

# Enhancing the Accuracy of the Interpolations and Anterpolations in MLFMA

Özgür Ergül and Levent Gürel

**Abstract**—We present an efficient technique to reduce the interpolation and anterpolation (transpose interpolation) errors in the aggregation and disaggregation processes of the multilevel fast multipole algorithm (MLFMA), which is based on the sampling of the radiated and incoming fields over all possible solid angles, i.e., all directions on the sphere. The fields sampled on the sphere are subject to various operations, such as interpolation, aggregation, translation, disaggregation, anterpolation, and integration. We identify the areas on the sphere, where the highest levels of interpolation errors are encountered. The error is reduced by employing additional samples on such parts of the sphere. Since the interpolation error is propagated and amplified by every level of aggregation, this technique is particularly useful for large problems. The additional costs in the memory and processing time are negligible, and the technique can easily be adapted into the existing implementations of MLFMA.

**Index Terms**—Anterpolation, Lagrange interpolation, multilevel fast multipole algorithm, transpose interpolation.

## I. INTRODUCTION

FOR the iterative solutions of large electromagnetic scattering problems, multilevel fast multipole algorithm (MLFMA) provides acceleration in the processing time and reduction in the memory requirement [1]. Employing MLFMA, complexities of both the peak memory and the number of floating-point operations for a matrix–vector multiplication become  $O(N \log N)$ , where  $N$  is the number of unknowns. Further speedup obtained with the parallelization of MLFMA makes it possible to solve problems with millions of unknowns on clusters of personal computers. MLFMA introduces three extra error sources at the cost of the provided efficiency. These error sources, which are controllable to some extent, arise in addition to others due to the moment methods [2], such as the simultaneous discretization of the geometry and Maxwellian integral equations.

Two of the errors are inherited from the fast multipole method (FMM), namely, the truncation of an infinite series and the angular integration over the unit sphere [3]. The third error stems from the multilevel structure of MLFMA and occurs in the aggregation and disaggregation processes. Due to the nature of the Helmholtz equation, the number of angular samples required to

satisfy a level of accuracy in representing the fields is related to the dimensions of the region containing the sources. Therefore, for the upper levels of MLFMA, where the cluster size is large, finer samplings are required to accurately represent the radiated and incoming fields. As a consequence, during the aggregation and disaggregation processes, interpolation and anterpolation operations are employed, respectively, in order to match the sampling rates between two successive levels. Interpolation and anterpolation constitute the third error source introduced by MLFMA.

As the problem size grows and more levels are required to construct the tree structure of MLFMA, it becomes critical to minimize the interpolation error. This is because the interpolation and anterpolation operations are performed between all consecutive levels and the overall error is accumulated during the aggregation and disaggregation steps. In this letter, we introduce a method to reduce the error around the two poles of the sphere, i.e., the north pole ( $\theta = 0$ ) and the south pole ( $\theta = \pi$ ). This is essential since the error in these regions is usually larger than the error in other regions. Error reduction is achieved by sampling the fields at the poles. Since anterpolation is implemented as the transpose of interpolation [4], enhanced accuracy can be obtained for both of them. Additional costs in the memory usage and processing time are negligible while the interpolation and anterpolation errors are significantly reduced.

The method will be demonstrated within the context of Lagrange interpolation, which is commonly used in MLFMA implementations. In addition to its efficiency, employing the poles can easily be adapted into the existing codes for MLFMA. Details and benefits of the method are reported in Sections III and IV, respectively, following a brief outline of the use of Lagrange interpolation in MLFMA.

## II. LAGRANGE INTERPOLATION

Lagrange interpolation is one of the preferred local-interpolation methods in the aggregation and disaggregation processes of MLFMA. From the local neighborhood of each target point on the fine grid, where the field is to be obtained by interpolation,  $2p \times 2p$  points on the coarse grid are selected. This is illustrated in Fig. 1 for  $p = 2$ , where the values of the field at  $4 \times 4$  points (shaded circles) are employed to compute the value at the target point (star). Let  $f(\theta, \phi)$  represent a scalar field as a function of the spherical coordinates. Then, two-dimensional Lagrange interpolation can be written as

$$\tilde{f}(\theta, \phi) = \sum_{i=s+1-p}^{s+p} w_i(\phi) \sum_{j=t+1-p}^{t+p} v_j(\theta) f(\theta_j, \phi_i) \quad (1)$$

Manuscript received July 24, 2006; revised September 8, 2006. 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 105E172, and by contracts from ASELSAN and SSM.

