Incomplete LU Preconditioning Strategies for MLFMA [1]

Tahir Malas and Levent Gürel*
Department of Electrical and Electronics Engineering
Bilkent University, TR-06800, Bilkent, Ankara, Turkey

Introduction

We consider the solution of large electromagnetic scattering problems involving three-dimensional arbitrarily shaped targets using the multilevel fast multipole algorithm (MLFMA). A class of incomplete LU preconditioners are employed to reduce the iteration counts for the solutions of the electric-field integral equation (EFIE) and the combined-field integral equation (CFIE). Even though CFIE produces better conditioned systems compared to EFIE, we still need EFIE for problems involving open geometries since CFIE is limited to the solution of closed geometries.

Surface integral equations can be converted into dense linear systems of equations in the form of $\overline{\mathbf{A}} \cdot \mathbf{x} = \mathbf{b}$, using the method of moments (MOM). High numbers of unknowns arising in practical applications prohibit the use of direct solution methods, which require $\mathcal{O}(n^2)$ memory and $\mathcal{O}(n^3)$ CPU time for n unknowns. Nonetheless, application of fast iterative methods with MLFMA offers great advantages for the solution of large problems. MLFMA reduces both memory and computational complexity of matrix-vector multiplication from $\mathcal{O}(n^2)$ to $\mathcal{O}(n \log n)$, which is required at least once per iteration.

In order to achieve convergence in a reasonable number of iterations, we need an effective preconditioner. To reduce the storage complexity to $\mathcal{O}(n \log n)$, MLFMA stores only the near-field part of the MOM matrix, denoted by $\overline{\mathbf{N}}$. Since $\overline{\mathbf{N}}$ is the best available approximation to the system matrix $\overline{\mathbf{A}}$, we can use it as a preconditioner and solve (for example) the left-preconditioned system $\overline{\mathbf{N}}^{-1} \cdot \overline{\mathbf{A}} \cdot \mathbf{x} = \overline{\mathbf{N}}^{-1} \cdot \mathbf{b}$. The inversion of the near-field matrix can be performed via LU factorization. However, during the factorization of sparse matrices, except some specific cases, sparsity is lost. Nonetheless, by sacrificing some (hopefully) small elements of the exact factorization, we end up with an incomplete LU (ILU) preconditioner. This idea is widely used and proven to be successful in many areas of scientific computation.

Depending on the criteria for the elimination of the matrix entries, two general kinds of ILU preconditioners can be prescribed. For the first one, elimination is based on the entry locations. For example, consider an incomplete factorization of the near-field matrix, $\overline{\mathbf{N}} \approx \overline{\mathbf{L}} \cdot \overline{\mathbf{U}}$. If we retain nonzero values of $\overline{\mathbf{L}}$ and $\overline{\mathbf{U}}$ only at the nonzero positions of $\overline{\mathbf{N}}$, we end up with the no-fill ILU method, or ILU(0). For well-conditioned problems this simple idea works well [5]. On the other hand, since this technique does not consider the numerical values of the entries, it becomes ineffective in predicting the locations of the largest entries for particularly non-diagonal dominant and indefinite matrices. Alternatively, a second class of ILU preconditioners is based on the priciple of dropping the matrix elements depending

[1] This work was supported by the Turkish Academy of Sciences in the framework of the Young Scientist Award Program (LG/TUBA-GEBIP/2002-1-12), by the Scientific and Technical Research Council of Turkey (TUBITAK) under Research Grant 103E008, and by contracts from ASELSAN and SSM.

on their magnitudes. Among such methods, ILUT(τ, p) has been successful for systems obtained from a wide range of applications. During the factorization, ILUT drops matrix elements that are smaller than τ times the 2-norm of the current row; and of all the remaining entries no more than the p largest ones are kept. ILUT is known to yield more accurate and stable factorizations than level-of-fill methods with the same amount of fill-in [4].

However for some cases, even though the factorization terminates normally, the incomplete factors sometimes turn out to be unstable. The common reasons of instability are in general excessive dropping and small pivots [4]. If the problem is related to the small pivots, one can significantly increase the quality of the ILUT preconditioner, by using partial pivoting as in the complete factorization case. The resulting preconditioner is called ILUTP [5].

