A GRID Computer Implementation of the Multilevel Fast Multipole Algorithm for Full-Wave Analysis of Optical Devices

Jan FOSTIER^{†a)}, Nonmember and Femke OLYSLAGER[†], Member

SUMMARY We present a parallel multilevel fast multipole algorithm aimed at low cost parallel computers such as GRID computer environments and clusters of workstations. The algorithm is a scheduling algorithm where work packets are handled in a certain order to ensure minimal idle time of the processors and to avoid simultaneous bursts of communication between the processors. The algorithm is implemented on a method of moment discretization of a two-dimensional TM electromagnetic scattering problem. Examples of several optical devices with a size up to 28 500 wavelengths are presented.

key words: parallel MLFMA, electromagnetic scattering, GRID computing

1. Introduction

The Multilevel Fast Multipole Algorithm (MLFMA) has gained considerable interest during the past decade to solve very large electromagnetic scattering problems using surface integral equation techniques. For an introduction we refer the reader to [1]. The application of the Method of Moments (MoM) technique to discretize an integral equation leads to a dense system of linear equations of dimension N by N where N denotes the number of discretizations. An iterative solution of this system where the matrix-vector products are evaluated with the MLFMA allows for a solution with a time complexity of $O(N \log N)$ and a similar memory complexity. If the number of iterations can be limited then the MLFMA allows for the solution of very large problems with limited computer resources.

The close to linear increase in complexity as a function of the number of unknowns means that the size of the problems that can be handled increases significantly when sufficient computational resources are available. The GRID technology allows for an easy and affordable construction of a parallel computer system with distributed memory. Parallelization of electromagnetic simulation techniques has been a subject of considerable research in the past. However, it gained momentum after the introduction of GRID computing since affordable and very versatile parallel systems became available. For example the book [2] introduces the reader to GRID computing and is essentially focused on the Finite Difference Time Domain technique. GRID computing facilities come in very different configurations, ranging from a large set of computers widely distributed and

connected through the 'slow' internet or a cluster of similar computers connected through a 'fast' switch. In each case the computers only have fast access to their local memory and rely on the message passing paradigm for inter-processor communication. It is obvious that this communication between processors is an important aspect. The less communication is needed the higher the performance of the algorithm, this becomes an essential issue when considering a GRID facility depending on internet communication.

Parallel MLFMA has already gained some attention, we refer e.g. to [3]–[6]. In this contribution, which significantly extends the conference contribution [7], we will present a parallel MLFMA aimed at GRID computing facilities. The algorithm is based on a heuristic that tries to reduce idle time of processors and that tries to avoid overloading the switch by distributing communication between processors in time. All stages within the algorithm (the setup stage, the matrix-vector product and the output calculation) are fully parallelized.

Although the algorithm does not make any specific assumptions about the dimensionality of the problem we will apply it to a two-dimensional TM electromagnetic scattering problem. This problem is solved using a boundary integral equation based on Huygens principle that has already been developed in [8]. The far interactions in the resulting MoM matrix are evaluated with the MLFMA and the near interactions are evaluated extremely accurate by not only evaluating the self-patch contributions analytically but also the neighbor-patches. Another application of this algorithm, although not parallelized, has been shown in [9]. Since we aim at simulating very large structures in terms of wavelengths, a high-frequency MLFMA is used. To accelerate the setup stage of the near interactions, we will exploit all symmetries in the structure using a symmetry extracting algorithm based on splay trees [10]. In this paper, with specific optical applications in mind, we will suffice with a block-Jacobi preconditioner.

We present a number of full-wave optical simulations: a multi-layered dielectric mirror with up to seven layers, a lens system with coatings and a very large Fresnel lens with a size of 28 500 wavelengths, modelled with more than 3 million unknowns and simulated on a parallel GRID system with up to 12 processors.

First the boundary integral equation and its MoM discretization are introduced. Then we will discuss the MLFMA after which the heuristic used for the parallelization is explained. To validate the parallel code, the bistatic

Manuscript received December 26, 2006.

Manuscript revised March 26, 2007.

[†]The authors are with the Department of Information Technology (INTEC), Faculty of Engineering, Ghent University, Ghent, Belgium.

a) E-mail: jan.fostier@intec.ugent.be

DOI: 10.1093/ietcom/e90-b.9.2430

radar cross section for a number of canonical examples is compared to its analytical solution. Finally, some examples will demonstrate the performance of the algorithm.

