Accurate numerical modeling of the TARA reflector system

A. Heldring, J.M. Rius, L.P. Ligthart and A. Cardama

Abstract— **The radiation pattern of the large parabolic reflectors of the TARA system (Transportable Atmospheric RAdar), developed at Delft University of Technology (TUD), has been accurately simulated. The Electric Field Integral Equation (EFIE) formulation has been applied to a model of the reflectors including the feed housing and supporting struts, discretised using the Method of Moments. Because the problem is electrically large (the reflector has a diameter of 33λ) and non-symmetrical, this lead to a badly conditioned linear system of approximately half a million unknowns. In order to solve this system, an iterative solver (GMRES) was used, in combination with the Multilevel Fast Multipole Method (MLFMA). Because of the bad conditioning, the system could only be solved by using a huge preconditioner. A new block-ILU preconditioning algorithm has been employed to allow for efficient out-of-computer core memory preconditioning.**

Keywords— **Reflector antennas, Numerical modeling, Integral Equations, Method of Moments, MLFMA, Preconditioning.**

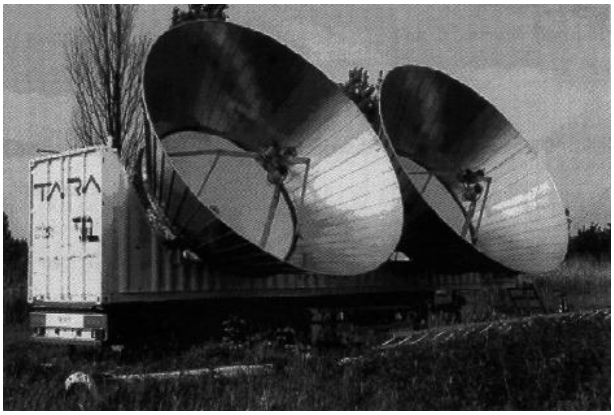

Fig. 1. The TARA measurement facility in operation.

I. Introduction

One of the major projects that has been conducted recently within the International Research Centre for Telecommunications and Radar (IRCTR) of the TUD is TARA (Transportable Atmospheric RAdar system)[1]. The TARA project, financed by the Netherlands Technology Foundation constitutes the design and construction of a mobile measurement facility incorporating the latest developments in antenna and FM-CW technology. The purpose of the measuring facility is the study of atmospheric phenomena such as clouds, precipitation and clear air turbulence.

The TARA project started in 1995 and the measurement facility has become operational in 2000. TARA consists of two large shielded parabolic reflector antennas, one for transmitting, one for receiving, mounted on a truck (see Fig. 1). The operating frequency is 3.3 GHz. The diameter of the reflectors is 3 m, or approximately 33 wavelengths. The reflector antennas are fed by an advanced system of especially designed feed arrays, allowing for multiple beams in different directions and electronic beam-switching [2]. Fig. 2 shows a photograph of one of the two identical reflector antennas (without the shield), mounted in the IRCTR outdoor far field measurement range [3].

Because the system is mobile and meant to function at arbitrary locations, and because of the high sensitivity of the measurements, an essential requirement is a very high suppression of the interference from possible nearby obstacles. For this reason, one of the system specifications is a sidelobe level in the order of -70 to -80 dB at 90^o from the main beam.

The usual appproach to radiation pattern calculations of reflector antennas is to use high frequency techniques. Physical optics (PO) yields accurate results for the maximum gain and for the first few sidelobes. PO can be combined with Physical Theory of Diffraction (PTD) to account for edge diffraction. But this approach fails to accurately incorporate the effect due to feed housings and struts. Alternatively, Geometrical Optics (GO) in combination with Geometrical Theory of Diffraction (GTD) or Uniform Theory of Diffraction (UTD) can be used. However, taking account of all the details of the TARA reflector becomes extremely complicated in practice with these methods.

The TARA problem requires a very high accuracy, in particular for the far sidelobes, which can only be provided by a full wave solution of the entire problem. Such a solution can be obtained with Integral Equation methods (IE) [5], in conjunction with Method of Moments (MoM) [6] discretization. The IE-MoM formulation leads to a full linear system of order N, where N is the number of unknowns of the problem. The operation count for the solution grows with N^3 for direct solution (LU decomposition) or with

A. Heldring, J.M. Rius and A. Cardama are with Department of Signal Theory and Communications (TSC), Universitat Politècnica de Catalunya (UPC), Spain. A. Heldring was former PH.D. candidate at the IRCTR, Delft University of Technology (TUD).

L.P. Ligthart is with the International Research Centre for Telecommunications and Radar (IRCTR), Delft University of Technology (TUD), The Netherlands.

This work was supported by the IRCTR under project "Modeling of Scattering and Radiation Problems", the "Departament d'Universitats Recerca i Societat de la Informació (DURSI)" of the "Generalitat de Catalunya" under "Distinció de la Generalitat de Catalunya per a la Promoció de la Recerca Universitaria", the Spanish "Comisión Interministerial de Ciencia y Tecnología (CICYT)" through grant TIC 2001-2364-C01-01 and the European Commission through FET project IST-2001-33055.

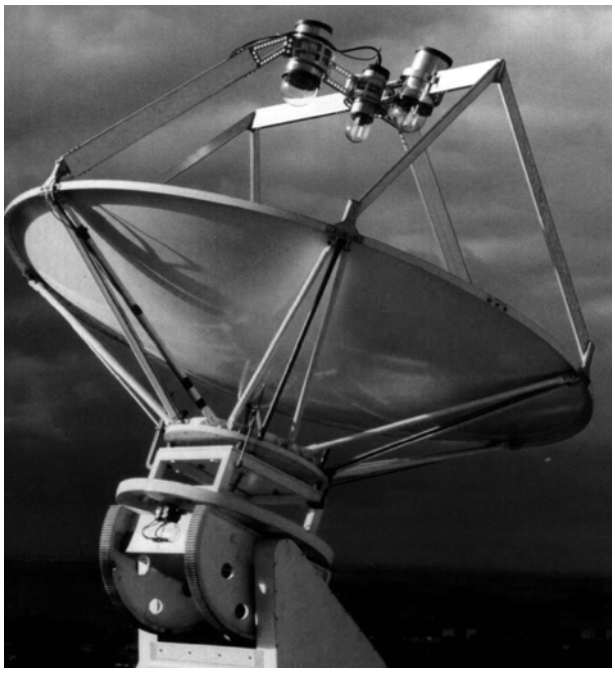

Fig. 2. TARA reflector in measurement setup.

