Radio Science, Volume 33, Number 1, Pages 7–16, January–February 1998

Scattering from relatively flat surfaces using the adaptive integral method

Hristos T. Anastassiu, Mikhail Smelyanskiy, Sunil Bindiganavale, and John L. Volakis

Radiation Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan Ann Arbor

Abstract. The adaptive integral method is applied to electromagnetic scattering from relatively flat, perfectly conducting surfaces. It is demonstrated that for nearly flat scatterers, as is the case with corrugated geometries and rough surfaces, the memory requirements and complexity of the technique are extremely low. Furthermore, for such geometries the algorithm can be effectively parallelized without excessive effort. Several numerical results are included, proving the accuracy and computational efficiency of the method.

1. Introduction

Integral equation methods, such as the method of moments (MOM) [*Harrington*, 1961], have been extensively applied to electromagnetic simulations. Although very accurate, MOM is very difficult to use for the analysis of electrically large problems, due to excessive memory requirements and high computational complexity. When iterative solvers are used, the fast Fourier transform (FFT) has been successfully utilized in the solution of integral equations [*Sarkar et al.*, 1986; *Peters and Volakis*, 1988; *Volakis and Barkeshli*, 1991], but its three-dimensional applications are restricted to structures with special geometric properties. More recently, several compression algorithms have been developed to reduce the memory and complexity yet retain the accuracy and generality of MOM. Among these, the adaptive integral method (AIM) [*Bleszynski et al.*, 1994, 1996] utilizes the Toeplitz property of the Green's function kernel to reduce the storage requirements of MOM and also accelerate calculations by use of the FFT. The method is very general and can handle arbitrary geometries in two or three dimensions. Memory requirements and complexity are always asymptotically lower than traditional MOM (for very large unknown counts); however, exact storage and operation estimates strongly depend on the shape of the scatterer. An interesting property of AIM is that memory and CPU time requirements are dramatically reduced for relatively flat (but not necessarily planar) surfaces. Examples of such scatterers include corrugated geometries, rough surfaces, and antenna arrays. For such scatterers, significant speedup can be attained by simply parallelizing the FFT algorithm used in the computation of matrix-vector products. This paper focuses on the merits of AIM as applied to to electromagnetic scattering from nearly flat surfaces.

Paper number 97RS02104.
0048-6604/98/97RS-02104$11.00

2. General Description of AIM

To apply AIM for scattering by perfectly conducting, relatively flat surfaces, we follow an approach similar to that of *Bleszynski et al.* [1996] yet much simplified and more rigorous. We consider an arbitrary, perfectly conducting (PEC) surface Σ illuminated by some incident field $\mathbf{E}^i$. The scattered field $\mathbf{E}^s$ is given by

$$\mathbf{E}^s = -j\omega\mathbf{A} - \nabla\phi \tag{1}$$

where the magnetic potential $\mathbf{A}$ and electric potential ϕ are convolution integrals of the surface current and charge densities $\mathbf{J}_s$ and ρ_s, respectively [*Harrington*, 1961].

For numerical solution of the pertinent integral equation the surface is discretized into small triangular patches, and the unknown current $\mathbf{J}_s$ is expanded using a suitable set of basis functions $\mathbf{f}_n(\mathbf{r})$. That is,

$$\mathbf{J}_s(\mathbf{r}) \simeq \sum_{n=1}^{N} I_n \mathbf{f}_n(\mathbf{r}) \tag{2}$$

where I_n are unknown coefficients (elementary currents). A popular choice for $\mathbf{f}_n(\mathbf{r})$ are the Rao-

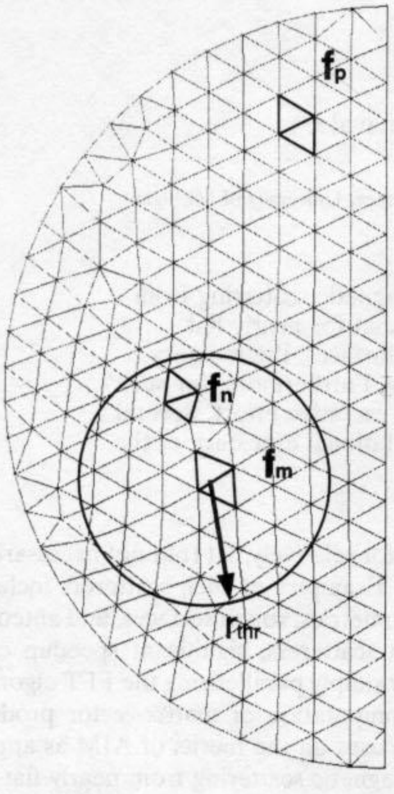

Figure 1. Definition of the far-field region. Here $\mathbf{f}_n$ lies in the near zone, whereas $\mathbf{f}_p$ is in the far zone of $\mathbf{f}_m$.

