

Solution of Large-Scale Scattering Problems with the Multilevel Fast Multipole Algorithm Parallelized on Distributed-Memory Architectures

Özgür Ergül^{1,2} and Levent Gürel^{1,2}

¹Department of Electrical and Electronics Engineering

²Computational Electromagnetics Research Center (BiLCEM)

Bilkent University, TR-06800, Bilkent, Ankara, Turkey

E-mail: ergul@ee.bilkent.edu.tr, lgurel@bilkent.edu.tr

Abstract—We present the solution of large-scale scattering problems involving three-dimensional closed conducting objects with arbitrary shapes. With an efficient parallelization of the multilevel fast multipole algorithm on relatively inexpensive computational platforms using distributed-memory architectures, we perform the iterative solution of integral-equation formulations that are discretized with tens of millions of unknowns. In addition to canonical problems, we also present the solution of real-life problems involving complicated targets with large dimensions.

I. INTRODUCTION

For the numerical solution of scattering problems in electromagnetics, integral-equation formulations provide accurate results when they are discretized appropriately by using small elements with respect to wavelength [1]. Simultaneous discretizations of the scatterer and the integral equations lead to dense matrix equations, which can be solved iteratively using efficient acceleration methods, such as the multilevel fast multipole algorithm (MLFMA) [2]. On the other hand, accurate solutions of many real-life problems require discretizations with millions of elements, which result in dense matrix equations with millions of unknowns. To solve these large problems, it is helpful to increase computational resources by assembling parallel computing platforms and at the same time by parallelizing the solvers. Of the various parallelization schemes for MLFMA, the most popular use distributed-memory architectures by constructing clusters of computers with local memories connected via fast networks [3]–[8]. However, parallelization of MLFMA is not trivial. Simple parallelization strategies usually fail to provide efficient solutions because of the communications between the processors and the unavoidable duplication of some of the computations over multiple processors.

In this paper, we present a parallel MLFMA implementation for the efficient solution of scattering problems involving tens of millions of unknowns. Our approach involves load-balancing and partitioning techniques to distribute the tasks equally among the processors and to minimize the inter-processor communications. We demonstrate the accuracy and efficiency of our implementations on a scattering problem involving a sphere of radius 110λ discretized with 41,883,638 unknowns. To the best of our knowledge, this is the largest

integral-equation problem ever solved. In addition to canonical problems, we also solve real-life problems involving complicated geometries discretized with large numbers of unknowns.

II. NUMERICAL SOLUTION OF THE INTEGRAL EQUATIONS

For the solution of the scattering problems involving three-dimensional arbitrary shapes, the unknown surface current density $\mathbf{J}(\mathbf{r})$ is expanded in a series of basis functions $\mathbf{b}_n(\mathbf{r})$ as

$$\mathbf{J}(\mathbf{r}) = \sum_{n=1}^N a_n \mathbf{b}_n(\mathbf{r}), \quad (1)$$

where a_n represents the unknown coefficients of the basis functions for $n = 1, 2, \dots, N$. Testing the boundary conditions on the surface of the scatterer and applying discretization on the resulting integral equation, we obtain $N \times N$ dense matrix equation

$$\sum_{n=1}^N Z_{mn} a_n = v_m, \quad m = 1, 2, \dots, N, \quad (2)$$

where the matrix element Z_{mn} is the electromagnetic interaction of the n th basis and m th testing functions. Among various choices for the integral-equation formulations, we prefer the combined-field integral equation (CFIE), which produces well-conditioned matrix equations that are easy to solve iteratively [9],[10].