N^2 (a matrix-vector product) per iteration for an iterative method.

The advent of very efficient methods for computing matrix-vector products in the iterative solution of the IE-MoM linear system has brought the solution of problems involving full matrices and a very large number of unknowns (several hundred thousands) within the reach of present computers. Examples of such efficient algorithms are the multilevel fast multipole algorithm (MLFMA) [7][8], or the Multilevel Matrix Decomposition Algorithm (MLMDA) [9][10]. For a problem with a complex three dimensional geometry, embedded in free space, the MLFMA is the most appropriate choice. It has been successfully applied to scattering problems with millions of unknowns [4], using the Combined Field Integral Equation, which leads to a well conditioned linear system of equations and thus allows for fast convergence of iterative solvers.

To our knowledge however, the MLFMA has not yet been applied to the problem of a large reflector antenna radiation pattern. The specific geometry of the TARA problem, a large and electrically thin metal surface (an open problem), calls for the use of the Electric Field Integral Equation (EFIE) instead of the CFIE, which cannot be applied to open problems. However, the EFIE impedance matrix has a much higher condition number that grows with the electrical size of the problem. Very large open problems like the TARA reflectors can only be solved with an iterative method if an efficient preconditioner is applied.

In this article, we present an MLFMA implementation with a new preconditioning scheme that allows to simulate the non-symmetrical TARA reflectors with half a million unknowns on a desktop computer [11]. In section II some aspects of our MLFMA implementation are presented. Concerning the theoretical background of the MLFMA, references to the relevant published literature are provided. In section III we address the preconditioning problem and present our new preconditioning scheme. In section IV, some numerical experiments are presented that illustrate the accuracy of MLMFA and the new preconditioning scheme. In the last section, our simulation results for the TARA radiation pattern are presented and compared to the available measured data.

II. MLFMA

We have implemented the MLFMA on top of a MoM code using a triangular patch model of the problem geometry and Rao, Wilton and Glisson (RWG) basis functions [12]. In order to implement the MLFMA algorithm, the object must be subdivided into a set of non-overlapping boxes at multiple levels in the form of an octal tree: the entire problem geometry is contained within a large cube, that is subdivided into eight cubes, which are subdivided into eight cubes again, and so on recursively, until a chosen minimum rib length is reached. The level of the largest cube is $l_{\max}$ and the smallest boxes are at level 1. The set of basis functions situated inside the same cube at a given level will be referred to as 'a group'. At every level, the interactions between different groups are calculated using the Fast Multipole Method (FMM) if the mutual distance between the cubes is larger than the threshold parameter given below in eq. (2). At level 1 (smallest cubes), interactions between basis functions inside cubes that are too close together are computed in the conventional element by element fashion.

A. Fast Multipole Method

The FMM at the individual levels has been implemented according to the formulation given by Coifman, Rokhlin and Wandzura [7]. It involves a series expansion of the Green's function (the multipole expansion) and a transformation of the fields radiated by the groups to a plane wave expansion and back. The trade off between the accuracy and the efficiency of the FMM is governed by three parameters:

1. The number of terms L in the multipole expansions.
2. The choice of the directions $\widehat{k}_i$ $(i = 1 \cdots K)$ in the plane wave expansion.
3. The minimum distance $R_{\min}$ between the cube centers.

The characteristic parameter for the groups is D_g, the diameter of the sphere enclosing the cubes, equal to $\sqrt{3}$ times the rib length. Coifman, Rokhlin and Wandzura [7] proposed the following empirical formula relating D_g and L:

$$L \approx kD_g + P \log(kD_g + \pi) \tag{1}$$

where k is the wave number and P is a constant, to be chosen as a function of the desired accuracy of the computation. This formula has been adopted and confirmed by various researchers. The minimum distance between the cube centers for the multipole expansion to converge equals

$$R_{\min} = L/k \tag{2}$$

According to (1) and (2), the minimum distance equals the cube diameter plus a term that is important for groups that are small in terms of wavelength, but decreases with growing group size.

For the distribution of the plane wave directions $\widehat{k}_i$ on the unit-sphere, Coifman, Rokhlin and Wandzura proposed to take the zeros of $P_L(\cos\theta)$ (Legendre poynomial of order L) as abscissae in the θ-direction, and $2L$ evenly distributed points in ϕ-direction. This choice guarantees to maintain the chosen accuracy. Gyure and Stalzer [16] show that it is exact for the set of spherical harmonics that form the base of the multipole expansion. The total number of directions then adds up to $K = 2L^2$.

Coifman, Rokhlin and Wandzura state that for $P = 5$, the relative accuracy of the expansion is to single precision (10^{-7}) and for $P = 10$ to double precision (10^{-14}). Song et al. [8] confirm the single precision result for $P = 5$, and claim it is true for the error in the entire MoM computation of a surface current. In their experience, a value of $P = 1$ is sufficient for an error in the surface current of less than 10%. They reason that the computation of the FMM interactions need not be very accurate for an overall good result, because the FMM is only applied to MoM matrix elements that are relatively far from the diagonal, and usually about two degrees of magnitude smaller than the diagonal value. Gyure and Stalzer [16] give the formula $P = \frac{d}{1.6}$ where d is the required number of significant digits in the expansion, which is consistent with Coifman, Rokhlin and Wandzura but more strict than Song et al. for $P = 1$. Darve [17] reports empirical evidence that for $P = 5$, the relative accuracy is 10^{-6} and for $P = 3$, it is 10^{-3}. Furthermore, he finds that there is an optimum in the trade off between accuracy and computational effort for $P = 2.25$. Sertel et al. [18] report that the error in the entire solution of a problem, due to the FMM (the difference between the FMM solution and the direct solution) is not entirely independent of the problem geometry and the excitation. The above estimates should therefore be handled with care. In our implementation we have chosen a value of $P = 2$, based on numerical experiments reported in section II-B.