The authors are with the Department of Electrical and Electronics Engineering Bilkent University, TR-06800, Bilkent, Ankara, Turkey (e-mail: ergul@ee.bilkent.edu.tr; lgurel@bilkent.edu.tr).

Digital Object Identifier 10.1109/LAWP.2006.885010

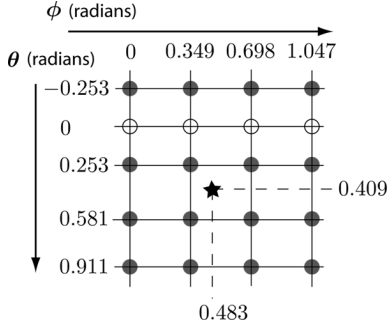


Fig. 1. Lagrange interpolation employing  $4 \times 4$  points (shaded circles) located on the coarse grid to evaluate the function at a point (star) located on the fine grid. Sampling values of  $\theta$  and  $\phi$  are specified in radians and selected from a practical case.

where  $\theta_j$  and  $\phi_i$  are the coordinates of the sampling points on the coarse grid, and  $\check{f}(\theta, \phi)$  represents the value of the field at  $(\theta, \phi)$  perturbed by the interpolation error. In (1),  $s$  and  $t$  are adjusted by the location of the target point, and  $w_i(\phi)$  and  $v_j(\theta)$  represent the interpolation weights derived as

$$w_i(\phi) = \prod_{\substack{k=s+1-p \\ k \neq i}}^{s+p} \frac{\phi - \phi_k}{\phi_i - \phi_k} \quad (2)$$

for the  $\phi$  direction, and

$$v_j(\theta) = \prod_{\substack{l=t+1-p \\ l \neq j}}^{t+p} \frac{\theta - \theta_l}{\theta_j - \theta_l} \quad (3)$$

for the  $\theta$  direction, respectively.

Considering all of the samples in the fine grid, interpolation in (1) can also be expressed as a matrix–vector multiplication, i.e.,

$$\check{\mathbf{F}} = \bar{\mathbf{I}}_{\check{S}S} \cdot \mathbf{F}, \quad (4)$$

where  $\check{\mathbf{F}}$  and  $\mathbf{F}$  are arrays corresponding to the discretization of  $f(\theta, \phi)$  with high and low sampling rates, respectively. In (4),  $\bar{\mathbf{I}}_{\check{S}S}$  represents an  $\check{S} \times S$  interpolation matrix, where  $\check{S}$  and  $S$  are the number of sampling points in the  $\theta$ – $\phi$  space for the fine and coarse grids, respectively. In the case of the Lagrange interpolation,  $\bar{\mathbf{I}}_{\check{S}S}$  becomes sparse and it can easily be stored in the memory to be used multiple times during the aggregation and disaggregation steps of MLFMA.

Anterpolation is required in MLFMA in order to compute angular integrations in the form of

$$I = \int d\hat{\mathbf{k}} f(\hat{\mathbf{k}})g(\hat{\mathbf{k}}) \quad (5)$$

both accurately and efficiently. Assuming that the discretization of  $f(\hat{\mathbf{k}})$  has a higher sampling rate compared to the discretization of  $g(\hat{\mathbf{k}})$ , let  $\check{\mathbf{F}}$  and  $\mathbf{G}$  represent the arrays corresponding to  $f(\hat{\mathbf{k}})$  and  $g(\hat{\mathbf{k}})$ , respectively, and  $\check{\mathbf{W}}$  represents the array of weight coefficients for the integration in the fine grid. An accurate way to calculate the integral in (5) is to interpolate  $\mathbf{G}$  so that

$$I \approx (\check{\mathbf{W}} \star \check{\mathbf{F}}) \cdot (\bar{\mathbf{I}}_{\check{S}S} \cdot \mathbf{G}) \quad (6)$$

where “ $\star$ ” operation represents element-by-element product, i.e.,

$$(\check{\mathbf{W}} \star \check{\mathbf{F}})[n] = \check{\mathbf{W}}[n]\check{\mathbf{F}}[n]. \quad (7)$$

