(4):1331–1356, August 1997.

[5] S. G. Michlin. *Integral Equations and their Applications to Certain Problems in Mechanics, Mathematical Physics, and Technology.* Pergamon Press, New York, 1957.

[6] K. Nabors, F. T. Korsmeyer, F. T. Leighton, and J. White. Preconditioned, adaptive, multipole-accelerated iterative methods for three-dimensional first-kind integral equations of potential theory. *SIAM Journal on Scientific and Statistical Computing*, 15(3):713–735, May 1994.

[7] K. Nabors and J. White. Fastcap: A multipole-accelerated 3-D capacitance extraction program. *IEEE Transactions on Computer-Aided Design*, 10(10):1447–1459, November 1991.

[8] V. Rokhlin. End-point corrected trapezoidal quadrature rules for singular functions. *Journal of Computational and Applied Mathematics*, 20:51–62, 1990.

[9] D. Rosen and D. E. Cormack. The continuation approach: A general framework for the analysis and evaluation of singular and near-singular integrals. *SIAM J. Applied Mathematics*, 55(3):723–762, June 1995.

[10] Y. Saad and M. H. Shultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, July 1986.

[11] J. R. Shewchuk. Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator. In *First Workshop on Applied Computational Geometry*, pages 124–133. Association for Computing Machinery, May 1996.

[12] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis.* Springer-Verlag, New York, 1979.

[13] J. Strain. Locally-corrected multidimensional quadrature rules for singular functions. *SIAM Journal of Scientific Computing*, 16:992–1017, 1995.

[14] A. H. Stroud. *Gaussian Quadrature Formulas.* Prentice-Hall, New York, 1966.

[15] J. Zhao, S. Kapur, D. E. Long, and W. W.-M. Dai. Efficient three-dimensional extraction based on static and full-wave layered Green's functions. In *Proceedings of the 35th Design Automation Conference*, June 1998.