A Fast MoM Solution for Large Arrays: Green's Function Interpolation With FFT

B. J. Fasenfest, *Student Member, IEEE*, F. Capolino, *Senior Member, IEEE*, D. R. Wilton, *Fellow, IEEE*, D. R. Jackson, *Fellow, IEEE*, and N. J. Champagne, *Senior Member, IEEE*

***Abstract*—A new type of fast method of moments (MoM) solution scheme for large arrays is developed using standard basis functions. Both fill and solve times are improved with respect to standard MoM solvers. The efficiency of the method relies on approximating the Green's function as a sum of separable interpolation functions defined on a relatively sparse uniform grid, along with use of the fast Fourier transform. The method permits the analysis of arrays with arbitrary contours and/or missing elements. Preliminary results show the effectiveness of the method for planar array elements in free space.**

***Index Terms*—Adaptive integral method, antenna arrays, fast solvers.**

I. Introduction

THE numerical modeling of large arrays remains a challenging problem, although much progress has been made [1]–[5]. Large array problems solved by standard method of moments (MoM) procedures require huge fill and solve times, and make large storage demands for the associated matrices.

Here, we present a new algorithm that combines the use of Green's function interpolation and fast Fourier transforms (GIFFT). The method is similar to the adaptive integral method (AIM) [3] in the sense that it projects the solution domain onto a regular grid to enable use of the fast Fourier transform (FFT) algorithm. The key difference is how the projections are done. In AIM, radiating basis and testing functions are replaced by a neighboring grid of approximately equivalent monopole sources, and the usual Green's function is used to compute the interactions between these groups of equivalent sources. In GIFFT, the Green's function is approximated on an interpolation grid, and the basis and testing function integrations are done in the usual manner using the interpolated Green's function. The method is able to efficiently handle large arrays while still maintaining the generality of standard MoM solvers with respect to excitations, array contour geometries, missing or defective array elements, etc.

Manuscript received December 19, 2003; revised May 7, 2004. This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory, the University of California, under Contract W-7405-Eng-48.

B. J. Fasenfest, D. R. Wilton, and D. R. Jackson are with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204-4005 USA (e-mail: bfasenfe@mail.uh.edu; wilton@uh.edu; djackson@uh.edu).

F. Capolino is with the Department of Information Engineering, University of Siena, 53100 Siena, Italy (e-mail: capolino@dii.unisi.it).

N. J. Champagne was with Lawrence Livermore National Laboratory, Livermore, CA 94550-9234 USA. He is now with Louisiana Tech University, Ruston, LA 71270 USA (e-mail: champagn@latech.edu).

Digital Object Identifier 10.1109/LAWP.2004.833713

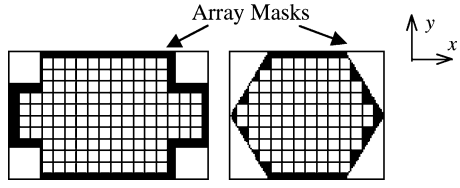

Fig. 1. Array mask describes the location of array elements within the array boundary outline, which is in turn contained within a bounding box.

The efficiency of the GIFFT method is based on two key steps. First, interactions between sufficiently separated array elements may be performed via a relatively coarse interpolation of the Green's function on a uniform grid commensurate with the array's periodicity. The Green's function is approximated over the interpolation grid as a sum of separable functions. This allows the system matrix to be filled efficiently. Second, since the interpolatory coefficients in the Green's function interpolation are functions only of the separation between source and observation grid points they are of convolutional form and, hence, an FFT may be used to efficiently calculate the matrix-vector product in an iterative solver. The solution time for the GIFFT method can be further improved by combining the method with suitable preconditioners such as that introduced in [6], although further details are omitted here.

II. Green's Function Interpolation With FFT

The GIFFT method shown in the following is applicable to arrays with identical elements and arbitrary boundaries. The approach can also be used on large arbitrary scatterers (nonarray problems), but without the advantage of array-cell reusability.