All the above references are based on the use of the $K = 2L^2$ distribution of the plane wave directions. We have employed an adapted version of this distribution. We found experimentally that without loss of accuracy, one can reduce the number of points in ϕ-direction depending on the latitude (value of θ) and use $K_\phi \approx 2L\sin(\theta)$ points instead of $2L$ points. This reduces the number of directions asymptotically to

$$K \sim 1.27L^2 \tag{3}$$

yielding a non negligible gain in efficiency.

B. Multilevel implementation

The number and distribution of the plane wave directions K_l at level l depends on the size of the cubes at that level. The key to an efficient implementation of the MLFMA is a method to obtain the plane wave representation of a group at level $l+1$ from that of its 'children' (the eight cubes that make it up) at level l. This operation involves an interpolation and a shifting step [8]. The shifting step is straightforward. It shifts the origin of the plane wave representation from the child group centers to the parent group center. Darve [17] gives an overview of the proposed interpolation methods for the common $2L^2$-distribution of multipole directions $\widehat{k}$. The interpolation can be made exact (without loss of information), but at a high price in computational efficiency. For the MLFMA to have a computational complexity of $N \log N$ the interpolation operator must be sparse and therefore approximative.

We have adopted a simple sequence of one dimensional Lagrange interpolation schemes. In both the θ and the ϕ direction, the interpolation is obtained by fitting a polynomial of degree w through the $w+1$ points that are nearest to the evaluation point. For the $1.27K^2$-distribution this is a sequence of three steps. For interpolation between the levels l and $l+1$ a sparse operator $\mathbf{\Phi_l}$ is applied that interpolates from the ϕ_l abscissa to an intermediary set of abscissa ϕ'that are equal for all meridians. Then an operator $\mathbf{\Theta_l^{l+1}}$ interpolates between the old and the new θ-abscissa. Finally, an operator $\mathbf{\Phi^{l+1}}$ interpolates from the ϕ' to the new ϕ_{l+1}. The entire interpolation operator

$$\mathbf{W_l^{l+1}} = \mathbf{\Phi^{l+1}\Theta_l^{l+1}\Phi_l} \tag{4}$$

is a sparse $K_{l+1} \times K_l$ matrix with αK_{l+1} non-zeros, where α only depends on the degree of the interpolation.

Fig. 3 shows the non zero patterns for $\mathbf{W_l^{l+1}}$ for three different values of w. The construction of $\mathbf{W_l^{l+1}}$ from its three factors causes little extra non zero elements (fill-in). For example, for second degree interpolation (three points per row in the factors), the number of non zero elements per row in $\mathbf{W_l^{l+1}}$ converges to 11 for large group diameter.

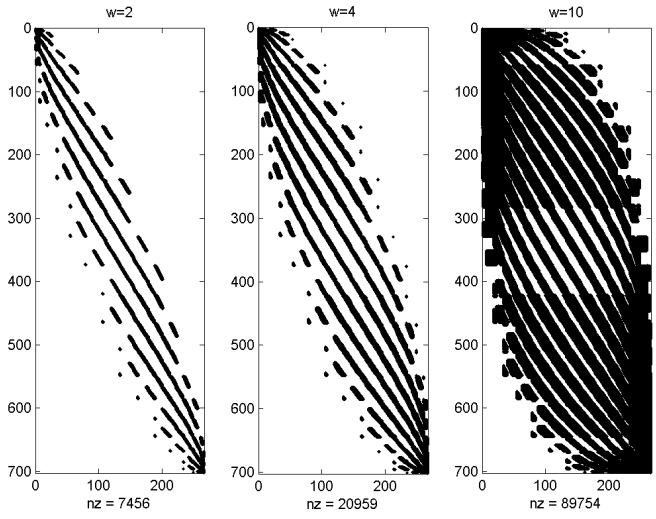

Fig. 3. Non zero patterns of MLFMA interpolation matrices (1260×390 elements) for different degrees of Lagrange schemes (2, 4 and 10). The number of non zeros per row is respectively 5.9, 16.6 and 71.2.

In order to assess the accuracy of of our implementation of the MLFMA, we have conducted a numerical experi-

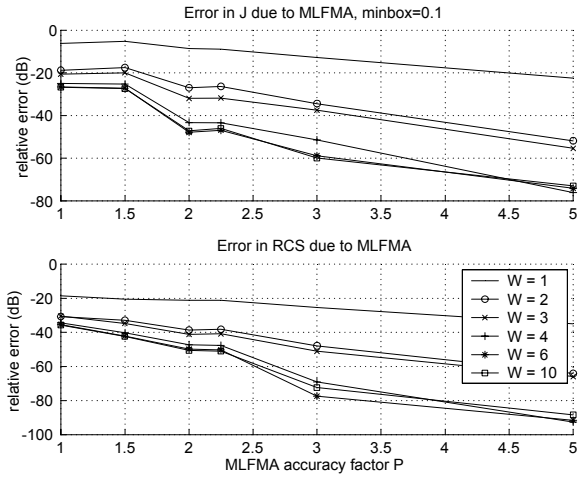

Fig. 4. Relative difference between direct solution and MLFMA solution for a $0.2\lambda \times 6.4\lambda$ perfectly conducting plate, for various values of the MLFMA precicion factor P and the interpolation degree w.

ment comparing the MLFMA to a direct LU decomposition of the full impedance matrix. As a testing problem we chose an infinitely thin rectangular perfectly conducting plate with dimensions $0.2\lambda \times 6.4\lambda$. This geometry allows for a direct solution because it can be modelled with relatively few unknowns ($N = 318$), but at the same time, since it spans several wavelengths, the MLFMA decomposition contains multiple levels ($l_{\max} = 5$). The plate was illuminated by a plane wave normal to the surface. We determined the relative difference

$$\Delta_{\mathrm{j}} = \frac{|\mathbf{j}_{direct} - \mathbf{j}_{MLFMA}|}{|\mathbf{j}_{direct}|} \tag{5}$$

where the $\mathbf{j}$ are vectors of RWG basis function coefficients representing the computed surface current. The relative error for RCS in the specular direction is computed similarly. The experiment was repeated for a sequence of values of the precision parameter P and the interpolation degree w. The results are shown in Fig. 4. In our experience, the RWG model with a discretisation size of around 0.1λ has a relative accuracy for the far field in the range -20 dB to -40 dB, depending on the problem geometry. Clearly, it is useless to choose the MLFMA parameters such that the error due to MLFMA is several orders of magnitude lower than the error that is inherent to the RWG model, at a high computational cost. The results for the rectangular plate, shown in Fig. 4, indicate a choice of $P = 2$, $w = 4$ to keep the additional MLFMA error well below the -40 dB level. However, reasonable results with an additional MLFMA error lower than -30 dB can be obtained with the combination $P = 1.5$, $w = 2$.

