−2 dB gain compression point for the optimum and nonoptimum second-harmonic terminating impedance. We can conclude that, with a difference of only 58° between the optimum and nonoptimum source second-harmonic reflection coefficient phase, tuning for minimum spectral regrowth needs to be carefully done. With an optimum choice of the harmonic-terminating impedance, an amplifier for use in PHS can operate into the −3 dB gain compression point and still meet the adjacent channel requirements. The method presented here can be applied to any modulation format, and the reduction in spectral regrowth can be optimized for a given adjacent channel power requirement.

REFERENCES

1. D. Jing, W.S. Chan, and C.W. Li, Spectral regrowth of digital signal through an amplifier and a new linearity method, Proc Asia Pacific Microwave Conf, Hong Kong, Dec. 1997, pp. 525−528.
2. M. Nakamura, M. Maeda, H. Takehara, H. Masato, S. Morimoto, and Y. Ota, High efficiency linear amplifiers using source second-harmonic control for digital cellular phones, Proc Asia Pacific Microwave Conf, KAIST, Taejon, Korea, Oct. 1995, pp. 64−67.
3. M. Maeda, H. Masato, H. Takehara, M. Nakamura, S. Morimoto, H. Fujimoto, Y. Ota, and O. Ishikawa, Source second harmonic control for high efficiency power amplifiers, IEEE Trans Microwave Theory Tech 43 (1995), 2952−2958.
4. Focus Microwaves Inc., Concept for load pull measurements with harmonic impedance control, Application Note 24-A, 1995.

INCOMPLETE LU PRECONDITIONER FOR FMM IMPLEMENTATION

Kubilay Sertel[1] and John L. Volakis[1]
[1] Radiation Laboratory
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, Michigan 48109-2122

ABSTRACT: *An incomplete LU (ILU) preconditioner using the near-field matrix of the fast multipole method (FMM) is investigated to increase the efficiency of the iterative conjugate gradient squared (CGS) solver. Unlike the conventional LU, ILU requires no fill ins, and hence no extra memory and CPU time in computing the LU decomposed preconditioner. It is shown that, due to the nature of the near-field matrix, ILU preconditioning decreases the number of iterations dramatically.* © 2000 John Wiley & Sons, Inc. Microwave Opt Technol Lett 26: 265−267, 2000.

1. INTRODUCTION

The fast multipole method (FMM) has been used to speed up the method-of-moments (MoM) solution of large-scale electromagnetic scattering and radiation problems [1–11]. FMM achieves its speed up by using an indirect fast computation of the matrix−vector product in the context of an iterative solution of the MoM matrix. Various iterative solvers [4] are available, including the conjugate gradient (CG), biconjugate gradient (BiCG), conjugate gradient squared (CGS), quasi-minimal residual (QMR), and generalized minimal residual

(GMRES). The convergence of all of these iterative solvers is, of course, dictated by the matrix condition, which typically deteriorates as the matrix size increases. Therefore, iterative solutions of such large, fully populated matrix systems inevitably require some kind of preconditioning. Otherwise, the propagation of numerical errors during the execution of the iterative solution may lead to convergence failures.

Various preconditioning techniques for improving the condition of the system can be found in [12]. Diagonal and block-diagonal preconditioners for multilevel FMM implementations were reported in [6]. Here, we present an ILU preconditioner for MoM and FMM implementations. By its nature, the ILU preconditioner is constructed using the near-field FMM matrix, and is shown to significantly reduce the number of iterations required for convergence.

2. METHOD OF MOMENTS AND THE FAST MULTIPOLE METHOD

The MoM formulation of electromagnetic scattering problems using second-order curvilinear quadrilateral elements for surface modeling is given in [13]. The FMM implementation used herewith was built upon that in [13, 14].