2. Method of Moments

Consider a number of cylindrical PEC and dielectric objects with material parameters ϵ_i and μ_i of arbitrary shape, parallel to the x -axis, embedded in free space, that are illuminated by a TM incoming electromagnetic field e_x^i, h_t^i (see Fig. 1). Each of these objects can be embedded into other dielectrics. The subscript ' t ' indicates vectors in the yz -plane. If an $e^{j\omega t}$ time dependence is suppressed, the boundary integral equation for a single object can be written as [8]

$$\begin{aligned} e_x^i - \lim_{r \rightarrow C^+} \oint_C \left[e_x \frac{\partial G_0}{\partial n'} - \frac{jk_0^2}{\omega \epsilon_0} G_0 h_t \right] dc' \\ = \lim_{r \rightarrow C^-} \oint_C \left[e_x \frac{\partial G}{\partial n'} - \frac{jk^2}{\omega \epsilon} G h_t \right] dc' \end{aligned} \quad (1)$$

$$\begin{aligned} h_t^i - \lim_{r \rightarrow C^+} \oint_C \left[-\frac{j\omega \epsilon_0}{k_0^2} e_x \frac{\partial^2 G_0}{\partial n \partial n'} - \frac{\partial G_0}{\partial n} h_t \right] dc' \\ = \lim_{r \rightarrow C^-} \oint_C \left[-\frac{j\omega \epsilon}{k^2} e_x \frac{\partial^2 G}{\partial n \partial n'} - \frac{\partial G}{\partial n} h_t \right] dc' \end{aligned} \quad (2)$$

with ϵ and μ the material parameters of that object, C its boundary curve, $k^2 = \omega^2 \epsilon \mu$ and where the two dimensional Green function is given by

$$G(\mathbf{r}|\mathbf{r}') = \frac{j}{4} H_0^{(2)}(k|\mathbf{r} - \mathbf{r}'|), \quad (3)$$

and similarly for k_0^2 and G_0 . C^- and C^+ denote that the contour C is approached from the inside and outside respectively. The unknowns are the tangential electric field e_x and the tangential magnetic field h_t to the contour C . The e_x and h_t unknowns can be seen as equivalent magnetic and electric currents on the boundary C . The magnetic currents are flowing along the boundary C in the yz -plane and the electric currents are longitudinal currents in the x -direction. Equation (1) imposes that the electric field, generated by the contributing sources of the outside and inside medium the object is continuous, taking into account a contribution from an incident field for the background medium. Similarly (2)

expresses the continuity of the tangential component of the magnetic field. One might argue that this way of working is not very economical. Indeed, one could use e.g. an electric field integral equation, reducing the number of unknowns by one half. However, we opt for the more expensive approach (1) and (2) since our experience showed that it leads to better conditioned systems.

For a PEC body only the first equation and the h_t unknown remain, i.e.

$$e_x^i + \frac{jk_0^2}{\omega \epsilon_0} \lim_{r \rightarrow C^+} \oint_C G_0 h_t dc' = 0 \quad (4)$$

The extension of the integral formulations (1), (2) and (4) to multiple objects is achieved by taking the summation over the contributing objects to the fields of the inner (only for dielectrics) and outer medium respectively. In particular, we also consider objects embedded into other objects.

The contour C is divided into a number of segments on which pulse basis functions and overlapping triangular basis functions are defined. The pulse basis functions are used to expand h_t and as test functions for (1) and (4). Conversely the triangular basis functions are used to expand e_x and as test functions for (2). In this way a consistent Galerkin MoM is obtained (see also [8]).

In order to obtain a high accuracy the logarithmic singular part is extracted from the Green function in some cases. The basis and test function integrations for this logarithmic part are evaluated analytically for self-patch integrations (i.e. the basis and test functions are defined over the same segment) and for neighbor-patch integrations (i.e. the basis and test functions are defined over adjacent segments). Especially the last ones require tedious analytical calculations. We extract the following singular parts

$$[G(\mathbf{r}|\mathbf{r}')]_{\text{sing}} = \frac{1}{2\pi} \ln(k|\mathbf{r} - \mathbf{r}'|) \quad (5)$$