In order to understand the quality of the preconditioner, or to understand the reason for failure when it occurs, we can use $\|(\overline{\mathbf{L}} \cdot \overline{\mathbf{U}})^{-1} \cdot \mathbf{e}\|_\infty$, where $\mathbf{e}$ is the vector of ones. This statistic is called *condest* (for condition estimate) and it provides an upper bound for $\|(\overline{\mathbf{L}} \cdot \overline{\mathbf{U}})^{-1}\|_\infty$[4]. If the *condest* value is not very high, but the preconditioner still does not work, we can deduce that fill-in should be increased to achieve a successful preconditioner. On the other hand, if the *condest* value is high, we can first try pivoting to remedy the situation instead of including more elements in the incomplete factors.

ILU preconditioners were previously applied to scattering problems by Sertel and Volakis [1], Carpentieri et al. [2], and Lee et al. [3]. ILU(0) was used in [1] and [2] producing contradictory results for EFIE, but was successful for CFIE in [1]. Later, ILUT was used with the hybrid surface-volume integral equations and shown to be successful on many test problems [3]. However, the study neither included the commonly used EFIE and CFIE formulations, nor the application of pivoting or any other techniques to increase the effectiveness of the preconditioner. Considering the prior success on many problems and the availability of ILU preconditioners in many publicly accessible packages (for example, see [6]), in this work we seek to establish a strategy for selecting both the most appropriate types of ILU preconditioners and their associated sets of parameters that will render them robust preconditioners in CFIE and EFIE formulations.

Numerical Results

In our experiments, we use the GMRES solver with no restart, since we find it to be more robust for EFIE compared to other Krylov subspace solvers. For CFIE, we also use the same solver since it results in fewer number of iterations compared to others, reducing the solution time. Starting with the zero vector as the initial guess, we stop the iterations when the initial residual is reduced six orders of magnitude. The maximum number of iterations is set to 1500.

For ILUT and ILUTP, instead of presenting several results with changing parameters, we suggest to fix the drop tolerance as low as 10^{-6}, and then let the maximum number of nonzero elements per row be the same as the average number of nonzero elements in the near-field matrix. Hence, we guarantee that the memory requirement of the preconditioner does not exceed that of the near-field matrix. Also, our experiments reveal that this strategy is close to an optimum solution.

For CFIE, we use two canonical geometries, i.e., a sphere (S) and a cube (C); two quasi-canonical geometries, i.e., a thin box (TB) and a wing (W); and two real-life problems, i.e., a helicopter (H) and a stealth target called Flamme (F). For EFIE we use a patch (P), a half sphere (HS), a triangular prism (TP), and an open cube (OC). Both triangular prism and open cube have an open face.

Since CFIE produces quite diagonally-dominant matrices with low condition numbers, ILU(0) can be expected to work well. Though not shown here, when we compare ILU(0) and ILUT, we observe that they result in very similar *condest* values and iteration numbers. The setup time of ILUT is higher, hence we conclude that ILU(0) should be the preconditioner of choice for CFIE among the ILU preconditioners.

Geom-		LU	Block-Jacobi		ILU(0)			SAI		
etry	n	*iter*	*iter*	*time*	*iter*	*setup*	*time*	*iter*	*setup*	*time*
S	132003	29	32	684	29	22	665	29	23,102	23,721
C	131436	26	28	419	26	27	485	27	25,065	25,488
TB	147180	41	106	1,289	45	270	1,025	64	298,478	299,301
W	117945	31	52	779	32	46	541	37	73,100	73,586
F	78030	63	115	1,096	66	43	693	76	96,369	96,369
H	183546	42	106	3,080	44	145	1,738	61	234,614	236,463

Table 1: Comparisons of preconditioners for CFIE

In Table 1 we compare ILU(0) with other commonly used preconditioners. The block-Jacobi preconditioner constructed from the self-interactions of the last-level clusters has negligible setup time. We also compare the sparse approximate inverse (SAI) preconditioner, for which we let the pattern of the approximate inverse be the same as the near-field matrix, so that it has the same storage cost as that of ILU(0). For comparison purposes, we also include the exact solution of the near-field matrix, which we denote by LU. For all geometries, we see that ILU(0) reduces the iteration numbers with respect to block-Jacobi. Though the same observation holds also for SAI, its setup time is too high, rendering it inapplicable to sequential programs. Since the iteration numbers of ILU(0) are very close to those of LU, ILU(0) seems to be the optimal choice of preconditioning CFIE in the context of this study.