III. STRUCTURE OF MLFMA

In the iterative solution of the scattering problems, matrix-vector multiplications (MVMs) are required at each iteration [11]. MLFMA reduces the complexity of the MVMs related to an $N \times N$ dense matrix equation from $O(N^2)$ to $O(N \log N)$ [2]. This is achieved by considering the matrix elements as the electromagnetic interactions and calculating the far-field interactions in group-by-group manner. As depicted in Fig. 1, the scatterer is included in a cubic box and the computational domain is recursively divided into subboxes (clusters). The smallest clusters include the basis and testing functions. Then, using the clustering information, a multilevel

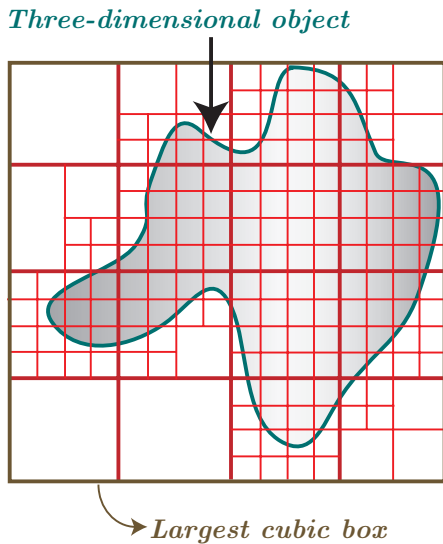


Fig. 1. Recursive clustering to divide the computational domain into subdomains (clusters).

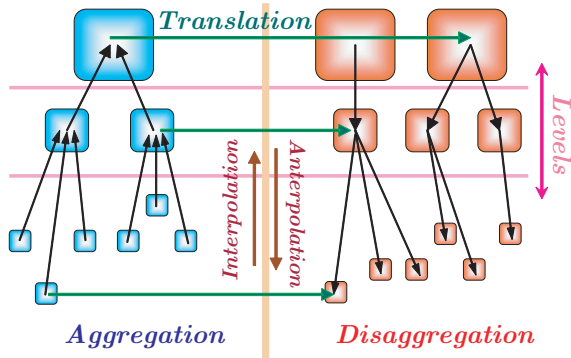


Fig. 2. Multilevel tree structure and MLFMA operations.

tree structure is constructed as depicted in Fig. 2. MLFMA splits the MVMs as

$$\bar{\mathbf{Z}} \cdot \mathbf{x} = \bar{\mathbf{Z}}_{NF} \cdot \mathbf{x} + \bar{\mathbf{Z}}_{FF} \cdot \mathbf{x}. \quad (3)$$

In (3), the near-field interactions denoted by $\bar{\mathbf{Z}}_{NF}$ are calculated directly and stored in memory. These interactions are related to the basis and testing functions that are located in the same or in two touching clusters in the lowest level. On the other hand, the rest of the interactions, i.e., the far-field interactions denoted by $\bar{\mathbf{Z}}_{FF}$, are computed approximately via three main stages performed on the multilevel tree [12]:

- 1) **Aggregation:** Radiated fields at the centers of the clusters are calculated from the bottom of the tree structure to the highest level. Oscillatory nature of the Helmholtz solutions requires that the sampling rate for the fields depend on cluster size as measured by the wavelength [13]. During the aggregation stage, we employ local Lagrange interpolation to match the different sampling rates of the consecutive levels [14].
- 2) **Translation:** Radiated fields at the centers of the clusters

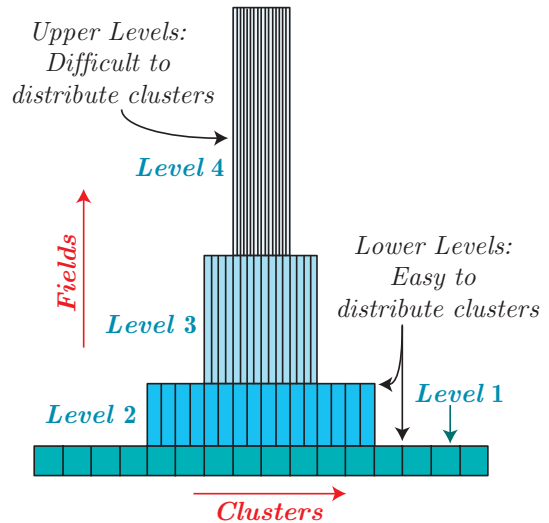


Fig. 3. Simple partitioning of the tree structure based on distributing the clusters among the processors in all levels.

are translated into incoming fields for other clusters. For a basis cluster at any level, there are $O(1)$ testing clusters to translate the radiated field.