$$\left[\frac{\partial}{\partial n'} G(\mathbf{r}|\mathbf{r}') \right]_{\text{sing}} = -\frac{1}{2\pi} \frac{\mathbf{n}' \cdot (\mathbf{r} - \mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|^2} \quad (6)$$

$$\begin{aligned} \left[\frac{\partial^2}{\partial n \partial n'} G(\mathbf{r}|\mathbf{r}') \right]_{\text{sing}} &= \frac{k^2}{4\pi} (\mathbf{n} \cdot \mathbf{n}') \ln(k|\mathbf{r} - \mathbf{r}'|) \\ &+ \frac{1}{\pi} \left[\frac{(\mathbf{n} \cdot (\mathbf{r} - \mathbf{r}'))(\mathbf{n}' \cdot (\mathbf{r} - \mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|^4} - \frac{\mathbf{n} \cdot \mathbf{n}'}{2|\mathbf{r} - \mathbf{r}'|^2} \right] \end{aligned} \quad (7)$$

As an example of a neighbor-patch integral let us consider the interaction between two adjacent pulse basis and pulse test functions originating from the h_t unknowns in Eq. (1). The extracted singularity is (5) and the situation is shown in Fig. 2. With the notations of Fig. 2 this contribution is given by

$$\begin{aligned} \frac{1}{2\pi d_1 d_2} \int_0^{d_2} \int_0^{d_1} \left[\ln(k) + \ln(\sqrt{s^2 + s'^2 - 2ss' \cos \phi}) \right] ds' ds \\ = \frac{1}{2\pi} [\ln(k) + f(d_2, d_1) + f(d_1, d_2)] \end{aligned} \quad (8)$$

with

Fig. 1 Objects illuminated by an incoming electromagnetic field.

Fig. 2 Neighbor-patch interaction between two pulse segments.

$$f(d_1, d_2) = \frac{1}{4} \left(1 - \frac{d_1 \cos \phi}{d_2} \right) \ln(d_1^2 + d_2^2 - 2d_1 d_2 \cos \phi) + \frac{d_1 \sin \phi}{d_2} \left[\operatorname{atan} \left(\frac{d_2 - d_1 \cos \phi}{d_1 \sin \phi} \right) + \operatorname{atan}(\cot \phi) \right] - \frac{3}{4} + \frac{1}{2} \frac{d_1}{d_2} \ln(d_1) \cos \phi \quad (9)$$

The limit for $\phi = \pi$ is given by

$$\lim_{\phi \rightarrow \pi} f(d_1, d_2) = \frac{1}{2} \left(1 + \frac{d_1}{d_2} \right) \ln(d_1 + d_2) - \frac{3}{4} - \frac{d_1}{2d_2} \ln(d_1) \quad (10)$$

Note that the amplitude of a pulse function is taken equal to the inverse of the length of the segment on which it is defined. The expressions emerging from the other singularities are still far more complicated but all can be derived in closed form.

The remaining parts of the Green function and its derivatives, after subtracting the singular parts (5)–(7), are calculated using a polynomial approximation in the neighborhood of the origin. The basis and test function integrations with these remaining parts are evaluated using Gaussian quadrature rules. Also the non self- or neighbor-patch integrations are evaluated numerically using Gaussian quadrature rules.

Finally, the MoM results in a linear system of unknowns

$$Z \cdot X = B \quad (11)$$

with X a vector containing the unknown expansion coefficients of e_x and h_t in triangular and pulse basis functions respectively. Z is the interaction matrix and B is a vector representing the tested incoming fields. This system is solved iteratively using the TFQMR algorithm [11].

We note that for the simulation of large-scale problems, the accuracy of the matrix-elements is fundamental. In order to limit the number of iterations, analytical treatment of the self- and neighbor-patches becomes indispensable.

3. Near Interactions — Splay Trees

Often large structures contain symmetries where two pairs of interacting segments are geometrically equal. This obviously means that the corresponding two elements in the interaction matrix Z are equal. Computing time during the set-up stage can be saved when these symmetries are recognized, because the corresponding interactions only need to be calculated once. If one has N segments then there are

Table 1 Comparison between near interaction setup times with and without splay tree (ST).