The *array boundary* is defined by specifying the vertices of a closed, piecewise linear curve. An *array mask* is defined as a matrix whose row and column dimensions equal, respectively, the number of rows and columns of array elements that would fit into its bounding box. Its entries indicate by the presence of a 1 or 0 whether an array element is present or not at that index location. Typical array boundaries are represented in Fig. 1. The displacement between the $\mathbf{p}'$th and $\mathbf{p}$th array cells is $(p_1 - p_1')\boldsymbol{s}_1 + (p_2 - p_2')\boldsymbol{s}_2$, where $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are two arbitrary lattice vectors lying in the xy plane. The vector $\mathbf{r} = \boldsymbol{\rho} + z\hat{\mathbf{z}}$, with $\boldsymbol{\rho} = x\hat{\mathbf{x}} + y\hat{\mathbf{y}}$, is in cell $\mathbf{p} = (p_1, p_2)$ and $\mathbf{r}' = \boldsymbol{\rho}' + z'\hat{\mathbf{z}}$ is in cell $\mathbf{p}' = (p_1', p_2')$. The surface current on array elements within cells $\mathbf{p}$ and $\mathbf{p}'$ are discretized, for instance, using the basis functions of [7], with

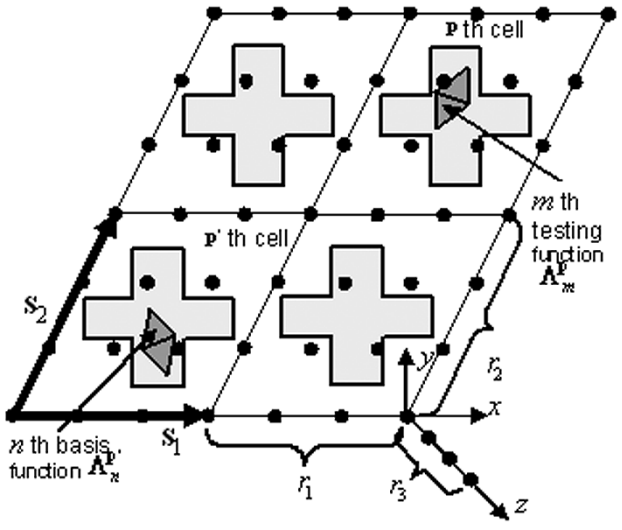

Fig. 2. Array cell index definitions and arbitrary skew lattice vectors. The periodic grid on which the Green's function is sampled is shown superimposed on the array cells. Within an array cell, the Green's function is evaluated on an array of $r_1 \times r_2 \times r_3$ points.

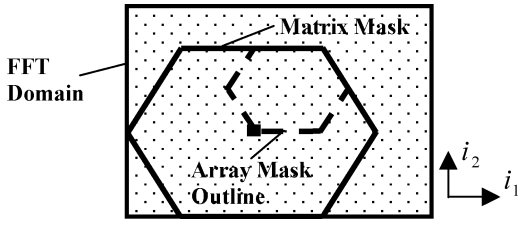

Fig. 3. Matrix mask for a hexagonal array is obtained by translating the array mask about one array element of the array. The FFT domain is obtained by padding to the next power of 2 the number of array cells (blocks) of the matrix mask in both array directions.

primed and unprimed indexes denoting source and testing functions, respectively (Fig. 2).

A. Array and Matrix Masks, and the FFT Domain

From the array mask, a *matrix mask* (Fig. 3) is synthesized that indicates permissible cell index separations $\mathbf{p}-\mathbf{p}'$ between pairs of interacting elements in the array; it is the locus of elements covered by all possible translations of the array mask about a fixed element. Thus, the matrix mask tabulates all possible interactions between elements in the array mask in terms of their allowable separation indexes; that is, it determines what sample values of the Green's function are needed for interpolation. A nonzero matrix entry represents coupling between a pair of array elements with the corresponding element index separation. Also shown in Fig. 3 is the *FFT domain*, which is the bounding box of the matrix mask padded to the next power of two in each lattice dimension.