The *condest* values for EFIE in Table 2 indicate that ILU(0) produces highly instable factors, hence precluding convergence. Since ILUT also produces instable factors for the half sphere geometry, we used 0.5 pivoting tolerance (ILUTP5) and 1.0 pivoting tolerance (ILUTP). Convergence is achieved with ILUTP, however, with a relatively large *condest* value for the half sphere, and hence with more iterations compared to ILUTP5.

When we compare ILUTP5 with other preconditioners in Table 3, we see that ILUTP5 reduces the iteration numbers by an order of magnitude compared to Jacobi when it succeeds to converge. Block-Jacobi is not included in Table 3 since it performs even poorer than no preconditioning. Though SAI produces comparable

Geom- etry	n	ILU(0) *condest*	 *iter*	ILUT *condest*	 *iter*	ILUTP5 *condest*	 *iter*	ILUTP *condest*	 *iter*
P	137792	6.3E+09	-	1,398	82	1,350	81	2,545	78
OC	171655	9.6E+05	-	192	377	240	376	2,892	376
TP	163871	5.3E+05	-	948	268	835	253	2,424	251
HS	116596	6.3E+05	-	1.5E+15	-	582	110	22,755	156

Table 2: ILU results for EFIE

iteration numbers, the setup time is still too high. Finally, we observe that iteration numbers of ILUTP5 are not extremely higher than those of LU, indicating that the ILUTP5 preconditioner provides a good approximation to the near-field matrix.

Geom- etry	n	LU *iter*	Jacobi *iter*	 *time*	 *iter*	ILUTP5 *setup*	 *time*	 *iter*	SAI *setup*	 *time*
P	137792	53	833	16,209	81	661	2,167	92	19,955	21,384
OC	171655	332	-	-	376	2,243	9,833	354	207,436	213,619
TP	163871	195	-	-	253	996	6,883	396	57,606	66,093
HS	116596	93	1,052	25,947	110	1,353	3,579	156	22,079	25,066

Table 3: Comparisons of preconditioners for EFIE

Conclusion

In this work, we show that the ILU preconditioners can be used in the iterative solutions of scattering problems. We deduce that ILU(0) can be safely applied to CFIE, yielding very close performance to the exact solution of the near-field matrix. For EFIE, establishing *condest* to be a strong indicator for the quality of the resulting ILU preconditioner, we propose the following strategy. Before the iterations begin, compute *condest* for ILUT. If the condition estimate is not very high, (e.g., less than 10^4), use ILUT as the preconditioner. Otherwise, switch to ILUTP5. With this strategy, we have obtained robust and effective preconditioners for our test problems.

References

[1] K. Sertel and J.L. Volakis, "ILU Preconditioner for Fast Multipole Method(FMM)," *Microwave and Optical Technical Letters*, vol. 28, pp. 265–267 2000.

[2] B. Carpentieri, I.S. Duff, and L. Giraud, "Experiments with sparse preconditioning of dense problems from electromagnetic applications," Technical Report, TR/PA/00/04, 1999.

[3] J. Lee, J. Zhang and, C. Lu, "Incomplete LU preconditioning for large scale dense complex linear systems from electromagnetic wave scattering problems," *J. Comput. Phys.*, vol. 185, pp. 158–175, 2003.

[4] E. Chow and Y. Saad, "Experimental study of ILU preconditioners for indefinite matrices," *J. Comput. Appl. Math.*, vol. 86, pp. 387–414, 1997.

[5] Y. Saad, *Iterative Methods for Sparse Linear Systems.* Philadelphia: SIAM, 2003.

[6] S. Balay, K. Buschelman, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, B. Smith, and H. Zhang, "PETSc Web page." Available at http://www.mcs.anl.gov/petsc.