Case	Unique interactions	No ST	With ST
Circle	0.38%	41.23s	0.57s
Square	10.92%	64.10s	7.78s
Hyp. lens	86.38%	61.03s	52.93s

N^2 interactions. A brute force method comparing all these interactions to each other would lead to $O(N^4)$ computing time, which obviously would jeopardize the whole algorithm. Even if one restricts symmetry extraction to the near interactions this would still lead to $O(N^2)$ computing time overwhelming the $O(N \log N)$ complexity of the MLFMA. To avoid this we extract symmetry for the near interactions using a splay tree [10]. A splay tree is a special kind of self-balancing binary tree with an amortized complexity of $O(\log N)$ for store and search operations. In contrary to other self-organizing data structures such as the red-black tree or the AVL tree, it does not require any additional meta-data to keep track of its balancing, which makes it memory efficient. Furthermore, frequently accessed elements are stored closer to the tree root, making it very fruitful for implementing caches.

Whenever an interaction between two segments needs to be evaluated a search operation is performed on the splay tree. If the interaction is not yet stored in the tree, it is evaluated and stored so it can be reused for future geometrically equal interactions. Because the time needed for a search or store operation is much smaller than the time needed for numerical integration over the segments, this yields an efficient way of working. This is illustrated in Table 1 where the setup time for the near interaction of several simple shapes is given. Each shape is modelled with approximately 500 unknowns. Even for a hyperbolic-plano lens, where most interactions are unique, we see some savings, indicating that the overhead imposed by unsuccessful search operations is limited. Note that in the situation where all interactions were unique, the time complexity for building a splay tree would be $O(N \log N)$.

The splay tree will require somewhat more memory, but this memory needs to be allocated only temporary, after which it can be reused to store e.g. the translation operators in the MLFMA.

4. Far Interactions — MLFMA

The number of unknowns in a scattering problem can be high for two reasons. First the structure can be large measured in wavelengths and at least 10 to 30 unknowns per wavelength are needed depending on the required accuracy. Second the geometry could contain small geometrical details requiring a fine discretization to resolve. The high-frequency MLFMA, based on plane wave expansions, is especially suited to tackle structures that are many wavelengths in size. On the other hand a multipole based low-frequency MLFMA is suited for small but complex struc-

Fig. 3 Two interaction groups in the MLFMA.

tures with many unknowns per wavelength. Both techniques can be combined. However, in our approach we mainly focus on problems that are (extremely) large compared to wavelength motivating the high-frequency MLFMA.

The MLFMA is a fast technique to evaluate the fields in a set of N points due to N sources located in those same points. We will not go into details of this technique and restrict ourselves to the main characteristics (for a detailed account see e.g. [1]). The segments are grouped into a number of groups and instead of calculating the individual interactions between each two segments the interactions are calculated groupwise. This is illustrated in Fig. 3 and mathematically expressed through the following expansion of the Green function

$$H_0^{(2)}(k|\mathbf{r}_o - \mathbf{r}_s|) \approx \sum_{n=-Q}^{n=+Q} \left[e^{j\mathbf{k}_n \cdot (\mathbf{r}_s - \mathbf{R}_s)} T_n(\mathbf{R}_o - \mathbf{R}_s) e^{-j\mathbf{k}_n \cdot (\mathbf{r}_o - \mathbf{R}_o)} \right] \quad (12)$$

where $\mathbf{R}_s$ and $\mathbf{R}_o$ respectively are the centers of the source and the observation group and where $\mathbf{r}_s$ and $\mathbf{r}_o$ are two arbitrary points on the segments in the source and observation groups. $\mathbf{k}_n$ are wavevectors along equidistant directions ϕ_n , $n = -Q, \dots, 0, \dots, Q$ and T_n is the translation operator given by

$$T_n(\mathbf{R}) = \frac{1}{2Q+1} \sum_{n'=-Q}^{n'=+Q} H_{n'}^{(2)}(k|\mathbf{R}|) e^{jn'(\Phi - \phi_n - \frac{\pi}{2})} \quad (13)$$

with Φ the angle between $\mathbf{R}$ and the y -axis.

The first part (the aggregation) in (12) depends only on the segments in the source group and is an expansion of plane waves along directions ϕ_n . The second part depends only on the centers of the observation and the source group and represents the translation of plane waves. The third part (the disaggregation) depends only on the segments of the observation group and is an aggregation of incoming plane waves.

In the MLFMA this is extrapolated hierarchically by grouping groups into groups and so on. In that way all the

interactions, normally requiring a computational complexity of $O(N^2)$ can be reduced to $O(N \log N)$.