However, (6) contradicts the structure of MLFMA, where the disaggregation is performed downward from top to bottom of the tree. Therefore, the integration in (5) should be performed in the coarse grid via

$$I \approx (\bar{\mathbf{I}}_{\check{S}\check{S}} \cdot \check{\mathbf{F}}) \cdot (\mathbf{W} \star \mathbf{G}) \quad (8)$$

where  $\check{\mathbf{F}}$  is decimated by an interpolation matrix  $\bar{\mathbf{I}}_{\check{S}\check{S}}$  and the integration is evaluated with the aid of weights  $\mathbf{W}$  defined for the coarse grid. The numerical integration in (8) is consistent with the MLFMA structure and it is also more efficient compared to (6) since the integration is computed in the coarse grid with low sampling rate. However, it is less accurate than the integration in (6).

Finally, integration by employing anterpolation is a third way of computing (5) and outperforms both of the methods in (6) and (8). In this case, the transpose of the interpolation matrix in (6) is used to downsample  $\check{\mathbf{F}}$  so that

$$I \approx [\bar{\mathbf{I}}_{\check{S}\check{S}}^T \cdot (\check{\mathbf{W}} \star \check{\mathbf{F}})] \cdot \mathbf{G}. \quad (9)$$

By employing anterpolation, the numerical integral is calculated as accurate as (6) and as efficient as (8). In addition, the integration in (9) is in agreement with the MLFMA structure and convenient for the disaggregation. We also note that the integration weights are also anterpolated in (9).

Since the sampling rates for the fields are determined by considering their spectral contents, we keep the number of the interpolation points  $4p^2$  fixed for the entire aggregation and disaggregation processes. In other words, the same number of interpolation points are employed at each level of MLFMA. Although the functions being interpolated get richer in terms of harmonic content for the upper levels, the sampling rate is also increased so that a fixed number of interpolation points is sufficient to obtain the same level of accuracy in all levels [5].

### III. USING POLES IN LAGRANGE INTERPOLATION

Fig. 1 demonstrates a practical case, where the value of the field is to be obtained at a point (star) with spherical coordinates  $(\theta, \phi) = (0.409, 0.483)$  specified in radians. The figure is related to an aggregation step from a level with a cluster size of  $0.25\lambda$  to the next level with a cluster size of  $0.5\lambda$ . Using the excess bandwidth formula for three digits of FMM accuracy [5], the truncation numbers are 8 and 12 for the lower and the higher levels, respectively. Therefore, the number of samples in the  $\theta(\phi)$  direction is 9 (18) for the coarse grid and 13 (26) for the fine grid. The interpolation parameter,  $p$ , is set to 2 so that  $4 \times 4$  points (shaded circles) are employed in the interpolation. According to the conventional sampling scheme [3], the samples are regularly spaced in the  $\phi$  direction while they are chosen as the Gauss–Legendre points in the  $\theta$  direction.

In Fig. 1, there is only one sample in the  $\theta$  direction on the coarse grid ( $\theta = 0.253$ ) above the target point up to the north

pole ( $\theta = 0$ ). Therefore, four of the required points for the interpolation are provided from the region on the other side of the pole. Considering the next sample in the decreasing  $\theta$  direction, these are the points with  $\theta = -0.253$ .<sup>1</sup> Although this is the best choice, there exists a wide gap in  $\theta$  direction from  $\theta = -0.253$  to  $\theta = 0.253$ . These wide gaps created near the poles in all levels lead to larger interpolation errors compared to the other regions far from the poles.