- 3) **Disaggregation:** The incoming fields at the centers of the clusters are calculated from the top of the tree structure to the lowest level. At the lowest level, the incoming fields are received by the testing functions. During the disaggregation stage, we employ local Lagrange interpolation (transpose interpolation) method to match the different sampling rates of the consecutive levels [14],[15].

We note that the lower levels of the multilevel tree include many clusters with low sampling rates for the radiated and incoming fields. On the other hand, higher levels involve a few clusters with large numbers of samples.

IV. PARTITIONING OF THE TREE STRUCTURE

Because of its complicated structure, parallelization of MLFMA is not trivial. For high efficiency, it is essential to distribute the tree structure among the processors with minimal duplication and communication between the processors. A simple partitioning of the multilevel tree is depicted in Fig. 3, which involves the distribution of the clusters among the processors. In this scheme, each cluster at any level is assigned to a single processor. This strategy works efficiently for the lower levels involving many clusters. However, problems arise when the clusters in the higher levels are distributed among processors, especially when the number of processors is comparable to the number of clusters [7]. For these levels, it is difficult to distribute the clusters among the processors without a duplication. In addition, dense communications are required between the processors, which reduce the efficiency of the parallelization significantly.

To improve the parallelization efficiency, a hybrid partitioning approach is introduced in [6], where different strategies are

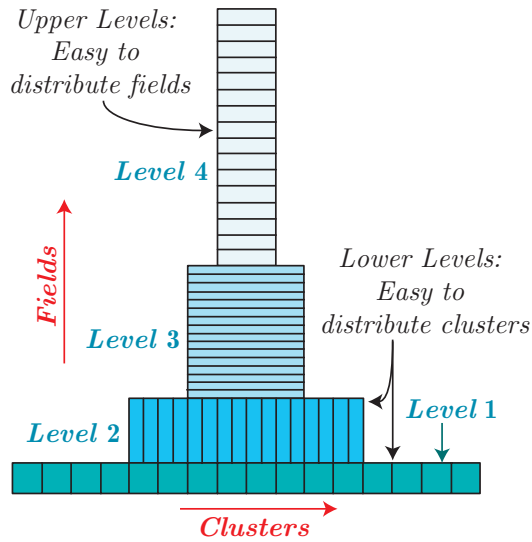


Fig. 4. Hybrid partitioning of the tree structure involving different strategies for the upper and lower levels.

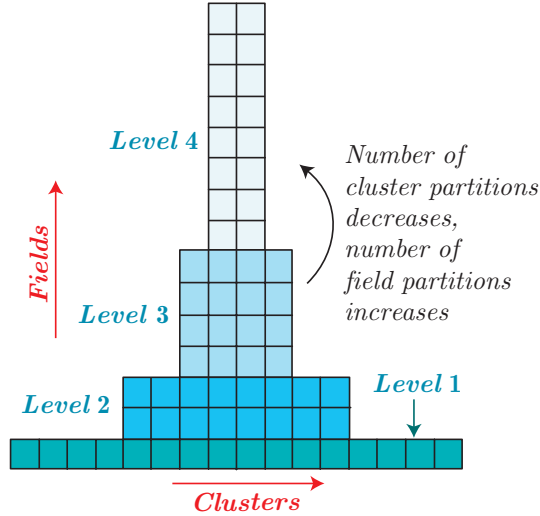


Fig. 5. Hierarchical partitioning based on adjusting the number of partitions in both directions (fields and clusters) appropriately.

applied for the lower and higher levels of the tree structure. As depicted in Fig. 4, each cluster in the lower (distributed) levels is assigned to a single processor similar to the simple partitioning scheme. In the higher (shared) levels, however, processor assignments are made on the basis of the fields of the clusters, not on the basis of the clusters themselves. Then, each cluster is shared by all processors and each processor is assigned to the same portion of the fields of all clusters. Since the fields in the higher levels have large sampling rates, the samples can be distributed efficiently among the processors. In addition, dense one-to-one communications between the processors during the translations are eliminated.