5. Parallelization Heuristic

The MLFMA divides the whole structure into a grid of hierarchic squares. Using a Morton or Hilbert space filling curve these squares are allocated to the different processors such that near squares are as much as possible allocated to the same processor. Of course, the geometries under consideration will encompass many empty squares that are eliminated. The load over the processors is balanced by dividing the space filling curve at a certain level such that each processor has an approximately equal amount of work to perform during each iteration step. This is a rather straightforward procedure.

The workload is divided in small packets, involving the calculation of near interactions, aggregations, translations or disaggregations. These packets are stacked in a certain order. The main principle is to handle first those packets that might result in data that another processor might need. If for a certain packet information from another processor is needed that is not yet available then it will be rescheduled to be handled later. As soon as information is available for another processor that information will be made available to be sent to the other processor and will be sent as soon as the other processor is ready to accept it. Near interactions usually require no communication between processors and are postponed as long as possible to a moment where the processor has nothing else to do, hence near interactions get lowest priority.

This heuristic has the advantage that communication does not come into bursts which would overload the interconnection network and which would result in waiting cycles in the processors. The overlap between communication and computation is maximized, making this way of working far more efficient for parallel computers with a slower interconnection network. Of course defining and scheduling all the work packets is a complex task.

Also the set-up stage is parallelized. Only the geometry local to each processor is stored to obtain good memory efficiency. The splay tree is divided over the processors, each having its own tree of a smaller size than the total tree. Because the symmetry in the geometry is usually of a local nature, these smaller splay trees are often more performant.

6. Numerical Examples

All simulations were performed on a system consisting of three machines with two AMD 64 bit Opteron 270 dual core processors each (4 cores per machine). This accounts for 12 cores running at 2 GHz with 2 GByte of memory each. A standard gigabit Ethernet switch was used to interconnect the machines. Note that the connection between a machine and the switch is shared between four cores. Possible fast communication between the cores in the same machine was not exploited. In what follows, no further distinc-

tion between the term *core* and *processor* is made and only the term *processor* is used. Parallelization and communication was implemented using the Message Passing Interface (MPI) [12]. Performance was optimized with the BLAS [13] wherever possible.

6.1 Canonical Examples

To validate the parallel implementation, we compare the calculated bistatic radar cross-section (RCS) for 4 different geometries with their analytical solution. First we consider a single dielectric ($\epsilon_r = 2$) cylinder (case 1) and a single PEC cylinder (case 2), both with a diameter of 1000λ . To test multi-object geometries, we consider a PEC cylinder embedded into a dielectric ($\epsilon_r = 2$) cylinder (case 3, see Fig. 4) and a dielectric ($\epsilon_r = 4$) cylinder embedded into another dielectric ($\epsilon_r = 2$) cylinder (case 4). In both cases, the outer cylinder has a radius of 1000λ while the inner cylinder has a radius of 500λ . In all four cases, the analytical solution can be easily derived in closed form. We determine the root mean square error for all cases as follows:

$$\text{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^N |\sigma_a(\theta_n) - \sigma_c(\theta_n)|} \quad (14)$$

with $\sigma_a(\theta_n)$ and $\sigma_c(\theta_n)$ the analytical and calculated RCS respectively, evaluated at equidistant angles $\theta_n = \frac{2\pi n}{N}$. The TFQMR solver was stopped when the relative residual error dropped below 10^{-5} . A $16\lambda \times 16\lambda$ preconditioner was used in all cases. The simulations were repeated using 1 up to 12 processors with identical results. Table 2 shows the RMS error for all four cases.

Figure 5 shows the full RCS for the geometry of Fig. 4 (case 3) and two more detailed RCS around $\theta = 0^\circ$ and $\theta = 60^\circ$. We conclude that we can obtain very low errors.

Fig. 4 Geometry for the analytical validation (case 3).

Table 2 The RMS error for the four canonical geometries.

	# unknowns	# iterations	RMS (dB)
case 1	31 416	6 512	5.43E-02
case 2	88 858	3 716	1.23E-02
case 3	111 073	11 109	1.63E-02
case 4	151 690	7 270	4.04E-02

6.2 Dielectric Mirror

As a first optical example we consider a dielectric mirror with a diameter of 1000λ as depicted in Fig. 6.

