Hierarchical parallelisation strategy for multilevel fast multipole algorithm in computational electromagnetics

Ö. Ergül and L. Gürel

A hierarchical parallelisation of the multilevel fast multipole algorithm (MLFMA) for the efficient solution of large-scale problems in computational electromagnetics is presented. The tree structure of MLFMA is distributed among the processors by partitioning both the clusters and the samples of the fields appropriately for each level. The parallelisation efficiency is significantly improved compared to previous approaches, where only the clusters or only the fields are partitioned in a level.

Introduction: Surface integral equations are commonly used to formulate electromagnetic scattering and radiation problems involving complicated three-dimensional objects with arbitrary shapes [1]. By discretising the integral-equation formulations, we obtain dense matrix equations. They can be solved iteratively by accelerating the matrix-vector multiplications using the multilevel fast multipole algorithm (MLFMA) [2]. Using MLFMA, matrix-vector multiplications related to an $N \times N$ dense matrix equation can be performed in $O(N\log N)$ time using $O(N\log N)$ memory. However, accurate solutions of many real-life problems require discretisations with millions of unknowns, which cannot be solved easily by the sequential implementations of MLFMA running on a single processor. To solve such large problems, it is helpful to increase computational resources by assembling parallel computing platforms and at the same time by parallelising MLFMA. In this way, it has become possible to solve problems with 20–30 million unknowns on relatively inexpensive computing platforms [3–8]. On the other hand, parallelisation of MLFMA is not trivial owing to the complicated structure of this algorithm. Simple parallelisation strategies usually fail to provide efficient solutions because of the communication among the processors and the unavoidable duplication of some of the computations over multiple processors [9]. In this Letter, we present a hierarchical strategy for the efficient parallelisation of MLFMA. We compare our strategy with previous parallelisation schemes to demonstrate the improved efficiency, especially when the number of processors is large.

Tree structure of MLFMA: Elements of an $N \times N$ matrix obtained by the discretisation of a surface integral equation correspond to the interactions of the basis and testing functions defined on the surface of the object. MLFMA performs the matrix-vector multiplications efficiently by calculating these interactions in a group-by-group manner involving three main stages, i.e. aggregation, translation and disaggregation [2]. These stages are performed in a multilevel scheme using a tree structure constructed by including the scatterer in a cubic box and recursively dividing the computational domain into subboxes. During the aggregation stage, radiated fields at the centres of the clusters (nonempty boxes) are calculated proceeding from the bottom of the tree structure to the highest level. Then, the translation stage is performed by translating the radiated fields at the centres of the clusters to the incoming fields at the centres of other clusters in the same level. Finally, the total incoming fields at the centres of the clusters are calculated from the top of the tree structure to the lowest level during the disaggregation stage.

In the lowest level of the multilevel tree, there are $O(N)$ clusters. The number of clusters decreases from each level to the next upper level and it becomes $O(1)$ in the highest level involving translations. The number of samples for the radiated and incoming fields depends on cluster size as measured by the wavelength. Therefore, fields of the clusters in the lower levels are sampled coarsely, while the fields of the clusters in the higher levels require finer sampling. Considering the number of clusters and the samples of the fields, all levels of MLFMA have $O(N)$ complexity in terms of processing time and memory. As a consequence, an efficient parallelisation of MLFMA should attempt to obtain the best partitioning for each level by minimising the communications and duplications among the processors.

Partitioning of multilevel tree: For the parallelisation of MLFMA, the main task is to distribute the tree structure among the processors. A simple partitioning of a three-level tree is shown in Fig. 1*a*, where the levels are represented by two-dimensional rectangular boxes including various numbers of clusters (horizontal dimension) and samples of the fields (vertical dimension). Each level is partitioned among eight processors. In the simple partitioning scheme, clusters in all levels are distributed among the processors and each cluster at any level is assigned to a single processor. This strategy works efficiently for lower levels involving many clusters. For higher levels, however, it is difficult to distribute small numbers of clusters among the processors without duplication [9]. In addition, dense communications among the processors during the translations become significant for higher levels since large amounts of data are transferred, which reduces the efficiency of the parallelisation significantly [6, 9].