III. Preconditioning

In order to solve iteratively the linear system resulting from MoM discretization, we have used the Generalised Minimum Residual method (GMRES, [13]). The GMRES is a robust method for indefinite matrices like the EFIE impedance matrix. In every GMRES iteration step, the bulk of the effort consists of one matrix-vector multiplication of the impedance matrix with a working vector $\mathbf{x}$. The MLFMA accelerates the matrix-vector multiplication with an operation count proportional to $N \log N$, rather than the N^2 factor arising from conventional direct multiplication.

The iteration is stopped when

$$\frac{|\mathrm{Z}\mathbf{x} - \mathbf{e}|}{|\mathbf{e}|} < \epsilon_r, \tag{6}$$

where $\mathbf{e}$ is the excitation vector and ϵ_r is a given threshold, typically 1%. However, the EFIE impedance matrix is generally badly conditioned, and the condition number grows with the electrical size of the problem. For large problems, the convergence can be very slow or even stagnate completely. The remedy is to use a left-preconditioner M^{-1}, to transform the original linear system as follows

$$\mathrm{Z}\mathbf{j} = \mathbf{e} \Rightarrow \mathrm{M}^{-1}\mathrm{Z}\mathbf{j} = \mathrm{M}^{-1}\mathbf{e}, \tag{7}$$

where $\mathbf{j}$ is the unknown surface current coefficient vector.

M^{-1} is chosen such that (a) it improves the total system condition number and (b) it is efficient to compute and to apply to the working vector. Condition (a) implies that M^{-1} should somehow approximate the inverse of Z. Condition (b) implies that both M and M^{-1} should be sparse or may be approximated by sparse matrices. It turns out that, in the case of the EFIE impedance matrix, we can obtain an efficient preconditioner by choosing for M a subset of those elements of Z that represent interactions between elements at a mutual distance smaller than a chosen radius R_{M}. As with the MLFMA, an octal tree subdivision can be used to find these elements in an efficient way.

Subsequently, we build an Incomplete LU preconditioner (ILU) [13], which is an LU factorisation of M with the elements on every row that are much smaller than the diagonal value dropped. The preconditioner is applied to the working vector $\mathbf{x}$ as

$$\mathrm{M}^{-1}\mathbf{x} \approx \mathrm{U}^{-1}\mathrm{L}^{-1}\mathbf{x} \tag{8}$$

by forward and back substitution.

If the condition number were independent of the problem size, as is the case for the CFIE, this preconditioner would scale linearly with N. For the EFIE however, large problems require larger R_{M} and lower threshold values in the ILU. Consequently, the size of M, L and U grows fast. There is no theory available to estimate the relation between threshold values, sizes and convergence rate in a general case. For problems of the scale adressed in this paper ($N > 1 \times 10^5$), the typical size of L and U matrices is several Gigabytes in double precision. This poses a problem for the ILU factorisation, since it needs simultaneous access to all elements in M,L and U. The conventional ILU algorithm is therefore limited to factorisations that fit in core memory.

We have developed a new block ILU algorithm that overcomes this problem, by partitioning the matrix M into sev-

eral blocks, and performing the factorisation on a block-by-block basis. This allows entire blocks to be swapped to and from hard disk storage, both in the factorisation process and in the subsequent application phase.

A. Block ILU algorithm

The proposed block ILU algorithm is based on the partitioned inverse formulas [14]. If a given matrix A is partitioned into four blocks

$$\mathrm{A}=\begin{bmatrix} \mathrm{P} & \mathrm{Q} \\ \mathrm{R} & \mathrm{S} \end{bmatrix} \tag{9}$$

where P and S are square submatrices, then the inverse can be expressed as

$$\mathrm{A}^{-1}=\begin{bmatrix} \widetilde{\mathrm{P}} & \widetilde{\mathrm{Q}} \\ \widetilde{\mathrm{R}} & \widetilde{\mathrm{S}} \end{bmatrix} \tag{10}$$

with

$$\begin{aligned} \widetilde{\mathrm{P}} &= (\mathrm{P}-\mathrm{QS}^{-1}\mathrm{R})^{-1} \\ \widetilde{\mathrm{Q}} &= -\widetilde{\mathrm{P}}(\mathrm{QS}^{-1}) \\ \widetilde{\mathrm{R}} &= -(\mathrm{S}^{-1}\mathrm{R})\widetilde{\mathrm{P}} \\ \widetilde{\mathrm{S}} &= \mathrm{S}^{-1}+(\mathrm{S}^{-1}\mathrm{R})\widetilde{\mathrm{P}}(\mathrm{QS}^{-1}) \end{aligned} \tag{11}$$

as can be verified by substituting (11) in the product AA^{-1}.

In order to use (11) recursively on the preconditioning matrix M, it is subdivided into a chosen number of blocks of approximately equal sizes, based on the problem geometry. As a result M looks like

$$\mathrm{M}=\begin{array}{|c|c|c|c|c|} \hline \mathrm{P}_n & \mathrm{Q}_n^{n-1} & \mathrm{Q}_n^{n-2} & \cdots & \mathrm{Q}_n^0 \\ \hline \mathrm{R}_n^{n-1} & \mathrm{P}_{n-1} & & & \vdots \\ \hline \mathrm{R}_n^{n-2} & & \ddots & & \\ \hline \vdots & & & \mathrm{P}_1 & \mathrm{Q}_1^0 \\ \hline \mathrm{R}_n^0 & \cdots & & \mathrm{R}_1^0 & \mathrm{S}_1 \\ \hline \end{array} \tag{12}$$

where $n+1$ is the total number of blocks. Since M contains the largest elements in each row of the linear system matrix A and zeros elsewhere, with proper column and row ordering the diagonal blocks P_m and S_1 are full or almost full, while many of the off-diagonal submatrices Q_m^k and R_m^k are very sparse. In the following, Q_m denotes the entire right-to-diagonal submatrix $[\mathrm{Q}_m^{m-1}\cdots\mathrm{Q}_m^0]$ and likewise R_m denotes the entire below-diagonal submatrix. Furthermore, $\mathrm{Q}_m^{(p)}$ denotes the block $[\mathrm{Q}_m^p\cdots\mathrm{Q}_m^0]$ and likewise for $\mathrm{R}_m^{(p)}$.