Fig. 5 Bistatic RCS for a 500λ PEC cylinder embedded into a 1000λ dielectric cylinder: $\theta \in [0^\circ, 180^\circ]$ (top), $\theta \in [0^\circ, 1^\circ]$ (middle), $\theta \in [60^\circ, 61^\circ]$ (bottom).

Fig. 6 Geometry for the dielectric mirror.

The mirror consists of alternating layers with a higher and a lower refractive index which can be modelled by embedding dielectric objects into each other. The mirror is curved with a radius of 1000λ . The mirror is illuminated by a Gaussian beam with a waist of 300λ at the center of the lens.

The refractive indices are borrowed from a real-life example, i.e. a combination of zinc sulfide (ZnS , $n_H = 2.32$) and magnesium fluoride (MgF_2 , $n_L = 1.38$). If the thickness d of the layers is chosen such that $n_L d_L = n_H d_H = \lambda_0/4$, constructive interference will occur between the reflections at the interfaces. When more layers are used, the total reflection increases. The ZnS/MgF_2 dielectric mirror is sometimes used in laser cavities. In that case, tens of layers are used to obtain nearly perfect reflection.

We considered two cases: a 3-layer mirror (modelled with two dielectric objects) and a 7-layer mirror (modelled with four dielectrics recursively embedded into each other). The number of unknowns is 80 060 for the 3-layer mirror and 160 224 for the 7-layer mirror. Both situations were simulated in parallel on four processors. Due to the fact that the layers are closely stacked onto each other these simulations contain many near interactions. Because of the circular shape of the layers however, many of these near interactions are equivalent and can be reused. This results in a set-up time of only 121s and 212s respectively when the splay tree is used.

The number of iterations required for solving the 3-layer mirror was 23 106 and 25 566 for the 7-layer mirror when the precision for the iterative TFQMR solver was set to 10^{-3} . The total run time was 2.4h and 7.1h respectively. This difference can be explained due to the fact that the near interactions require four times more computational time in the second case because of the close packing of the unknowns. When the number of iterations are compared to that for a lens medium [7] of equal size, one can observe that they are significantly higher. This is a manifestation of resonant behaviour, which occurs between the coatings, all spaced $\lambda/4$ from each other, making this example ill-conditioned. By using a block-Jacobi preconditioner with block sizes of $1\lambda \times 1\lambda$ some of these resonances can be captured. Without preconditioner, the number of iterations would become very high ($> 100\,000$), making the solution of such resonant structures impractical.

A comparison between the 3-layer and the 7-layer dielectric mirror is shown in Fig. 7 where the electric field is plotted along a line through the axis of the mirror. One can clearly notice the difference between the amount of energy that propagates through the mirror. By using more layers this could be further reduced. However, the computational cost by using more layers would rise quickly due to the increasing number of near interactions while the far interaction costs would remain approximately the same. Also note the continuity of the field at the interfaces.

In Fig. 10 a density plot of the electric field amplitude is shown for the 7-layered case. One can observe the reflection of the incident Gaussian beam which is focused by the curved mirror. A small amount of energy propagates

Fig. 7 Electric field along a line through the axis of the dielectric mirror.

Table 3 Simulation of the lens system on 8 processors.

Case	N	Memory	# iter.	time/iter.
Uncoated	984 292	3.78 GB	8 230	6.02s
Coated	1 420 292	11,72 GB	24 748	13.71s

through the mirror. The density plot for the 3-layered case is very similar.

6.3 Lens System

As a second and much larger example we consider a lens system consisting of two circular lenses and an PEC aperture. The size of the lenses is $2\,000\lambda$ and $5\,000\lambda$ respectively and they are $30\,000\lambda$ spaced apart. The PEC shields are placed $20\,000\lambda$ from the leftmost lens and the size of the central gap is $2\,500\lambda$. The leftmost lens is illuminated with a Gaussian beam with a width of 500λ . We consider both the cases where the two lenses are uncoated, and where the lenses are coated with an anti-reflective $\lambda/4$ coating (see [7] for another coated lens example). Field density plots for both cases are shown in Figs. 8 and 9. Note that the reflection at both lenses has almost completely disappeared. Also note that since the density plots comprise a region of $40\,000\lambda$ width, so-called Moiré-patterns occur, because each pixel corresponds with an area of approximately $10\lambda \times 10\lambda$. In [14] these simulations (at a lower frequency however) were compared to a 1.5D gauss bundel propagation technique and showed good correspondence.