Briefly, for the electric-field integral equation (EFIE), the resulting linear system after Galerkin's testing is of the form (an $e^{-i\omega t}$ time convention has been assumed and suppressed)

$$\sum_{n=1}^{N} Z_{mn} a_n = V_m, \qquad m = 1, 2, \ldots, N \tag{1}$$

where

$$Z_{mn} = \int_s ds\, \mathbf{f}_m(\mathbf{r}) \cdot \int_{s'} ds' \left[\mathbf{f}_n(\mathbf{r}') + \frac{1}{k^2} \nabla' \cdot \mathbf{f}_n(\mathbf{r}') \nabla \right] \frac{e^{ikR}}{R} \tag{2}$$

and

$$V_m = \frac{4\pi i}{k\eta} \int_s ds\, \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{E}^i(\mathbf{r}). \tag{3}$$

Here, $\{a_n\}$ refers to the column containing the unknown coefficients of the surface current expansion

$$\mathbf{J}(\mathbf{r}) \approx \sum_{n=1}^{N} a_n \mathbf{j}_n(\mathbf{r}). \tag{4}$$

Also, as usual, $\mathbf{r}$ and $\mathbf{r}'$ denote the observation and source point locations, $\mathbf{E}^i(\mathbf{r})$ is the incident excitation plane wave at $\mathbf{r}$, $\hat{t}$ is the vector tangent to the surface at $\mathbf{r}$, $\eta = 120\pi$ denotes the free-space impedance, and $k = 2\pi/\lambda$ is the free-space wavenumber.

The key components in an FMM implementation are

- iterative solution of the MoM system of equations $[Z]\{a\} = \{V\}$
- fast evaluation of the matrix−vector product $[Z]\{a\}$.

Fast evaluation of the matrix−vector product [using $O(N^{1.5})$ or fewer resources instead of $O(N^2)$] is attained by approximating the pertinent Green's function [1]. It can be shown [1] that the CPU time per iteration (or per matrix−vector product) for FMM is $O(N^{1.5})$ instead of $O(N^2)$. However, it should be understood that the above CPU estimates are asymptotic in the sense that they represent values which are approached for very large N. The actual efficiency of the

implementation depends on the choice of various parameters. These choices control the constant in front of the asymptotic behavior of the CPU requirements. However, choices of the FMM parameters for faster implementation inevitably lead to errors in the matrix−vector product, inducing erroneous minima for the iterative solver. This aspect may lead to unacceptable error since it may cause the solver to converge at one of these minima, especially when nonmonotonic solvers (e.g., CG, BiCG, or CGS) are employed.

The specific integral equation formulation, be it the EFIE, the magnetic-field integral equation (MFIE), or the combined-field integral equation (CFIE), also affects the solution convergence.

3. PRECONDITIONER

For large-scale simulations, possibly with geometrical surface details (e.g., antenna arrays on aircraft), the density of the surface mesh cannot be expected to be uniform. Nevertheless, a nonuniform mesh is well known to produce ill-conditioned MoM matrix equations. Also, different formulations of the same electromagnetic problem are associated with different condition numbers. For example, the CFIE formulation is known to give rise to much better conditioned systems than the EFIE or MFIE. Further, as noted above, the FMM implementation introduces erroneous minima in the solution domain. The use of a preconditioner is therefore essential for robust implementations of iterative solvers.

Although the diagonal preconditioner is simple and leads to significant convergence improvements, it does so for diagonally dominant matrices. Block-diagonal preconditioners are more robust, but require renumbering of the grid or matrix rearranging so that the dominant matrix terms are clustered around the diagonal. This can be done easily for 2-D problems, but is quite difficult, if at all possible, in three dimensions. Alternatively, when the FMM is used to speed up the iterative solution, we have the natural choice of using the near-field portion of the MoM matrix for preconditioning. These near-field elements are the largest in magnitude, and constitute the unapproximated portion of the system matrix.

One preconditioning approach is to perform a direct LU decomposition on the unapproximated part of the matrix. However, depending on the sparsity pattern of the near-field matrix, this may require a significant amount of fill ins. For large-scale simulations, these fill ins may become a bottleneck in memory utilization. Alternatively, the fill-in requirement of direct LU can be resolved by performing the ILU factorization. The ILU is the same as a direct LU algorithm, but avoids fill ins of elements in the decomposed LU matrices. This also results in less CPU utilization.

3.1. ILU Preconditioner for FMM. We employed the ILU algorithm from [12]. The pseudocode is repeated below for completeness:

```
for i = 2,...,n, do:
    for k = 1,...,i 1 and for (i,k) in NZ(Z)
  do:
        compute z_ik = z_ik/z_kk
        for j = k + 1,...,n and for (i, j) in
  NZ(Z) do:
            compute z_ij = z_ij − z_ik z_kj
        end do
    end do
end do
```

Here, NZ(Z) is the sparsity pattern of the near-field matrix Z, and the conventional LU decomposition algorithm is only applied to the nonzero entries of the matrix. Hence, memory utilization is not affected, and moreover, the sparsity pattern of the stored ILU matrix is identical to that of the original matrix. Thus, further memory savings are attained using ILU decomposition.