First, the ILU factorization of S_1 is computed with a drop tolerance τ_1. The operator $\left(U^{-1}L^{-1}\right)'$ with LU=S_1 is represented by Σ_1', where the prime means that thresholding has been applied to drop small elements. Σ_1' is applied on the right to Q_1 and on the left to R_1, yielding the factors

$$\Theta_1 = \mathrm{Q}_1\Sigma_1' \tag{13}$$

$$\Gamma_1 = \Sigma_1'\mathrm{R}_1 \tag{14}$$

All elements below a chosen threshold τ_2 with respect to the largest elements in Θ_1 and Γ_1 are dropped (set to zero). The resulting matrices are denoted with Θ_1' and Γ_1'. Subsequently, the factor

$$\Pi_1 = \mathrm{P}_1 - \mathrm{Q}_1\Gamma_1' \approx \mathrm{P}_1 - \mathrm{Q}_1\mathrm{S}_1^{-1}\mathrm{R}_1 \tag{15}$$

is computed, and a dropping with threshold τ_2 is applied to Π_1, yielding Π_1'. Then, Π_1' is ILU factorized. The operator $\mathrm{U}^{-1}\mathrm{L}^{-1}$ with LU= Π_1' is represented by $\widetilde{\mathrm{P}}_1$. The operators

$$\widetilde{\mathrm{R}}_1 = -\Gamma_1'\widetilde{\mathrm{P}}_1 \tag{16}$$

and

$$\widetilde{\mathrm{Q}}_1 = -\widetilde{\mathrm{P}}_1\Theta_1' \tag{17}$$

are not explicitly stored, but whenever they are needed as operators on a matrix or vector, equations (16) and (17) are invoked. This concludes step one.

Now $\widetilde{\mathrm{P}}_1$,$\widetilde{\mathrm{Q}}_1$ and $\widetilde{\mathrm{R}}_1$ are available as operators, and so is $\Sigma_1' \approx \mathrm{S}_1^{-1}$. Subsequently, the operators Γ_m', Θ_m' and $\widetilde{\mathrm{P}}_m$ are computed and stored sequentially for $m = 2\cdots n$. Θ_m is defined as $\mathrm{Q}_m\Sigma_m$ and can be found with the recursive formula

$$\mathrm{Q}_p\Sigma_q' = \begin{bmatrix} \left[\Xi_{q-1}^p\right] & \left[-\Xi_{q-1}^p\Theta_{q-1}' + \mathrm{Q}_p^{(q-2)}\Sigma_{q-1}\right] \end{bmatrix} \tag{18}$$

in which

$$\Xi_{q-1}^p = -(\mathrm{Q}_p^{q-1} - \mathrm{Q}_p^{(q-2)}\Gamma_{q-1}')\widetilde{\mathrm{P}}_{q-1} \tag{19}$$

Likewise, Γ_m is defined as $\Sigma_m\mathrm{R}_m$ and can be found with the recursive formula

$$\Sigma_q'\mathrm{R}_p = \begin{bmatrix} \Lambda_{q-1}^p \\ -\Gamma_{q-1}'\Lambda_{q-1}^p + \Sigma_{q-1}\mathrm{R}_p^{(q-2)} \end{bmatrix} \tag{20}$$

in which

$$\Lambda_{q-1}^p = \widetilde{\mathrm{P}}_{q-1}(\mathrm{R}_p^{q-1} - \Theta_{q-1}'\mathrm{R}_p^{(q-2)}) \tag{21}$$

Every operation in (18) and (20) is followed by a dropping with threshold τ_2, to obtain sparser Γ_m' and Θ_m' from Γ_m and Θ_m. At every level, once Γ_m' is known, $\widetilde{\mathrm{P}}_m$ is computed as $\widetilde{\mathrm{P}}_m = \mathrm{U}^{-1}\mathrm{L}^{-1}$ with LU= Π_m', where Π_m' is obtained from

$$\Pi_m = \mathrm{P}_m - \mathrm{Q}_m\Gamma_m' \tag{22}$$

and a dropping with threshold τ_2.

A close examination of (18) and (20) shows that the operators Σ_m' never have to be computed explicitly: for the computation of Γ_m', Θ_m' and $\widetilde{\mathrm{P}}_m$ only Γ_k', Θ_k' and $\widetilde{\mathrm{P}}_k$ with $k = m-1, m-2\cdots 1$ are needed. At each level m, all the previous factors are loaded and used recursively, down to level one. For symmetrical matrices like the EFIE impedance matrix, the factorisation workload and storage is reduced by a factor of about one half because $\Gamma_m = \Theta_m^{\mathrm{T}}$. The choice of the threshold values τ_1 and τ_2 is a trade-off between the effectiveness and the size of the preconditioner.

Once the preconditioner factors are computed, the preconditioner has to be applied at each iteration step to a working vector $\mathbf{x}$,

$$\mathbf{y} \approx \mathrm{M}^{-\mathbf{1}}\mathbf{x} \tag{23}$$

The vector $\mathbf{x}$ is subdivided into $n+1$ blocks corresponding to the blocks of M. Let $\mathbf{x_i}$ denote the subset of elements corresponding to block i and $\mathbf{x_{(i)}}$ the subset of elements corresponding to blocks 1 to i. The same notation applies to the vector $\mathbf{y}$. For block one,

$$\mathbf{y_1} = \mathbf{\Sigma'_1 x_1} \tag{24}$$

The following blocks are computed recursively for $m = 2 \cdots n+1$ with

$$\mathbf{y_{(m)}} = \begin{bmatrix} \boldsymbol{\nu}_\mathbf{m} \\ -\Gamma'_{m-1}\boldsymbol{\nu}_\mathbf{m} + \mathbf{y_{(m-1)}} \end{bmatrix} \tag{25}$$

where

$$\boldsymbol{\nu}_\mathbf{m} = \widetilde{\mathrm{P}}_\mathbf{m-1}(\mathbf{x_m} - \mathbf{\Theta'_{m-1}x_{(m-1)}}) \tag{26}$$

IV. Numerical experiments

In this section, a number of numerical experiments are presented in order to assess the accuracy of the computation. Firstly, in section IV-A, the RCS of a $10\lambda \times 10\lambda$ infinitely thin perfectly conducting (PEC) square plate is computed, comparing three different methods: MLFMA with a conventional ILU preconditioner, MLFMA with a 16 block ILU preconditioner and MoM with direct inversion.