For the simulation of the lens system without coatings 984 292 unknowns were required, while 1 420 292 unknowns were used when the coatings were introduced. Both simulations were run on 8 processors. Other data is given in Table 3. Note that the introduction of the anti-reflective coatings required four times more memory for storing the near interactions as compared to the simulation without coatings. Also, due to memory constraints, a smaller preconditioner was used for the coated lens system. This explains the higher number of required iterations. However, note that after 10 000 iterations, the precision in the iterative solver was already as low as $3.99 \cdot 10^{-3}$ after which it

Fig. 8 Field density $|e_x|$ (V/m) in the lens system with uncoated lenses.

Fig. 9 Field density $|e_x|$ (V/m) in the lens system with coated lenses.

Fig. 10 Field density $|e_x|$ (V/m) of the dielectric mirror.

Fig. 11 Field density $|e_x|$ (V/m) in the Fresnel lens.

required another 14 000 iterations to reach the tolerance of 10^{-3} . It is our experience that solving for an extra digit of precision requires lots of extra computational resources, an observation also confirmed by other research groups.

6.4 Fresnel Lens

Finally, we consider a large-scale simulation of a Fresnel lens with a diameter of 28 500 wavelengths. This corresponds with a lens of approximately 1 cm in diameter simulated at an optical frequency of $\lambda = 350$ nm (blue light). The Fresnel lens, depicted in Fig. 12 consists of several segments, which are translated towards the axis of the lens, making the lens much thinner than conventional lenses. The

Fresnel lens is equivalent to a spherical lens with a curvature radius of 1 cm. With a relative permittivity $\epsilon_r = 4$, this corresponds to a focal distance of 0.5 cm.

The lens, which consist of 3 044 052 unknowns was simulated on 12 processors and required almost 24 GByte of memory. As lens objects are well-conditioned, only 1 066 iterations were required to solve the problem to a precision of 10^{-5} in the iterative solver. The total solve time was 3.8h and it required only 73s to evaluate the field in 2048×2048 points to create Fig. 11.

6.5 Parallel Efficiency

An important aspect when considering parallel computa-

Fig. 12 Conventional cylindrical lens (left) and an equivalent Fresnel lens (right).

Fig. 13 Efficiency as a function of the number of processors.

tions is the efficiency $\eta = \frac{T_1}{pT_p}$, i.e. the ratio of the speed-up factor to the number of processors p (T_n is the runtime on n processors). In an ideal case, the speed of a simulation with p processors is p times higher compared to that of a single processor and the efficiency is 100%. However, due to communication overhead and load imbalance, efficiency is usually lower. The exact performance of an MLFMA implementation, as well as its parallel efficiency, depends on a great number of tunable parameters.

Figure 13 shows the parallel efficiency for a matrix-vector multiplication for two cases: a cylindrical lens modelled with 1.7 million of unknowns, and the Fresnel lens from the previous paragraph. The cylindrical lens was the largest example that could fit into the 8 GByte memory of a single machine. Due to memory constraints, this was not possible for the Fresnel lens, so the runtime on less than 12 processors had to be estimated. This was achieved by performing exactly the same work-cycle as for a normal iteration, but with a lot less memory by constantly reusing the same data structures. This produces incorrect matrix-vector products, but it is a good estimation for the iteration times.

In both cases, the parallel efficiency slowly decreases when the number of processors increases, due to the increasing amount of data which needs to be communicated. For 12 processors, the efficiency is in both cases slightly less than

70%. This corresponds with a speed-up factor of 8.

7. Conclusions

We have presented a parallel MLFMA algorithm that offers very high parallelization efficiency on a cheap GRID cluster connected by a fast switch. The algorithm was applied to a two-dimensional TM scattering problem allowing the full-wave simulation of optical structures of 28 500 wavelengths in size and about 3 million unknowns. The heuristic was proven to scale well up to 12 processors. Future work will comprise the extension of the algorithm to three dimensions and further analysis and improvement to the parallel scalability.