Although the hybrid partitioning strategy increases the parallelization efficiency significantly compared to simple paral-

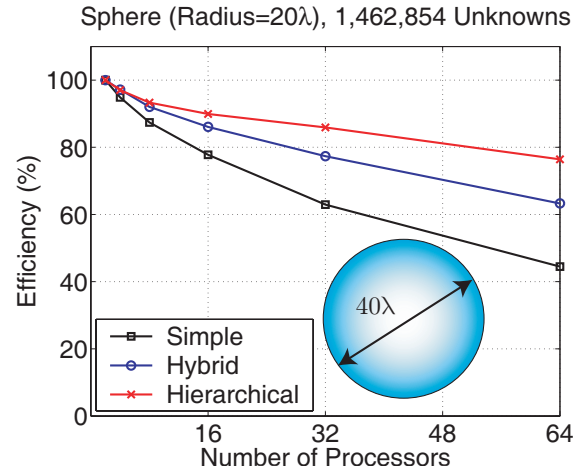


Fig. 6. Parallelization efficiency for the solution of a scattering problem involving a sphere of radius 20λ discretized with 1,462,854 unknowns.

lization approach, this is not sufficient. For some of the levels of the tree structure, neither distributing the fields nor the clusters among the processors is efficient. Consequently, we propose to use a hierarchical partitioning scheme as described in Fig. 5 to further increase the parallelization efficiency. In this strategy, partitioning is performed in both directions (clusters and fields) for all levels and we adjust the numbers of partitions appropriately by considering the numbers of clusters and the samples of the fields. As depicted in Fig. 5, the clusters in the lowest level are still distributed among all processors without any partitioning of the fields. As we proceed to the higher levels, however, the numbers of partitions for the clusters and the fields are systematically decreased and increased, respectively. The hierarchical partitioning strategy is detailed in the Appendix.

As an example, we demonstrate the efficiency of MLFMA parallelization for the solution of a scattering problem involving a sphere of radius 20λ . The problem is discretized with 1,462,854 unknowns and solved on a cluster of quad-core Intel Xeon 5355 processors connected via an Infiniband network. Fig. 6 depicts the efficiency when the solution is parallelized into various numbers of processes. The parallelization efficiency is defined as

$$\varepsilon_p = \frac{2T_2}{pT_p}, \quad (4)$$

where T_p is the processing time of the solution with p processes. Fig. 6 shows that the efficiency is increased significantly by using the hierarchical partitioning strategy compared to the hybrid and simple strategies, when all three strategies are optimized by employing load-balancing algorithms. For 16 processes, the efficiency is 91% using the hierarchical parallelization, while it is 86% and 77% for the hybrid and simple parallelization strategies, respectively. Even in the 64-process case, the parallelization efficiency is above 75% with the hierarchical partitioning approach.

V. COMMUNICATIONS IN PARALLEL MLFMA

In parallel MLFMA, processors need to communicate with each other to transfer data. Using appropriate partitioning schemes (such as hierarchical partitioning) and load-balancing algorithms significantly reduces the data traffic. However, the remaining communications must be organized carefully. Communications required in the MVMs by parallel MLFMA can be summarized as follows:

- 1) Near-Field to Far-Field Switch: Using load-balancing algorithms, the rows of the matrix equation are partitioned differently for the near-field and far-field interactions. Therefore, we perform all-to-one (gather) and one-to-all (scatter) communications in each MVM to match the different partitioning schemes for the output vector.
- 2) Inflation and Deflation for the Interpolation and Anterpolation Operations: During the aggregation stage, interpolation operations in a processor require samples that are located in other processors [6]. These are obtained by one-to-one communications. Similarly, some of the data produced by the anterpolation operations during the disaggregation stage should be sent to other processors via one-to-one communications.
- 3) Data Exchange From Level to Level: Using the hierarchical parallelization strategy, the partitioning should be changed between levels during the aggregation and disaggregation stages. This is achieved by exchanging data between pairs of processors.
- 4) Intra-Processor Translations: Some of the translations are related to basis and testing clusters that are located in different processors. Therefore, one-to-one communications are required between the processors to perform these translations [6],[7].

To improve the efficiency of the parallelization, we use nonblocking send and receive operations of message passing interface (MPI) to transfer the data. For high efficiency, it is also essential to use high-speed networks to connect the processors.