Secondly, in section IV-B, the dependence of the MLFMA computation time and convergence on the problem size and on the angle of incidence is investigated. This comparison is made both with a conventional ILU preconditioner and with a 16 block ILU preconditioner. For this investigation, a $10\lambda \times 10\lambda$ square plate (the same as in section IV-A), a $20\lambda \times 20\lambda$ square plate and a $40\lambda \times 40\lambda$ square plate are used.

Finally, in the next section, the radiation pattern of the TARA reflector is computed with the method presented in this paper and the result is compared with measurements.

A. $10\lambda \times 10\lambda$ Square plate

A 10 $cm \times$ 10 cm square plate was modelled into a mesh of right angled triangular patches, each triangle being one half of a 0.1 $cm \times$ 0.1 cm patch. This led to a model consisting of $N = 29800$ RWG basis functions. The incident field was a plane wave with a wavelength of 1 cm, with the incidence direction perpendicular to the plate surface and the electric field parallel to two edges of the plate. The MLFMA was applied to this model, with the following parameters: precision parameter $P = 1.5$ and degree of interpolation $w = 2$. The side lengths of the smallest boxes in the MLFMA octal tree were set to 0.1 cm. This resulted in a 7 level algorithm.

A preconditioning matrix M was constructed, as described in section III, with $R_\mathrm{M} = 0.25\lambda$. This lead to aproximately 66 non-zero elements per row in M, occupying about 40 MB of memory storage in double precision complex sparse storage format (approximately 20 bytes per element). From M, two different ILU decompositions were obtained, a conventional 'one block' ILU and a 16 block ILU, using the algorithm of section III-A. All the thresholds in the two ILU decompositions were set to 0.0025.

The system was solved using GMRES both with the one-block ILU preconditioner and with the 16 block ILU preconditioner. The stopping criterium for the GMRES was set to 1% for the relative residual. The matrix sizes are shown in table I (column A) and the computation times for the two cases in table II. These results show that the computation times and the effectivity of the two preconditioning methods are approximately equal.

The results of the two computations agreed well, with a 0.05% difference in the computed back scatter RCS. In order to validate the MLFMA model, the same problem was solved using the Method of Moments (MoM) with direct inversion. To this aim, the fourfold symmetry of the problem was exploited to reduce the number of unknowns to 7400. This was still too large for direct inversion, so the block ILU algorithm was invoked to build and invert the full impedance matrix on a block by block basis (16 blocks), setting all thresholds to zero. Building the matrix took 18 min. and the decomposition took 41 min. Fig. 5 gives the bistatic RCS of the square plate, for perpendicular incidence angle, as computed with MLFMA and MoM with direct block inversion. The relative difference at the maximum is 1.5%.

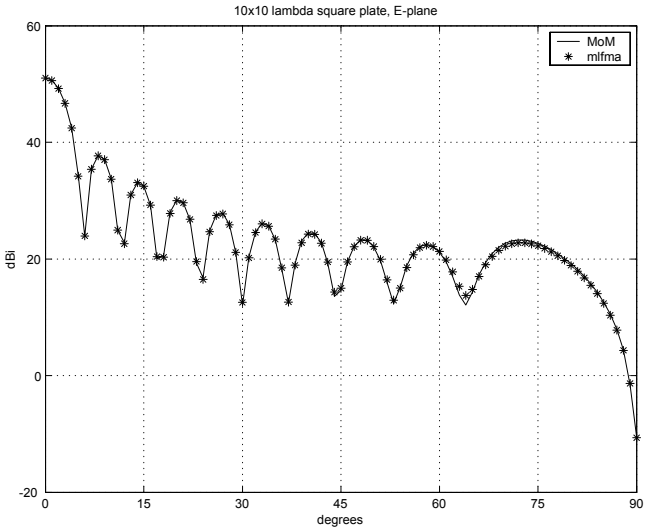

Fig. 5. Bistatic RCS in the E plane of a 10 $\lambda \times 10$ λ square plate with perpendicular incidence direction, computed with MLFMA and with MoM, direct inversion.

B. Sequence of square plates

In order to investigate the growth in computation time and required number of iteration steps to converge with growing problem size, the surface current has been computed for a sequence of square plates. The smallest plate was the same 10 $cm \times 10$ cm square plate from section IV-A, followed by square plates with edge lengths 20 cm and 40 cm. The illumination in all the computations was a $\lambda = 1$ cm plane wave. The computations were repeated for two different angles of incidence, perpendicular to the plate

and parallel to the plate, with the electric field parallel to two edges of the plate in both cases. All computations were done with the GMRES/MLFMA method with the MLFMA parameters set to: precision parameter $P = 1.5$ and degree of interpolation $w = 2$, as in section IV-A. The meshes were all constructed with the same triangle size, leading to a number of unknowns N that grows with a factor of approximately four, as Table I shows. The three square plates were labeled plate A, B and C, in the order of growing size.

TABLE I
SIZE IN WAVELENGTHS, NUMBER OF UNKNOWNS (RWG BASIS FUNCTIONS) AND MLFMA LEVELS FOR THREE DIFFERENT INFINITELY THIN PEC SQUARE PLATES.

plate	A	B	C
size (λ)	10×10	20×20	40×40
unknowns	29800	119600	479200
levels	7	8	9

TABLE II
RWG/MLFMA/GMRES SURFACE CURRENT CALCULATION OF A $10\lambda \times 10\lambda$ SQUARE PLATE WITH PERPENDICULAR ($\perp$) AND PARALLEL ($\parallel$) PLANE WAVE ILLUMINATION AND WITH TWO DIFFERENT PRECONDITIONERS

plate A ($10\lambda \times 10\lambda$)		
ILU	1 block	16 blocks
size of M	33 MB	33 MB
size of ILU	137 MB	132 MB
ILU build time	1 min 38 s	2 min 45 s
GMRES ($\epsilon_r < 1\%$)		
step time	8 s	8 s
$\perp$ incidence		
number of steps	13	13
iteration time	1 min 44 s	1 min 44 s
$\parallel$ incidence		
number of steps	20	20
iteration time	2 min 40 s	2 min 40 s