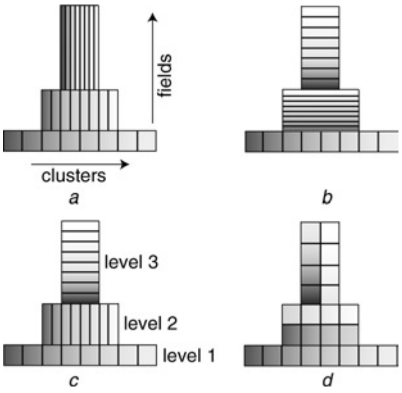

Fig. 1 *Various strategies for partitioning of tree structure of MLFMA*

a Simple partitioning, where clusters are distributed in all levels
b, *c* Hybrid partitioning with shared and distributed levels
d Hierarchical partitioning

To improve the parallelisation efficiency, a hybrid partitioning approach is introduced in [6], where different strategies are applied for lower and higher levels of the tree structure. As shown in Figs. 1*b* and 1*c*, the simple partitioning scheme is preserved in lower (distributed) levels so that the clusters in these levels are still distributed among the processors. In 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. In other words, each cluster is shared by all processors and each processor is assigned to the same portion of the fields of all clusters. In this way, higher levels are distributed efficiently among the processors, since the fields in those levels have high sampling rates. In addition, the translations in the shared levels can be performed efficiently without any communication among the processors.

The hybrid partitioning strategy increases the parallelisation efficiency significantly compared to the simple partitioning approach. Nevertheless, there are some levels at the middle of the tree structure (such as level 2 in Fig. 1) where distributing neither the fields nor the clusters among the processors is efficient. For such levels, even though distributing the fields eliminates the communication during the translations, dense communication is required elsewhere, i.e. for the interpolation and anterpolation operations during the aggregation and disaggregation stages, respectively [6]. Although such one-to-one data transfers are not problematic for higher levels (such as level 3 in Fig. 1), they become important for lower levels, where the number of processors is comparable to the number of samples. Therefore, even if the numbers of the shared and distributed levels are optimised, sufficient parallelisation efficiency may not be achieved.

In this Letter, we introduce a hierarchical partitioning scheme to further improve the parallelisation efficiency compared to the hybrid approach. This strategy is illustrated in Fig. 1*d*, where the partitioning is performed in both directions (clusters and samples of the fields) for all levels; we adjust the partitioning appropriately by considering the numbers of clusters and the samples of the fields at each level. In the lowest level, the clusters are distributed among the processors without any partitioning for the fields. Then, in the next level (level 2), the samples of the fields are divided between pairs of processors, while we reduce the number of partitions for the clusters by a factor of two. As we proceed to higher levels, the numbers of partitions for the clusters and the fields are systematically decreased and increased, respectively. In this way, the computations for all levels are distributed among the processors with improved load-balancing compared to partitioning with respect to only clusters or only samples of the fields.

With the strategy of partitioning in both dimensions, three different types of communications are required for each level (except for the lowest level) in the hierarchical parallelisation scheme. Consider level 2 in Fig. 1*d*; some of the processors need to communicate during the translations because of the partitioning of the clusters. Similarly, one-to-one communications are required during the aggregation and disaggregation stages owing to the partitioning of the fields. In addition to these, we also need data exchanges among the processors to modify the number of partitions between any two consecutive levels. Although the hierarchical partitioning increases the types of communication compared to the simple and the hybrid approaches, the amount of data transferred is not increased and the number of communication events is reduced. Hence, larger data packages are transferred at fewer times. This improves both communications and the load-balancing significantly.

Results: To demonstrate the improved efficiency of the hierarchical parallelisation, we present the solution of a scattering problem involving a conducting sphere of radius 20 λ discretised with 1 462 854 unknowns. The sphere is illuminated by a plane wave and seven-level MLFMA is used to solve the problem on a cluster of quad-core Intel Xeon 5355 processors connected via an Infiniband network. Fig. 2 shows the efficiency when the solution is parallelised into 2, 4, 8, 16, 32, 64 and 128 processors.