VI. SOLUTIONS OF LARGE-SCALE PROBLEMS

By constructing a sophisticated simulation environment based on parallel MLFMA, we are able to solve scattering problems discretized with tens of millions of unknowns. As an example, we present the solution of a very large scattering problem involving a sphere of radius 110λ , which is discretized with 41,883,648 unknowns. For the solution of the problem, 9-level MLFMA is employed and parallelized into 16 processes. The near-field and far-field interactions are calculated with 1% error. The setup and iterative solution parts take about 274 and 290 minutes, respectively. Using BiCGStab algorithm [11] and an efficient block-diagonal preconditioner (BDP) [2], the number of iterations to reduce the residual error below 10^{-3} is only 19. The peak memory requirement is 229 GB using the single-precision representation to store the data. To present the accuracy of the solution, Fig. 7 depicts the normalized bistatic radar cross section (RCS/λ^2)

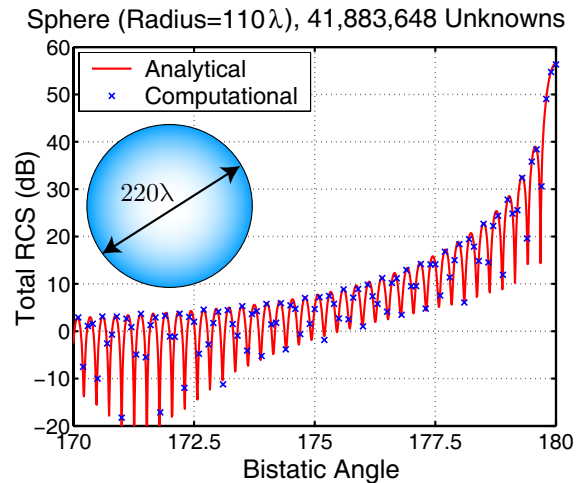


Fig. 7. Bistatic RCS (in dB) of a sphere of radius 110λ discretized with 41,883,648 unknowns from 170° to 180° , where 180° corresponds to the forward-scattering direction.

values in decibels (dB). Analytical values obtained by a Mie-series solution is plotted as a reference from 170° to 180° , where 180° corresponds to the forward-scattering direction. Fig. 7 shows that the computational values sampled at 0.1° are in agreement with the analytical curve. For more quantitative information, we define a relative error as

$$e_R = \frac{\|\mathbf{A} - \mathbf{C}\|_2}{\|\mathbf{A}\|_2}, \quad (5)$$

where \mathbf{A} and \mathbf{C} are the analytical and computational RCS values, respectively, $\|\cdot\|_2$ is the l^2 -norm defined as

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{s=1}^S |\mathbf{x}[s]|^2}, \quad (6)$$

and S is the number of samples. The relative error is 0.047 in the 170° – 180° range.

Next, we present the solution of a real-life problem involving the Flamme, which is a stealth airborne target, as detailed in [16] and also depicted in Fig. 8. The scattering problem is solved at 16 GHz and the maximum dimension of the Flamme is 6 meters, corresponding to 320λ . Using $\lambda/10$ triangulation, the problem is discretized with 24,782,400 unknowns. Solution of the problem is performed by a 10-level MLFMA parallelized into 16 processes. As shown in Fig. 8, the nose of the target is directed towards the x axis and it is illuminated by a plane wave propagating in the $-x$ direction. Both θ and ϕ polarizations are considered. After the setup, which takes about 104 minutes, the problem is solved twice (for two polarizations) in about 470 minutes. Using BiCGStab and BDP, the numbers of iterations to reduce the residual error below 10^{-3} are 39 and 36, respectively, for the θ and the ϕ polarizations of the plane-wave excitation. Fig. 9 presents the co-polar RCS values in dBm^2 on the x - y plane as a function of the bistatic angle ϕ . In the plots, 0° and 180° correspond to the back-scattering and forward-scattering directions, respectively.

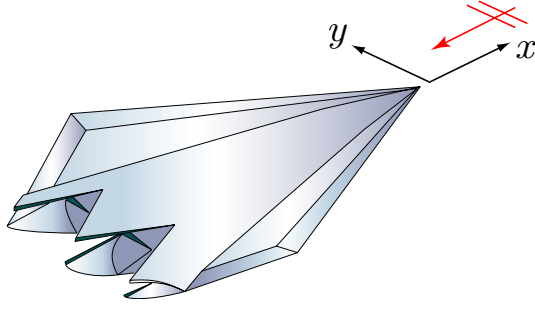


Fig. 8. A stealth airborne target Flamme.