B. Electric Field Integration Equation (EFIE) Formulation

Since we deal here with conducting array elements in free space, the MoM is constructed around the EFIE, written symbolically in compact form as

$$\int_S \boldsymbol{\mathcal{G}}^{\mathrm{E}}(\boldsymbol{\rho}-\boldsymbol{\rho}',z,z')\cdot\mathbf{J}(\mathbf{r}')dS' = -\mathbf{E}^{\mathrm{inc}}_{\tan} \tag{1}$$

where the integration domain S represents all conducting elements over the entire finite array. This integral equation has the discretized form

$$\sum_{\mathbf{p}'}\sum_{n=1}^{N}\left\langle \boldsymbol{\Lambda}^{\mathbf{p}}_m(\mathbf{r});\boldsymbol{\mathcal{G}}^{\mathbf{E}}(\boldsymbol{\rho}-\boldsymbol{\rho}',z,z');\boldsymbol{\Lambda}^{\mathbf{p}'}_n(\mathbf{r}')\right\rangle I^{\mathbf{p}'}_n = V^{\mathbf{p}}_m \ \forall m,\mathbf{p}$$

where $\boldsymbol{\Lambda}^{\mathbf{p}'}_n$ and $\boldsymbol{\Lambda}^{\mathbf{p}}_m$ are, respectively, the basis and testing functions of [7], and $V^{\mathbf{p}}_m = \langle \boldsymbol{\Lambda}^{\mathbf{p}}_m;\mathbf{E}^{\mathrm{inc}}\rangle$. The inner product symbolizes integration over both array source and observation domains. In supermatrix form, the discretized EFIE may be written symbolically as

$$\left[Z^{\mathbf{pp}'}_{mn}\right]\left[I^{\mathbf{p}'}_n\right] = \left[V^{\mathbf{p}}_m\right] \tag{2}$$

with

$$Z^{\mathbf{pp}'}_{mn} = -\left\langle \boldsymbol{\Lambda}^{\mathbf{p}}_m(\mathbf{r});\boldsymbol{\mathcal{G}}^{\mathrm{E}}(\boldsymbol{\rho}-\boldsymbol{\rho}',z,z');\boldsymbol{\Lambda}^{\mathbf{p}'}_n(\mathbf{r}')\right\rangle. \tag{3}$$

C. Lagrange–Green's Function Interpolation (GIFFT Filling Acceleration)

For elements separated by at least one array cell, i.e., with different $\mathbf{p}$ and $\mathbf{p}'$, we approximate the Green's function $\boldsymbol{\mathcal{G}}^{\mathrm{E}}(\boldsymbol{\rho}-\boldsymbol{\rho}',z,z')$ via Lagrange polynomial interpolation as

$$\boldsymbol{\mathcal{G}}^{\mathrm{E}}(\boldsymbol{\rho}-\boldsymbol{\rho}',z,z') \cong \sum_{\mathbf{i},\mathbf{i}',j,j'} L_{\mathbf{i}}(\boldsymbol{\rho})L_j(z)\boldsymbol{\mathcal{G}}^{\mathrm{E}}_{\mathbf{i}-\mathbf{i}',j,j'}L_{\mathbf{i}'}(\boldsymbol{\rho}')L_{j'}(z') \tag{4}$$

where the indexes on

$$\boldsymbol{\mathcal{G}}^{\mathrm{E}}_{\mathbf{i}-\mathbf{i}',j,j'} \equiv \boldsymbol{\mathcal{G}}^{\mathrm{E}}(\boldsymbol{\rho}^{(\mathbf{i})}-\boldsymbol{\rho}^{(\mathbf{i}')},z^{(j)},z^{(j')})$$

denote sampled values of the coordinates and $L_{\mathbf{i}}(\boldsymbol{\rho})L_j(z)$ are Lagrange interpolation polynomials defined on the $r_1r_2r_3$ interpolation points within a cell (c.f. Fig. 2). It is significant that in the evaluation of the interaction $Z^{\mathbf{pp}'}_{mn}$ between two array cells $\mathbf{p}$ and $\mathbf{p}'$, the interpolation scheme generally requires many fewer Green's function evaluations than in the usual case where subdomain interactions are evaluated directly, or even, apparently, than AIM usually requires. This becomes especially true as the complexity of an array element increases. Furthermore, the separable nature of the Green's function representation in (4) allows for the matrix to be filled with many fewer inner product calculations than with standard MoM. This may be seen by writing the mn element of the matrix block $Z^{\mathbf{pp}'}_{mn}$ of (3), corresponding to array cell $\mathbf{p}$ and source cell $\mathbf{p}'$, as