Acknowledgments

The work of J. Fostier was supported by a doctoral grant from the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

References

- [1] W.C. Chew, J.M. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*, Artech House, Boston, 2001.
- [2] L. Tarricone and A. Esposito, *Grid Computing for Electromagnetics*, Artech House, Norwood, MA, 2004.
- [3] S. Velamparambil, W.C. Chew, and J. Song, "10 million unknowns: Is it that big?," *IEEE Antennas Propag. Mag.*, vol.45, no.2, pp.43–58, April 2003.
- [4] F. Wu, Y. Zhang, Z.Z. Oo, and E. Li, "Parallel multilevel fast multipole method for solving large-scale problems," *IEEE Antennas Propag. Mag.*, vol.47, no.4, pp.110–118, Aug. 2005.
- [5] 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.
- [6] M.A. Carr and J. Volakis, "Spherical patch discretization of k-space for improvement interpolation and parallelization of the multilevel fast multipole method," *AP-S*, pp.1219–1222, June 2004.
- [7] J. Fostier and F. Olyslager, "A parallel multilevel fast multipole algorithm for grid computing allowing optical full-wave simulations," *Proc. International Symposium on Antennas and Propagation (ISAP 2006)*, Singapore, Nov. 2006.
- [8] F. Olyslager, D.D. Zutter, and K. Blomme, "Rigorous analysis of the propagation characteristics of general lossless and lossy multi-conductor transmission lines in multilayered media," *IEEE Trans. Microw. Theory Tech.*, vol.41, no.1, pp.79–88, Jan. 1993.
- [9] F. Olyslager and J. Fostier, "Fast and accurate evaluation of the shielding effectiveness of complex enclosures," *17th International Zurich Symposium on Electromagnetic Compatibility*, pp.30–33, Singapore, Feb. 2006.
- [10] D.D. Sleator and R.E. Tarjan, "Self-adjusting binary search trees," *J. ACM*, vol.32, pp.652–686, July 1985.
- [11] R.W. Freund, "A transpose-free quasi-minimal residual algorithm for non-hermitian linear systems," *SIAM J. Sci. Comput.*, vol.14, no.2, pp.470–482, 1993.
- [12] W. Gropp, E. Lusk, and A. Skjellum, *Using MPI: Portable Parallel Programming with the Message-Passing Interface*, MIT Press, Cambridge, MA, 1994.
- [13] J. Dongarra, J.D. Croz, S. Hannarling, and R. Hanson, "An extended set of fortran basic linear algebra subprograms," *ACM Trans. Math. Soft.*, vol.14, pp.1–17, 1988.

- [14] I. Ocket, B. Nauwelaers, J. Fostier, L. Meert, F. Olyslager, G. Koers, J. Stiens, R. Vounckx, and I. Jager, "Characterization of speckle/despeckling in active millimeter wave imaging systems using a first order 1.5d model," Proc. SPIE Photonics Europe 2006, vol.6194, pp.09-1-09-11, Strasbourg, France, April 2006.

Jan Fostier was born in 1982. He received the degree in physical engineering in 2005 from Ghent University, Ghent, Belgium. He is currently working toward the Ph.D. degree in physical engineering at the same university. His current research interests are parallel techniques for the Multilevel Fast Multipole Algorithm.

Femke Olyslager was born in 1966. She received the Electrical Engineering degree in 1989 and the Ph.D. degree in 1993, both from Ghent University, Ghent, Belgium. At present she is a Full Professor in electromagnetics at Ghent University. Her research concerns different aspects of theoretical and numerical electromagnetics. She authored or coauthored more than 200 papers in journals and proceedings. She coauthored *Electromagnetic and Circuit Modeling of Multiconductor Transmission Lines* (Ox-

ford, U.K.: Oxford University Press, 1993) and authored *Electromagnetic Waveguides and Transmission Lines* (Oxford, U.K.: Oxford University Press, 1999). Dr. Olyslager is Assistant Secretary General of the International Union of Radio Science (URSI), an Associate Editor of the IEEE Transactions on Antennas and Propagation and was an Associate Editor of Radio Science. In 1994 she became laureate of the Royal Academy of Sciences, Literature and Fine Arts of Belgium. She received the 1995 IEEE Microwave Prize for the best paper published in the 1993 IEEE Transactions on Microwave Theory and Techniques and the 2000 Best Transactions Paper award for the best paper published in the 1999 IEEE Transactions on Electromagnetic Compatibility. In 2002 she received the Issac Koga Gold Medal at the URSI General Assembly and in 2004 she became laureate of the Royal Flemish Academy of Belgium. She is a Fellow of the IEEE.