VII. CONCLUSION

In this paper, we consider fast and accurate solutions of large-scale scattering problems discretized with tens of millions of unknowns using a parallel MLFMA implementation. We demonstrate the accuracy of our implementations by considering a canonical scattering problem involving a sphere of radius 110λ discretized with 41,883,638 unknowns. We also demonstrate the effectiveness of our implementation on a real-life problem involving the Flamme geometry with a size larger than 300λ .

APPENDIX

HIERARCHICAL PARALLELIZATION OF MLFMA

In MLFMA, far-field interactions are calculated in a multilevel scheme using a tree structure constructed by placing the scatterer in a cubic box and recursively dividing the computational domain into subboxes. Without losing generality, we consider a smooth scatterer with a maximum electrical dimension of kD , where $k = 2\pi/\lambda$ is the wavenumber. Using a discretization with $\lambda/10$ mesh size for such a geometry leads to N unknowns, where $N = O(k^{1/2}D^{1/2})$. Constructing a tree structure with $L = O(\log N)$ levels, the smallest box size is in the range from 0.15λ to 0.3λ and there are $O(1)$ unknowns in each cluster in the lowest level. At level l from 1 to L , the number of clusters can be approximated as

$$N_l \approx N_{l-1}/4 \quad (l \neq 1), \quad (7)$$

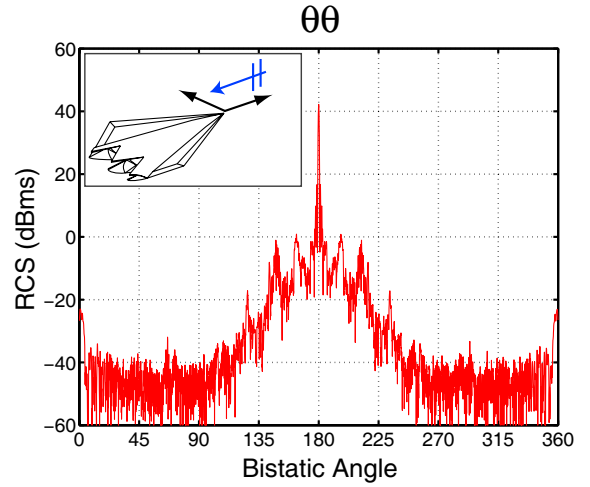
$$N_l \approx 4^{l-1}N_1, \quad (8)$$

where $N_1 = O(N)$ and $N_L = O(1)$. In other words, the number of clusters decreases approximately by a factor of four from a level to the next upper level.

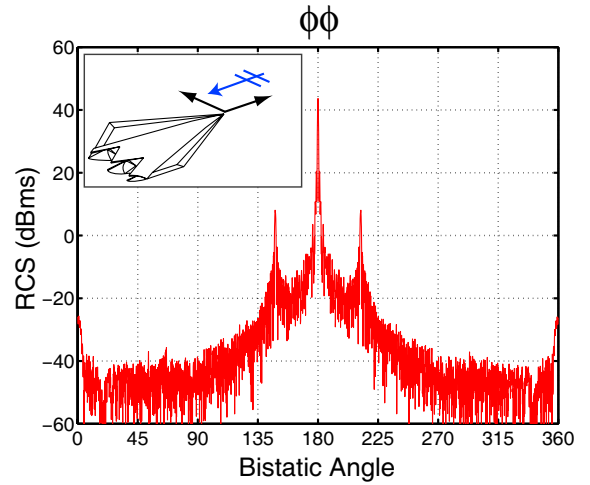
In our implementations, radiated and incoming fields of the clusters are sampled uniformly in the ϕ direction, while we use the Gauss-Legendre quadrature in the θ direction. There are a total of $S_l = (T_l + 1) \times (2T_l + 2)$ samples required for a cluster in level l , where T_l is the truncation number determined by the excess bandwidth formula as [13]

$$T_l \approx 1.73ka_l + 2.16(d_0)^{2/3}(ka_l)^{1/3}. \quad (9)$$

In (9), a_l is the cluster size at level l and d_0 is the desired digits of accuracy. We note that $S_1 = 2(T_1 + 1)^2 = O(1)$ since