TABLE III
RWG/MLFMA/GMRES SURFACE CURRENT CALCULATION OF A $20\lambda \times 20\lambda$ SQUARE PLATE WITH PERPENDICULAR ($\perp$) AND PARALLEL ($\parallel$) PLANE WAVE ILLUMINATION AND WITH TWO DIFFERENT PRECONDITIONERS

plate B ($20\lambda \times 20\lambda$)		
ILU	1 block	16 blocks
size of M	134 MB	134 MB
size of ILU	564 MB	548 MB
ILU build time	11 min 27 s	16 min 40 s
GMRES ($\epsilon_r < 1\%$)		
step time	38 s	39 s
$\perp$ incidence		
number of steps	16	17
iteration time	10 min 8 s	11 min 3 s
$\parallel$ incidence		
number of steps	41	41
iteration time	25 min 58 s	26 min 39 s

TABLE IV
RWG/MLFMA/GMRES SURFACE CURRENT CALCULATION OF A $40\lambda \times 40\lambda$ SQUARE PLATE WITH PERPENDICULAR ($\perp$) AND PARALLEL ($\parallel$) PLANE WAVE ILLUMINATION USING A 16 BLOCK ILU PRECONDITIONER.

plate C ($40\lambda \times 40\lambda$)	
ILU	16 blocks
size of M	541 MB
size of ILU (MB)	2.4 GB
ILU build time	4 h 1 min
GMRES ($\epsilon_r < 1\%$)	
step time	250 s
$\perp$ incidence	
number of steps	21
iteration time	1 h 28 min
$\parallel$ incidence	
number of steps	59
iteration time	4 h 6 min

In all three cases, the preconditioning matrix M was constructed as described in section III with $R_{\mathrm{M}} = 0.25\lambda$. The computer requirements are summarised in Tables II, III and IV. The tables show that the size of M grows approximately linearly with N, as expected. All the preconditioners were built with threshold $\tau_1 = 0.0025$ and for all the 16 block preconditioners, a second threshold $\tau_2 = 0.0025$ was used. Only for the two smaller plates, a conventional (single block) ILU preconditioner was built. For the largest plate, this could not be done due to core memory constraints.

The total size of the ILU preconditioner (sum of all the blocks) grows linearly with N. However, the ILU decomposition time grows faster. There are two reasons for this: Firstly, many elements are computed and then dropped, if their value is below the threshold level. This is a process which is impossible to predict because it depends on the actual composition of M and therefore on every individual problem. Secondly, in small problems a significant number of blocks are stored in memory, while in large problems almost all the blocks are stored in hard disk, thus saving a considerable amount of time if the entire preconditioner fits in core memory.

The time per GMRES step (iteration) is also given in the tables. The time per step includes the MLFMA matrix-vector product, the back and forward substitution of the preconditioner and a (small) overhead for the GMRES search-directions orthogonalisation. Theoreticaly, this time grows with $N \log N$ (due to the MLFMA). Again, the times given in the tables are influenced by the number of blocks that are stored in memory or in hard disk.

The computations were done with two different incident field directions, perpendicular and parallel to the plate surface. The convergence of the GMRES iteration turns out to depend highly on the incident field. In the case of perpendicular incidence, the number of iterations increases very slowly with the size of the plate. In contrast, with parallel incidence the convergence is highly dependent on the size of the plate.

V. TARA Reflector radiation pattern

We have modeled the TARA reflector system from Fig. 2 in detail using the meshing package GiD [19] developed at the International Center for Numerical Methods in Engineering (CIMNE), in Barcelona, Spain. The model is shown in Fig. 6. The parabolic dish has a diameter of 3 m and a focal distance of 1.535 m, the wavelength being 9.1 cm. The struts are metal beams with cross-sections of 12 $cm \times 2.5$ cm and 6 $cm \times 2.5$ cm. The feed housings have been modeled as closed PEC cylinders. The central feed housing has a height of 24 cm and a diameter of 11.3 cm. The off-axis feed housings have a height of 20 cm and a diameter of 26 cm. They are tilted 6^o with respect to the focal axis. The feed housings, struts and the reflector surface at a distance smaller than λ from the reflector edge were meshed with an average edgelength of 0.6 cm. The rest of the reflector surface was meshed with an average edgelength of 1.2 cm. This lead to $N = 479200$ RWG basis functions (38% of which represent the reflector surface, the rest is used for the feeds and supporting struts). The supporting structure underneath the reflector has been left out of consideration. The incident field used in the calculations was the measured radiation pattern of the central feed.

The induced surface current on the entire structure was computed using the MLFMA/GMRES method exposed in this paper, with the MLFMA parameters chosen to keep the error due to the MLFMA approximations in the far field below 1%. The parameters, matrix sizes and computation times for this computation are summarised in table V. The 40 block ILU preconditioner was built using thresholds $\tau_1 = \tau_2 = 0.0025$. The largest block had a size of 150 MB. The GMRES iteration was stopped at a relative residual below 1% after 126 steps and the resulting surface current was used to calculate the radiation pattern. To make sure that the stopping criterium was sufficient, the iteration was interrupted at $\epsilon_r = 2.01\%$ after 106 steps, and the radiation pattern was also computed with the surface current solution at that stage. The two patterns were undistinguishable.

Fig. 7 shows the computed and measured radiation patterns for the E-plane, with the feed polarisation perpendicular to the long horizontal strut visible in Fig. 2. The measurements have been made at the IRCTR outdoor measurement facility. The results agree well qualitatively. The measured maximum gain was 38 ± 0.5 dBi, the calculated maximum gain was 38.4 dBi. The sidelobe at approximately -10^o is considerably higher in the MLFMA simulation. This sidelobe is caused by a secondary reflection from the off-axis feed in the E-plane. A possible explanation for the divergence between the computed and the measured result, is that the physical feed structure, consisting of an array of two dielectric rods, disperses the field whereas the flat metal surface assumed in the computational model causes a specular reflection in the observed direction. The convergence of the solution is slow, as table V shows. This is caused by the multiple reflections between the reflector and the struts and feed housings.