$$\begin{aligned} Z^{\mathbf{pp}'}_{mn} &\approx \tilde{Z}^{\mathbf{pp}'}_{mn} \\ &\equiv -\sum_{\mathbf{i},\mathbf{i}',j,j'} \langle \boldsymbol{\Lambda}^{\mathbf{p}}_m(\mathbf{r}),L_{\mathbf{i}}(\boldsymbol{\rho})L_j(z)\rangle\cdot\boldsymbol{\mathcal{G}}^{\mathrm{E}}_{\mathbf{i}-\mathbf{i}',j,j'} \\ &\quad\cdot\left\langle L_{\mathbf{i}'}(\boldsymbol{\rho}')L_{j'}(z'),\boldsymbol{\Lambda}^{\mathbf{p}'}_n(\mathbf{r}')\right\rangle \end{aligned} \tag{5}$$

where the tilde $(\sim)$ denotes the approximated block. The following three important properties are then used to significantly speed the computation.

1) Unless the grid point $\mathbf{i}' = (i'_1, i'_2)$ lies within the cell $\mathbf{p}'$, $\langle L_{\mathbf{i}'},L_{j'},\boldsymbol{\Lambda}^{\mathbf{p}'}_n\rangle_{r_1\times r_2\times r_3\times N}$ vanishes since the Lagrange polynomial associated with an interpolation point is

nonzero only over the cell containing the interpolating point. The same holds for $\langle L_{\mathbf{i}} L_j, \mathbf{\Lambda}_m^{\mathrm{P}} \rangle_{r_1 \times r_2 \times r_3 \times N}$.

2) Since $\langle L_{\mathbf{i}'} L_{j'}, \mathbf{\Lambda}_n^{\mathrm{P}'} \rangle_{r_1 \times r_2 \times r_3 \times N}$ and $\langle L_{\mathbf{i}} L_j, \mathbf{\Lambda}_m^{\mathrm{P}} \rangle_{r_1 \times r_2 \times r_3 \times N}$ are identical for each array element, they need only be computed once. Furthermore, for Galerkin's method (i.e., when the basis and testing functions are identical) the quantities are equal.

3) One need only compute $\boldsymbol{\mathcal{G}}^{\mathrm{E}}_{\mathbf{i},j,j'}$ for index pairs $\mathbf{i} = (i_1, i_2)$ appearing within the matrix mask of Fig. 3.

If there are N basis functions per array element, then from properties 1) and 2), only $r_1 \times r_2 \times r_3 \times N$ distinct inner products are required. In addition, sampled values of the Green's function need only be calculated and stored on the interpolation grid, which is relatively sparse.

D. GIFFT Solution Acceleration: Fast Computation of Matrix-Vector Products

The approximated mutual coupling matrix (5) is inaccurate if the cell (index) separation is not sufficiently large since low order interpolation of the Green's function is inaccurate near the source point. To avoid this inaccuracy, the array element self-block coupling and that between neighboring element blocks is found by standard MoM, i.e., using standard integration rules and Green's function evaluation for the interaction between basis and test functions. With good accuracy, the original discretized EFIE in (2) is thus rewritten as

$$\left[\Delta Z_{mn}^{\mathrm{PP}'}\right]\left[\mathrm{I}_n^{\mathrm{P}'}\right] + \left[\tilde{Z}_{mn}^{\mathrm{PP}'}\right]\left[\mathrm{I}_n^{\mathrm{P}'}\right] = [V_m^{\mathrm{P}}] \tag{6}$$

where the block Toeplitz *difference matrix* $\Delta Z_{mn}^{\mathrm{PP}'} = Z_{mn}^{\mathrm{PP}'} - \tilde{Z}_{mn}^{\mathrm{PP}'}$ is taken as zero for elements satisfying $|\mathbf{p} - \mathbf{p}'| > c$ and is, hence, sparse. We also note that generally $\boldsymbol{\mathcal{G}}^{\mathrm{E}}_{\mathbf{i}-\mathbf{i}',j,j'} = \infty$ when $\mathbf{i} = \mathbf{i}'$, $j = j'$ in $[\Delta Z_{mn}^{\mathrm{PP}'}]$ and $[\tilde{Z}_{mn}^{\mathrm{PP}'}]$, but this infinite value can be replaced by any finite value with (6) remaining valid. To evaluate the matrix/vector product, we note that the product $[\Delta Z_{mn}^{\mathrm{PP}'}][\mathrm{I}_n^{\mathrm{P}'}]$ can be performed quickly since $\Delta Z_{mn}^{\mathrm{PP}'}$ is sparse, whereas $[\tilde{Z}_{mn}^{\mathrm{PP}'}][\mathrm{I}_n^{\mathrm{P}'}]$ is of convolutional form and can be evaluated quickly using a two-dimensional FFT as follows:

$$\left[\tilde{Z}_{mn}^{\mathrm{PP}'}\right]\left[\mathrm{I}_n^{\mathrm{P}'}\right] = \sum_{\mathbf{i},j,j'} \langle \mathbf{\Lambda}_m^{\mathrm{P}}, L_{\mathbf{i}} L_j \rangle \cdot \mathsf{MASK}_{\mathbf{i}} \mathsf{FFT}_{\mathbf{i}}^{-1} \cdot \left[\left(\mathsf{FFT}_{\mathbf{i}} \overline{\overline{\boldsymbol{\mathcal{G}}^{\mathrm{E}}_{\mathbf{i},j,j'}}}\right) \bullet \mathsf{FFT}_{\mathbf{i}} \left(\overline{\overline{\sum_{\mathbf{p}'} \sum_{n=1}^{N} \left\langle L_{\mathbf{i}} L_{j'}, \mathbf{\Lambda}_n^{\mathrm{P}'} \right\rangle \mathrm{I}_n^{\mathrm{P}'}}}\right)\right] \tag{7}$$

where the double bars over a quantity indicate that its length is extended so as to obtain a circular convolutional form and then zero-padded to obtain vectors of length 2^k for applying the FFT (FFT); FFT^{-1} denotes the inverse fast Fourier transform, and $\mathsf{MASK}_{\mathbf{i}}$ is the array mask restricting the result to array elements within the array boundary. The degree to which (6) approximates (2) depends on how many elements in $[\Delta Z_{mn}^{\mathrm{PP}'}]$ (called the *strong interaction matrix*) are set to zero, and on how many interpolation points are used for $[\tilde{Z}_{mn}^{\mathrm{PP}'}]$. Numerical experiments verify that setting $\Delta Z_{mn}^{\mathrm{PP}'}$ to zero for elements more distant than a wavelength provides good accuracy.

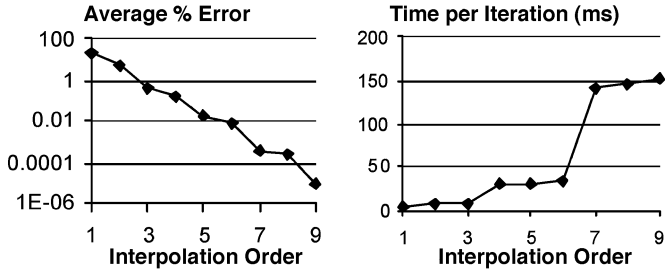

Fig. 4. Average error in dipole currents and time per iteration versus interpolation order.

III. Numerical Results

To assess the interpolation accuracy, it is sufficient to test a small array of 5×5 strip dipoles in free space, each discretized using 24 triangles and the basis functions of [7]. The dipoles are illuminated by a plane wave with the electric field polarized along the dipole axis at a frequency $f = 380\,\mathrm{MHz}$. The dipoles have length $l = 0.494\lambda$ and width $w = 0.025\lambda$. The interactions $\Delta Z_{mn}^{\mathrm{PP}'}$ have been set to zero for array cell pairs separated by more than one cell, i.e., for $|p_1 - p_1'| > 1$ or $|p_2 - p_2'| > 1$. The average percent error in the current at the center of each dipole of the array is plotted versus the Lagrange polynomial interpolation order in Fig. 4. The error at each element is calculated relative to a reference solution using an element-by-element MoM scheme, and then averaged over all the elements. The computation time per iteration taken by the BiCGstab iterative method increases with interpolation order since more terms are involved in each FFT matrix-vector multiplication.