To reduce the interpolation error described above, we employ the poles by sampling the fields at  $\theta = 0$  and  $\theta = \pi$ . Although the radiated and incoming fields in MLFMA are vectors with two spherical components represented as

$$\mathbf{f}_{r/i}(\theta, \phi) = \hat{\theta}\hat{\theta} \cdot \mathbf{f}_{r/i}(\theta, \phi) + \hat{\phi}\hat{\phi} \cdot \mathbf{f}_{r/i}(\theta, \phi) \quad (10)$$

we evaluate and store the fields at the poles in the  $x$  and  $y$  directions. As an example, for the north pole,  $\theta$  and  $\phi$  components are extracted as

$$\hat{\theta} \cdot \mathbf{f}_{r/i}(0, \phi) = \left[ \cos \phi f_{r/i}^x(\theta = 0) + \sin \phi f_{r/i}^y(\theta = 0) \right] \quad (11)$$

and

$$\hat{\phi} \cdot \mathbf{f}_{r/i}(0, \phi) = \left[ \cos \phi f_{r/i}^y(\theta = 0) - \sin \phi f_{r/i}^x(\theta = 0) \right] \quad (12)$$

whenever required for the interpolation. Consequently, independent of the value of  $\phi$ , all the interpolations performed near the poles are improved by this technique using (11) and (12) without having to store the values of  $\mathbf{f}_{r/i}(\theta = 0, \phi)$  for each sample of  $\phi$ . This is illustrated in Fig. 1, where the samples computed with (11) and (12) are represented by circles located at  $\theta = 0$ .

To include the calculations related to the poles, four more complex numbers are required (two for each pole) for each basis and testing functions. The extra memory and the increase in the processing time are negligible compared to the base requirements, since the data size for each basis and testing function is raised from  $4(L+1)^2$  to  $4(L+1)^2 + 4$ , where  $L$  is at least 7 for two digits of accuracy. During the aggregation and disaggregation processes, the values at the poles are calculated and stored for each cluster to improve the interpolation and anterpolation at all levels. However, the poles do not contribute to the angular integration and they do not have effect on the two errors inherited from FMM.

#### IV. NUMERICAL EXAMPLE

As a numerical example, Fig. 2 depicts the relative interpolation error related to a basis cluster  $C_1$  with the size of  $2\lambda$  in the fourth level from the bottom of the tree structure. For the field of the cluster, the number of angular samples in the  $\theta$  and  $\phi$  directions are 33 and 66, respectively, determined by the excess bandwidth formula for three digits of FMM accuracy. By the row-wise arrangement of the  $\theta$ - $\phi$  space, the interpolation error is plotted with respect to the samples. Only the error in the  $\theta$  component of the field, which is the dominant component in this example, is plotted. The relative interpolation error is defined as

$$e_n = \frac{|\tilde{f}(\theta_n, \phi_n) - f(\theta_n, \phi_n)|}{\max_{(\theta_j, \phi_j)} |f(\theta_j, \phi_j)|} \quad (13)$$

<sup>1</sup>Although a negative value of  $\theta$  is shown here, the actual locations are determined by  $(-\theta, \phi) = (\theta, \phi + \pi)$ .

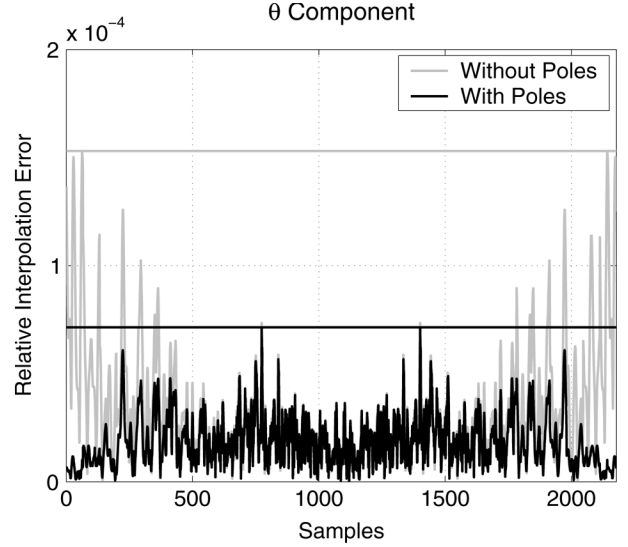


Fig. 2. Relative interpolation error defined in (13) with respect to the samples on a  $33 \times 66$  grid converted into one-dimensional data by a row-wise arrangement of the  $\theta$ - $\phi$  space. The reference data is obtained without interpolation. To obtain the interpolated data, aggregation is performed from the lowest (first) level to the fourth level by employing interpolations with (black) and without (gray) poles.