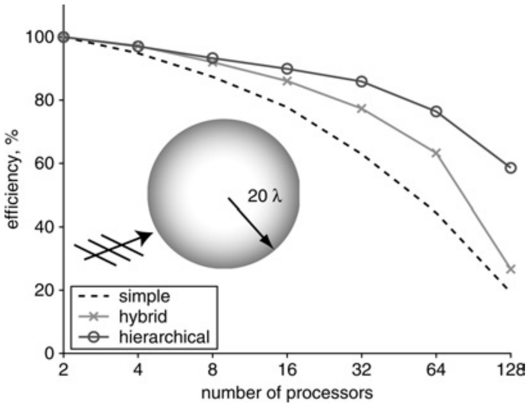

Fig. 2 *Parallelisation efficiency for solution of scattering problem involving sphere of radius 20 λ discretised with 1 462 854 unknowns*

The parallelisation efficiency is defined as

$$\varepsilon_p = \frac{2T_2}{pT_p} \tag{1}$$

where T_p is the processing time of the solution with p processors. Fig. 2 shows that the hierarchical parallelisation improves the efficiency significantly compared to both simple and hybrid parallelisation approaches. All parallelisation schemes are optimised via load-balancing algorithms. Although the hybrid parallelisation, which includes three shared levels, performs better than the simple parallelisation scheme, its efficiency drops below 30% for 128 processors. In this case, the hierarchical parallelisation provides 60% efficiency, which corresponds to 38-fold speed-up compared to the two-processor solution. Using 128 processors and the hierarchical parallelisation scheme, the total processing time, including the setup and the iterative solution with 27 BiCGStab iterations, is only 300 s for this 1.5-million-unknown problem.

Conclusions: Using a hierarchical strategy, the parallelisation efficiency of MLFMA can be improved significantly. Compared to previous approaches based on partitioning in one direction (only clusters or only samples of the fields), hierarchical parallelisation provides higher efficiency, especially when the number of processors is large.

Acknowledgments: This work was supported by the Scientific and Technical Research Council of Turkey (TUBITAK) under Research Grants 105E172 and 107E136, 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.

18 August 2007
Electronics Letters online no: 20082282
doi: 10.1049/el:20082282

Ö. Ergül and L. Gürel (*Computational Electromagnetics Research Center (BiLCEM) and Department of Electrical and Electronics Engineering, Bilkent University, Bilkent, Ankara TR-06800, Turkey*)

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

References

1 Poggio, A.J., and Miller, E.K.: ‘Computer techniques for electromagnetics’ (Pergamon Press, Oxford, UK., 1973, chap. 4)

2 Song, J., Lu, C.-C., and Chew, W.C.: ‘Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects’, *IEEE Trans. Antennas Propag.*, 1997, **45**, (10), pp. 1488–1493

3 Velamparambil, S., Chew, W.C., and Song, J.: ‘10 million unknowns: is it that big?’, *IEEE Antennas Propag. Mag.*, 2003, **45**, (2), pp. 43–58

4 Hastriter, M.L.: ‘A study of MLFMA for large-scale scattering problems’, PhD thesis, University of Illinois at Urbana-Champaign, 2003

5 Sylvand, G.: ‘Performance of a parallel implementation of the FMM for electromagnetics applications’, *Int. J. Numer. Methods Fluids*, 2003, **43**, pp. 865–879

6 Velamparambil, S., and Chew, W.C.: ‘Analysis and performance of a distributed memory multilevel fast multipole algorithm’, *IEEE Trans. Antennas Propag.*, 2005, **53**, (8), pp. 2719–2727

7 Gürel, L., and Ergül, Ö.: ‘Fast and accurate solutions of extremely large integral-equation problems discretised with tens of millions of unknowns’, *Electron. Lett.*, 2007, **43**, (9), pp. 499–500

8 Ergül, Ö., and Gürel, L.: ‘Fast and accurate solutions of large-scale scattering problems with parallel multilevel fast multipole algorithm’. Proc. IEEE Antennas and Propagation Soc. Int. Symp, 2007, pp. 3436–3439

9 Ergül, Ö., and Gürel, L.: ‘Efficient parallelization of multilevel fast multipole algorithm’. Proc. European Conf. on Antennas and Propagation (EuCAP), 2006, 350094