The GIFFT method is next applied to an array of 25×25 square conducting patches in free space, illuminated by a plane wave at 6 GHz from a direction perpendicular to the array plane. The patches are 11.4 mm on a side with a spacing of 3.2 mm between the edges of adjacent patches. A triangular meshing of each patch creates $\mathrm{N} = 65$ unknowns per patch. The GIFFT method uses fifth-order interpolating polynomials in both planar directions and the block Toeplitz property to aid in filling the $\Delta Z_{mn}^{\mathrm{PP}'}$ matrix. The method's performance is compared with a standard MoM that also exploits the block Toeplitz properties. GIFFT required the storage of only 25 Green's function evaluations per array element, whereas the MoM required $\mathrm{N}^2 = 4225$ matrix entries per element. A dramatically reduced fill and solution time is also observed using GIFFT: The fill time for GIFFT was 158 times faster than for the standard MoM, while the solution time was 49 times faster. The GIFFT solution also agrees well with the standard MoM solution: the average error in the far field was 0.1% and average error of the currents on the plates was 0.9%. The far field patterns for both methods are shown in Fig. 5.

IV. Conclusion

The GIFFT method has been developed for arrays with arbitrary geometries. The method may be summarized as follows. An array mask function is used to identify array boundaries and to specify the domain over which the Green's function is interpolated using a separable representation involving Lagrange polynomials; an FFT is then used to accelerate the matrix-vector

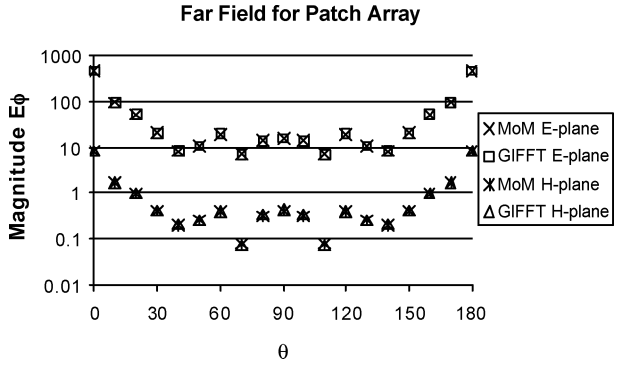

Fig. 5. Far field pattern for scattered E-field for a 25 × 25 array of conducting patches in free space. The angle θ indicates displacement from broadside ($\theta = 0$).

products in an iterative solver. Preliminary results show the effectiveness of the method for large array problems. The method is similar to the AIM method, the main difference being that the Green's function—rather than the basis functions—is projected onto a regular interpolation grid. The interpolation requires relatively few interpolation points per cell, making the method attractive for multi-layered media problems, for example, where the cost for Green's function evaluations is relatively high.

The initial investigations reported here are deliberately limited to cell sizes on the order of a half wavelength. Because the accuracy of the Green's function interpolation depends on the number of interpolation points per wavelength, it is expected that larger array cells would require higher-order interpolation. However, electrically large array cells could also be split into subcells over which low-order interpolation is used, possibly even reducing the cost of computing near interactions.

REFERENCES

[1] A. Neto, S. Maci, G. Vecchi, and M. Sabbadini, "Truncated floquet wave diffraction method for the full wave analysis of large phased arrays: Parts I and II," *IEEE Trans. Antennas Propagat.*, vol. 48, pp. 594–600, April 2000.

[2] O. A. Civi, P. H. Pathak, H.-T. Chou, and P. Nepa, "A hybrid UTD-MoM for efficient analysis for radiation/scattering from large finite planar arrays," *Radio Sci.*, vol. 32, pp. 607–620, Mar.–Apr. 2000.

[3] E. Bleszynski, M. Bleszynski, and T. Jaroszewicz, "AIM: adaptive integral method for solving large scale electromagnetic scattering and radiation problems," *Radio Sci.*, vol. 31, pp. 1225–1251, 1996.

[4] P. De Vita, A. Mori, A. Freni, and M. Calamia, "A BMIA/nCAG formulation for the analysis of large stacked patch antennas," presented at the IEEE AP-S Symp., San Antonio, TX, June, 16–21 2003.

[5] R. W. Kindt and J. L. Volakis, "The array decompositon-fast multipole method," presented at the IEEE AP-S Symp., Columbus, OH, June 2003.

[6] F. Capolino, D. R. Wilton, and D. R. Jackson, "Physical preconditioning for numerical modeling of large periodic arrays," presented at the XXVIIth URSI General Assembly, Maastricht, The Netherlands, Aug. 2002.

[7] S. S. M. Rao, D. R. Wilton, and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Trans. Antennas Propagat.*, vol. AP-30, pp. 409–418, May 1982.