where  $f$  is the exact field,  $\tilde{f}$  is the perturbed field obtained via the interpolations, and  $n = 1, 2, \dots, 2178$  is the sample index. The exact data in (13) is calculated by evaluating the field for each basis function inside the cube with the sample rate defined for the fourth level so that it is obtained without any interpolation. However, in the case of the perturbed data, the fields of the basis functions are sampled according to the smallest box size, which is  $0.25\lambda$ . Then, three aggregation steps are performed from the lowest (first) level to the fourth level. Consequently, the perturbed data is the practical case, where the interpolation error is introduced at three passages between the levels, i.e., from first to second, second to third, and third to fourth.

In Fig. 2, the interpolation error is plotted when the poles are not employed as in the conventional case (gray) and when they are employed as suggested in this letter (black). For a clear comparison, the maximum errors are also indicated in the plot with horizontal lines. Employing the poles reduces the errors and the maximum relative error is reduced approximately to its half from  $1.530 \times 10^{-4}$  to  $7.147 \times 10^{-5}$ . Other numerical experiments also reveal similar results.

In Fig. 2, the reduction in the error is clearly visible at the two ends of the horizontal axis corresponding to the points located near the poles. However, the improvement extends beyond these narrow polar regions toward the middle of the horizontal axis corresponding to the points located around the equator. This is due to the fact that an interpolation error made in the first steps of the aggregation is propagated toward the equator region in the next steps. Consequently, the use of the poles improves the interpolation accuracy also for the points located far from the poles. In general, improvements obtained by adding the poles in the Lagrange interpolation become more significant as the problem size grows and more levels are required in the MLFMA tree.