4. PERFORMANCE OF THE PRECONDITIONED CGS SOLVER

To evaluate the performance of the ILU preconditioner, we considered a perfectly electrically conducting (PEC) ogive geometry (depicted in Fig. 1). For this study, the ILU preconditioner was implemented in the matrix systems based on the EFIE, MFIE, and CFIE formulations. As described above, the FMM near-field matrix is used as a preconditioner in the context of the CGS iterative solver. The size of the matrix system was 480, and refers to a 10 in × 2 in × 2 in ogive with its long axis coincident with the *x*-axis. All calculations were carried out at 5.91 GHz. This is indeed a very small system, and serves the purpose of validating the preconditioning scheme. Also, the ogive was chosen due to its irregular grid at the tips. A uniformly meshed sphere does not serve as a good test example due to its well-conditioned system.

Figure 2 shows the residual error as a function of iteration number for the EFIE matrix. It is seen that, due to the irregular mesh around the sharp tips of the ogive, the CGS solver does not converge in fewer than 50 iterations without preconditioning. However, when the ILU preconditioner is introduced, convergence is dramatically improved, requiring only $N/50$ iterations to achieve an error of 10^{-5}.
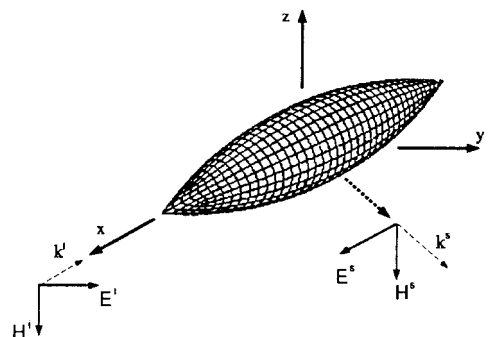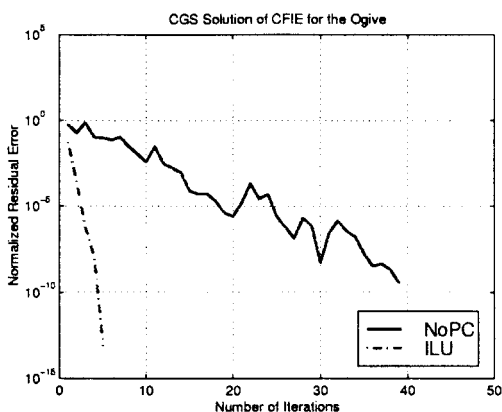

Figure 1 Ogive geometry and the simulation setup
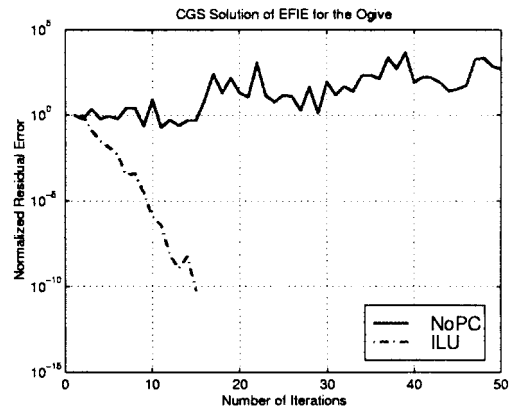
Figure 2 Performance of ILU in conjunction with EFIE formulation
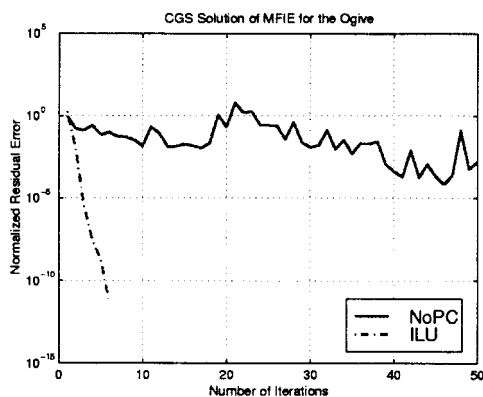
Figure 3 Performance of ILU in conjunction with MFIE formulation

Figure 4 Performance of ILU in conjunction with CFIE ($\alpha = 0.5$) formulation

Nonpreconditioned and preconditioned solution data for the MFIE matrix are given in Figure 3. Since the MFIE formulation produces better conditioned systems, the residual error behavior of the MFIE is better than that of the EFIE. Nevertheless, convergence without preconditioning is very slow. When the ILU preconditioner is included, convergence is reached down to a residual of 10^{-5} in about $N/150$ iterations. Figure 4 gives the corresponding convergence curves for the CFIE matrix. Since the CFIE system is better conditioned (than the EFIE and the MFIE), convergence is now achieved, even without preconditioning. The use of ILU simply reduces the number of iterations from $N/25$ down to $N/150$ to reach a residual error of 10^{-5}.