(a)



(b)

Fig. 9. Bistatic RCS (in dBm²) of the stealth airborne target Flamme at 16 GHz. Maximum dimension of the Flamme is 6 meters corresponding to 320λ . The target is illuminated by a plane wave propagating in the $-x$ direction. Co-polar RCS is plotted for (a) θ and (b) ϕ polarizations of the plane-wave excitation.

the size of the clusters in the lowest level is independent of N . In general,

$$S_l \approx 4S_{l-1} \quad (l \neq 1), \quad (10)$$

$$S_l \approx 4^{l-1}S_1, \quad (11)$$

and $S_L = O(N)$. Considering the number of clusters (N_l) and the samples of the fields (S_l), all levels of MLFMA have equal importance with $N_l S_l = N_1 S_1 = O(N)$ complexity in terms of processing time and memory.

Using the hierarchical partitioning strategy, we distribute the clusters and the samples of the radiated and incoming fields among the processors. The partitioning is performed carefully by considering the numbers clusters and the samples at each level. In a parallelization scheme with p processors, where $p = 2^i$ for some integer $i > L - 1$, it is appropriate to choose

the number of partitions for the clusters as

$$p_{c,l} = \frac{p}{2^{l-1}}. \quad (12)$$

Then, the samples of the fields are divided into

$$p_{s,l} = \frac{p}{p_{c,l}} = 2^{l-1} \quad (13)$$

partitions for each level $l = 1, 2, \dots, L$. We note that the samples are partitioned only along θ direction (not along ϕ direction). Using the hierarchical partitioning, the number of clusters assigned to each processor $q = 1, 2, \dots, p$ at level $l = 1, 2, \dots, L$ can be approximated as

$$N_l^q \approx \frac{N_l}{p_{c,l}} \approx 4^{1-l} N_1 \frac{2^{l-1}}{p} = 2^{1-l} \frac{N_1}{p}, \quad (14)$$

while the number of samples assigned to each processor is

$$S_l^q \approx \frac{S_l}{p_{s,l}} \approx \frac{4^{l-1} S_1}{2^{l-1}} = 2^{l-1} S_1. \quad (15)$$

During the aggregation and disaggregation stages, one-to-one communications are required due to the partitioning of the samples [6]. For a cluster in level l , the processor q has $S_l^q = (T_l^q + 1) \times (2T_l + 2)$ samples, where $T_l^q \approx T_1$ is approximately constant for the entire tree structure. This is an important advantage of using the hierarchical partitioning strategy, which provides a well-balanced distribution of the samples for all levels. For an interpolation operation in a processor, the amount of data required from other processors is proportional to number of samples in the ϕ direction, i.e., $2T_l + 2$, per cluster. Similarly, an anterpolation operation produces same amount of data to be sent to other processors. Considering all clusters, the processing time for communications during the aggregation or disaggregation from level l to next level can be expressed as

$$t_{agg/dis} \propto N_l^q T_l = 2^{1-l} \frac{N_1}{p} 2^{l-1} T_1 \propto \frac{N_1 T_1}{p}, \quad (16)$$

which is the same for all levels. To switch the partitioning scheme from level to level, each processor exchanges half of its data produced during the aggregation and disaggregation stages. The processing time for these communications can be expressed as

$$t_{exch} \propto N_l^q S_l^q = 2^{1-l} \frac{N_1}{p} 2^{l-1} S_1 = \frac{N_1 S_1}{p}. \quad (17)$$

Finally, due to the partitioning of the clusters, some of the translations are related to the basis and testing clusters that are located in different processors. Therefore, one-to-one communications are required also during the translation stage. These communications are achieved by pairing the processors and transferring the radiated fields of the clusters between the pairs [7]. In general, each processor is paired one by one with other $p_{c,l} - 1$ processors, while the number of cluster-cluster interactions required to be performed for each pair is proportional to $N_l^q = 2^{1-l} N_1/p$. In addition, the amount of the transferred data is $S_l^q = 2^{l-1} S_1$ for each cluster-cluster

interaction. As a consequence, the communication time for the translations can be approximated as

$$t_{trans} \propto \frac{p}{2^{l-1}} \frac{2^{1-l} N_1}{p} 2^{l-1} S_1 = \frac{N_1 S_1}{2^{l-1}}. \quad (18)$$