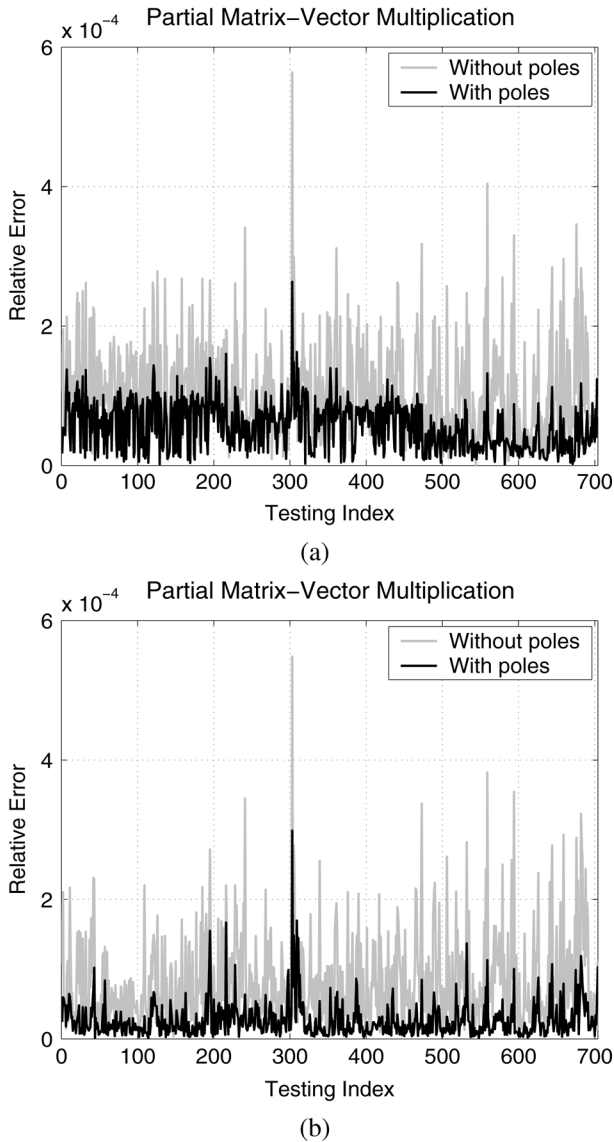


Fig. 3. Relative errors defined in (14) in the partial matrix–vector multiplications related to two clusters  $C_1$  and  $C_2$  in the fourth level with respect to the index of the testing function located in  $C_2$ . The reference data is obtained without interpolation and anterpolation. (a) Relative error when interpolation and anterpolation are employed. (b) Relative error when interpolation is eliminated and the error is only due to the anterpolation.

Demonstrating of the anterpolation error is not as straightforward as the interpolation error since the anterpolated function is different from the original function. Therefore, to prove that employing poles also increases the accuracy of the anterpolation in MLFMA, we present in Fig. 3 the errors in the partial matrix–vector multiplications. After translating the radiation of  $C_1$  into an incoming wave for a testing cluster  $C_2$ , we perform the disaggregation steps from the fourth level to the first level. This way, as presented in Fig. 3(a), we obtain the error in the partial matrix–vector multiplication related to the clusters  $C_1$  and  $C_2$  with respect to the index of the testing functions in  $C_2$ . Similar to (13), a relative error is defined as

$$e_n = \frac{|\tilde{v}_n - v_n|}{\max_j |v_j|} \quad (14)$$

where

$$v_n = \sum_{m=1}^{N_{C_1}} Z_{nm} a_m \quad (n = 1, 2, \dots, N_{C_2}) \quad (15)$$

is the result of the partial multiplication without interpolation and anterpolation errors. This reference data is obtained by evaluating the radiation and receiving patterns of the basis and testing functions inside the clusters  $C_1$  and  $C_2$ , respectively, with the sample rate defined for the fourth level so that interpolation and anterpolation are not involved. In (15),  $a_m$  represents the coefficients of the basis functions inside  $C_1$  for  $m = 1, 2, \dots, N_{C_1}$ ,  $Z_{nm}$  represents the matrix elements calculated by MLFMA, and  $n$  is the testing index from 1 to  $N_{C_2} = 704$ . The perturbed data in (14), i.e.,

$$\tilde{v}_n = \sum_{m=1}^{N_{C_1}} \tilde{Z}_{nm} a_m \quad (n = 1, 2, \dots, N_{C_2}) \quad (16)$$

is obtained by performing the usual aggregation and disaggregation steps so that the values in Fig. 3(a) contain both interpolation and anterpolation errors. Fig. 3(a) displays a significant improvement in the accuracy obtained by using the poles. Next, we eliminate the interpolation error by computing the incoming waves to the center of the testing cluster  $C_2$  without employing interpolation. The result is depicted in Fig. 3(b), where the error in the partial matrix–vector multiplication is due to only the anterpolation. Therefore, Fig. 3(b) clearly demonstrates that employing the poles enhances the accuracy of the anterpolations, similar to the improvement shown in Fig. 2 for interpolations.

## V. CONCLUSION

In this letter, we present an efficient technique to reduce the interpolation and anterpolation errors in MLFMA. Interpolation errors are significantly decreased for the points located near the poles of the unit sphere, where the errors are generally large in the conventional sampling scheme. By using the same technique, anterpolation errors are also reduced and the matrix–vector multiplications become more accurate.

## REFERENCES

- [1] C.-C. Lu and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," *IEEE Trans. Antennas Propagat.*, vol. 45, no. 10, pp. 1488–1493, Oct. 1997.
- [2] R. F. Harrington, *Field Computation by Moment Methods*. New York: Macmillan, 1968.
- [3] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription," *IEEE Antennas Propagat. Mag.*, vol. 35, no. 3, pp. 7–12, Jun. 1993.
- [4] A. Brandt, "Multilevel computations of integral transforms and particle interactions with oscillatory kernels," *Comp. Phys. Commun.*, vol. 65, pp. 24–38, Apr. 1991.
- [5] S. Koç, 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.