Table 1 summarizes the performance of the ILU preconditioner for a larger problem. Much like the ogive, the scatterer in this simulation has sharp edges and tips, as well as smooth sections. Also, the mesh is quite distorted and nonuniform around these edges. Nevertheless, the performance of the ILU preconditioner is quite impressive. Specif-

ically, ILU improved the convergence of the CFIE ($\alpha = 0.5$) matrix down to $N/10,000$, leading to a solution time of only 5 min for a 53,000-unknown system on an eight-processor SGI Origin 2000.

Based on the above performance evaluations, we can conclude that the ILU preconditioner can be used[1] to improve the performance of iterative solvers in FMM implementations without increasing the memory utilization for the preconditioner matrix.

REFERENCES

1. R. Coifman, V. Rokhlin, and S. Wandzura, The fast multipole method for the wave equation: A pedestrian prescription, IEEE Antennas Propagat Mag 35 (1993), 7–12.
2. W.C. Chew, J. Jin, C. Lu, E. Michielssen, and J.M. Song, Fast solution methods in electromagnetics, IEEE Trans Antennas Propagat 45 (1997), 533–543.
3. N. Lu and J.-M. Jin, Application of the fast multipole method to finite-element boundary-integral solution of scattering problems, IEEE Trans Antennas Propagat 44 (1996), 781–786.
4. J.L. Volakis, A. Chatterjee, and L.C. Kempel, Finite element methods for electromagnetics, IEEE Press, Piscataway, NJ, 1998.
5. R.J. Burkholder and D.H. Kwon, High frequency asymptotic acceleration of the fast multipole method, Radio Sci 31 (1996), 1199–1206.
6. J.M. Song, C.C. Lu, and W.C. Chew, Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects, IEEE Trans Antennas Propagat 45 (1997), 1488–1493.
7. S.S. Bindiganavale and J.L. Volakis, Comparison of three FMM techniques for solving hybrid FE/BI systems, IEEE Trans Antennas Propagat 39 (1997), 47–59.
8. R.L. Wagner and C.W. Chew, A ray-propagation fast multipole algorithm, Microwave Opt Technol Lett 7 (1997), 435–438.
9. N. Lu and W.C. Chew, Fast far-field approximation for calculating the RCS of large objects, Microwave Opt Technol Lett 8 (1995), 238–241.
10. J.M. Song and W.C. Chew, Multilevel fast multipole algorithm for solving combined field integral equations of electromagnetic scattering, Microwave Opt Technol Lett 10 (1995), 14–19.
11. J.M. Song and W.C. Chew, Fast multipole method solution using parametric geometry, Microwave Opt Technol Lett 7 (1994), 760–765.
12. Y. Saad, Iterative methods for sparse linear systems, PWS Publishing, Boston, MA, 1996.
13. G.E. Antilla and N.G. Alexopoulos, Scattering from complex 3D geometries by a curvilinear hybrid finite element-integral equation approach, J Opt Soc Amer A 11 (1994), 1445–1457.
14. K. Sertel and J.L. Volakis, The effect of FMM parameters on the solution of PEC scattering problems, IEEE Int Symp Antennas Propagat Dig, Orlando, FL, 1999, pp. 624–627.

© 2000 John Wiley & Sons, Inc.

[1]The code used in this, FMM-SWITCH, is based on the SWITCH code developed by Northrop Grumman Corporation.

TABLE 1   Performance of ILU for a Large-Scale Complex Target with Sharp Edges and Tips on an Eight-Processor SGI Origin 2000

Number of Unknowns	Matrix Fill Time (min) (Eight Processors)	Preconditioned LU Time (min) (One Processor)	Number of Iterations	Residual Error	Time per Solution (min) (Eight Processors)
53,000	77	81	5	0.001	5