ACKNOWLEDGMENT

This work was supported by the Scientific and Technical Research Council of Turkey (TUBITAK) under Research Grant 105E172, by the Turkish Academy of Sciences in the framework of the Young Scientist Award Program (LG/TUBA-GEBIP/2002-1-12), and by contracts from ASELSAN and SSM. Computer time was provided in part by a generous allocation from Intel Corporation.

REFERENCES

- [1] A. J. Poggio and E. K. Miller, "Integral equation solutions of three-dimensional scattering problems," in *Computer Techniques for Electromagnetics*, R. Mittra, Ed. Oxford: Pergamon Press, 1973, Chap. 4.
- [2] J. Song, C.-C. Lu, and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," *IEEE Trans. Antennas Propag.*, vol. 45, no. 10, pp. 1488–1493, Oct. 1997.
- [3] S. Velamparambil, W. C. Chew, and J. Song, "10 million unknowns: Is it that big?," *IEEE Ant. Propag. Mag.*, vol. 45, no. 2, pp. 43–58, Apr. 2003.
- [4] M. L. Hastriter, "A study of MLFMA for large-scale scattering problems," Ph.D. thesis, University of Illinois at Urbana-Champaign, 2003.
- [5] G. Sylvand, "Performance of a parallel implementation of the FMM for electromagnetics applications," *Int. J. Numer. Meth. Fluids*, vol. 43, pp. 865–879, 2003.
- [6] S. Velamparambil and W. C. Chew, "Analysis and performance of a distributed memory multilevel fast multipole algorithm," *IEEE Trans. Antennas Propag.*, vol. 53, no. 8, pp. 2719–2727, Aug. 2005.
- [7] Ö. Ergül and L. Gürel, "Efficient parallelization of multilevel fast multipole algorithm," in *Proc. European Conference on Antennas and Propagation (EuCAP)*, no. 350094, 2006.
- [8] L. Gürel and Ö. Ergül, "Fast and accurate solutions of integral-equation formulations discretised with tens of millions of unknowns," *Electronics Lett.*, vol. 43, no. 9, pp. 499–500, Apr. 2007.
- [9] D. R. Wilton and J. E. Wheeler III, "Comparison of convergence rates of the conjugate gradient method applied to various integral equation formulations," *Progress in Electromagnetics Research PIER 05*, pp. 131–158, 1991.
- [10] L. Gürel and Ö. Ergül, "Comparisons of FMM implementations employing different formulations and iterative solvers," in *Proc. IEEE Antennas and Propagation Soc. Int. Symp.*, vol. 1, 2003, pp. 19–22.
- [11] S. Balay, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang, *PETSc Users Manual*, Argonne National Laboratory, 2004.
- [12] W. C. Chew, J.-M. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Boston, MA: Artech House, 2001.
- [13] S. Koc, J. M. Song, and W. C. Chew, "Error analysis for the numerical evaluation of the diagonal forms of the scalar spherical addition theorem," *SIAM J. Numer. Anal.*, vol. 36, no. 3, pp. 906–921, 1999.
- [14] Ö. Ergül and L. Gürel, "Enhancing the accuracy of the interpolations and anterpolations in MLFMA," *IEEE Antennas Wireless Propagat. Lett.*, vol. 5, pp. 467–470, 2006.
- [15] A. Brandt, "Multilevel computations of integral transforms and particle interactions with oscillatory kernels," *Comp. Phys. Comm.*, vol. 65, pp. 24–38, Apr. 1991.
- [16] L. Gürel, H. Bağcı, J. C. Castelli, A. Cheraly, and F. Tardivel, "Validation through comparison: measurement and calculation of the bistatic radar cross section (BRCS) of a stealth target," *Radio Sci.*, vol. 38, no. 3, Jun. 2003.