TABLE V
RWG/MLFMA/GMRES surface current solution of the TARA reflector. The computation times correspond to a personal computer with Athlon XP1900+ CPU.

RWG	
number of unknowns	479200
MLFMA	
P	2
interpolation degree	4
levels	9
40 block ILU	
R_{M}	0.35λ
size of M	8.4 GB
size of ILU	3.42 GB
ILU build time	4h 45 min
GMRES ($\epsilon_r < 1\%$)	
step time	8min 47 sec
number of steps	126
iteration time	18 h 26 min
total computation time	25 h 15 min

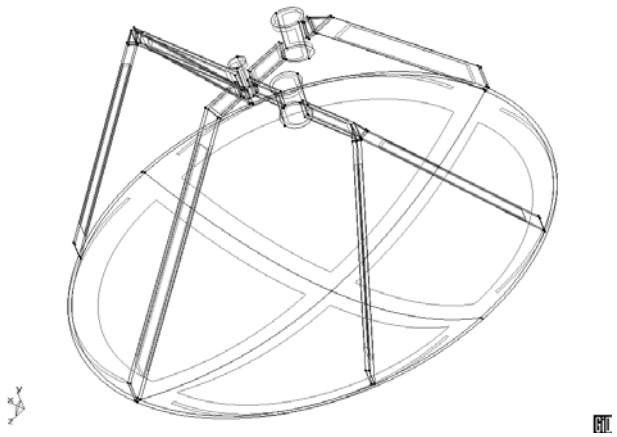

Fig. 6. Model of the TARA reflector with feeds and struts.

VI. Conclusion

We have presented a new block Incomplete LU preconditioner scheme that is as efficient as the well known conventional Incomplete LU preconditioner but, due to its block structure, allows the use of a preconditioner much larger than the available core memory. Indeed, the only limita-

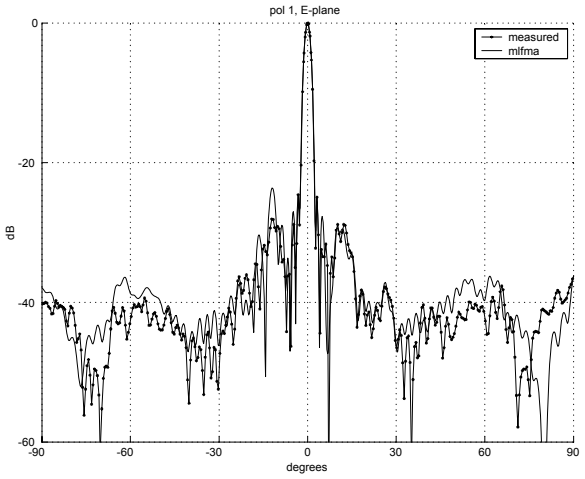

Fig. 7. Measured and computed radiation pattern (E-plane) of the TARA reflector.

tion is hard-disk space. Using this new technique in combination with the MLFMA, we have been able to perform a full wave analysis of an electrically large reflector antenna, including all the relevant details. The results are presented and compared to measurements of the reflector antenna.

References

[1] S. Heijnen and L.P. Ligthart,"TARA: Transportable Atmospheric Radar", Proc. 28th European Microwave Conf, p 61, 1998

[2] A. Moumen, "Analysis and synthesis of compact feeds for large multiple-beam reflector antennas", PhD Thesis, Delft University of Technology, 2001

[3] P. Aubry, P. Hakkaart, J.H. Zijderveld and L.P. Ligthart,"Implementation of an outdoor Far-Field Measurement system at IRCTR", Proc. 28th European Microwave Conf., pp. 549-552, 1998

[4] S. Velamparambil and W.C. Chew, "Parallelization of MLFMA on distributed memory computers", *International Conference on Electromagnetics in Advanced Applications* (ICEAA-01), Torino 10-14 September 2001, pp. 141-144.

[5] N. Morita, N. Kumagai and J.R. Mautz, *Integral Equation Methods for Electromagnetics*, Artech House, 1990.

[6] R. F., Harrington, *Field Computation by Moment Methods*, MacMillan, New York, 1968.

[7] R. Coifman, V. Rohklin and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian description", *IEEE Antennas Propag. Mag.*, Vol. 35, pp. 7-12, 1993.

[8] 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, October 1997.

[9] J. M. Rius, J. Parrón, E. Úbeda and J. R. Mosig, "Multilevel matrix decomposition algorithm for analysis of electrically large electromagnetic problems in 3-D", *Microwave and Optical Technology Letters*, Vol. 22, No. 3, pp. 177-182, August 1999.

[10] J. Parrón, J.M. Rius, and J.R. Mosig, "Application of the Multilevel Decomposition Algorithm to the Frequency Analysis of Large Microstrip Antenna Arrays", *IEEE Trans. on Magnetics*, Vol.38, No.2, pp. 721-724, March 2002.

[11] A. Heldring, "*Full Wave Analysis of Electrically Large Reflecto Antenna*s", Ph. D. Thesis, Delft University Press, April 2002.

[12] R. M. Rao, D. R. Wilton and A. W. Glisson, "Electromagnetic Scattering by Surfaces of Arbitrary Shapes ", *IEEE Transactions on Antennas and Propagation*, Vol 30, No 3, pp. 409-418, May 1982.

[13] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.

[14] W.H. Press e.a, *Numerical Recipes in C*, Cambridge University Press, Cambridge, 2nd Ed. 1994.

[15] S.H. Heijnen and L.P. Ligthart, "TARA: Transportable Atmospheric RAdar ", *Proc. 28th European Microwave Conference*, Vol. 1, pp.61-66, 1998.

[16] M.F. Gyure and M.A. Stalzer,"A Prescription for the Multilevel Helmholtz FMM", *IEEE Computational Science and Engineering*, pp 39-47, July 1998.

[17] E. Darve,"The Fast Multipole Method: Numerical Implementation", *Journal of Computational Physics*, Vol. 160, pp. 195-240, 2000.

[18] K. Sertel, M. Carr and J.L. Volakis, "Error Evaluation of Fast Integral Methods for RCS Computations", *Millenium Conference on Antennas and Propagation (AP2000)*, Davos, Switzerland, 2000.

[19] R. Ribó et al. "*GiD reference manual*". CIMNE , 1998.