Wilton-Glisson (RWG) basis functions defined by *Rao et al.* [1982]. It must be emphasized that $\mathbf{f}_n(\mathbf{r})$ is tangential to the nth patch and vanishes outside it.

To develop the integral equation for the solution of $\mathbf{J}_s$, we enforce the boundary condition

$$(\mathbf{E}^i + \mathbf{E}^s) \cdot \mathbf{t} = 0 \qquad \text{on } \Sigma \tag{3}$$

where $\mathbf{t}$ is any tangential vector on the surface or approximately tangential to the triangular patches. Subsequently, Galerkin's technique is employed to cast (3) into a linear system of equations. Specifically, (3) is tested with the mth basis function to get

$$\int_\Sigma \mathbf{E}^i \cdot \mathbf{f}_m \, d^2S = \int_\Sigma j\omega\mathbf{A} \cdot \mathbf{f}_m \, d^2S + \int_\Sigma \nabla\phi \cdot \mathbf{f}_m \, d^2S \tag{4}$$

$$m = 1, \cdots, N$$

and upon introducing the expansion (2), we obtain the linear system

$$[\mathbf{Z}]\{\mathbf{I}\} = \{\mathbf{V}\} \tag{5}$$

The entries of the impedance matrix $[\mathbf{Z}]$ are double surface integrals over Σ corresponding to the interactions among the elementary currents, and

$$V_m = \int_\Sigma \mathbf{E}^i \cdot \mathbf{f}_m \, d^2S \tag{6}$$

Since the impedance matrix in (5) is fully populated, serious memory limitations arise when the geometry is electrically large. Also, the required numerical integration to evaluate the $[\mathbf{Z}]$ matrix entries and the subsequent inversion of $[\mathbf{Z}]$ are both extremely time consuming tasks. These difficulties can be overcome by utilizing the AIM technique.

A basic AIM concept is to partition all pairs of matrix entries into "far-field" and "near-field" ones. A far-field threshold r_{thr} is set according to the desired accuracy. If the centers of two interacting current elements mth and nth lie at a distance larger than r_{thr}, they are considered to be in the far-field region of each other (see Figure 1), and their interaction can be modeled in an alternative way.

To describe the far-field interactions, we envision the entire scatterer being submerged into a uniform rectangular grid (Figure 2). We introduce an auxiliary set of basis functions, namely, $\boldsymbol{\psi}_m$, represented by clusters of delta current sources located at the nodes of the grid, namely,

$$\boldsymbol{\psi}_m(\mathbf{r}) = \sum_{q=1}^{M^3} \delta(x - x_{mq})\delta(y - y_{mq})\delta(z - z_{mq}) \cdot [\Lambda_{mq}^x \hat{\mathbf{x}} + \Lambda_{mq}^y \hat{\mathbf{y}} + \Lambda_{mq}^z \hat{\mathbf{z}}] \tag{7}$$

where M is equal to the expansion order and $\mathbf{r}_{mq}$ are points on the grid surrounding the mth edge. In Figures 2 and 3 we depict the relative position of the nth RWG patch submerged in a grid of order $M = 2$. We note, however, that (7) is only one of type of expansion which can be used for transferring the fields to a uniform rectangular grid.

The Λ coefficients are determined to ensure that the two sets of basis functions $\boldsymbol{\psi}_m$ and $\mathbf{f}_m$ are equivalent. However, since the $\boldsymbol{\psi}_m$ are defined in the whole volume, whereas each $\mathbf{f}_m$ is nonzero only on the mth patch, formal equivalence can be achieved only after a slight modification of the RWG basis function definition. Specifically, let

$$\tilde{\mathbf{f}}_m(\mathbf{r}) \equiv \mathbf{f}_m(\mathbf{r})\delta(z_m^{\pm}) \tag{8}$$

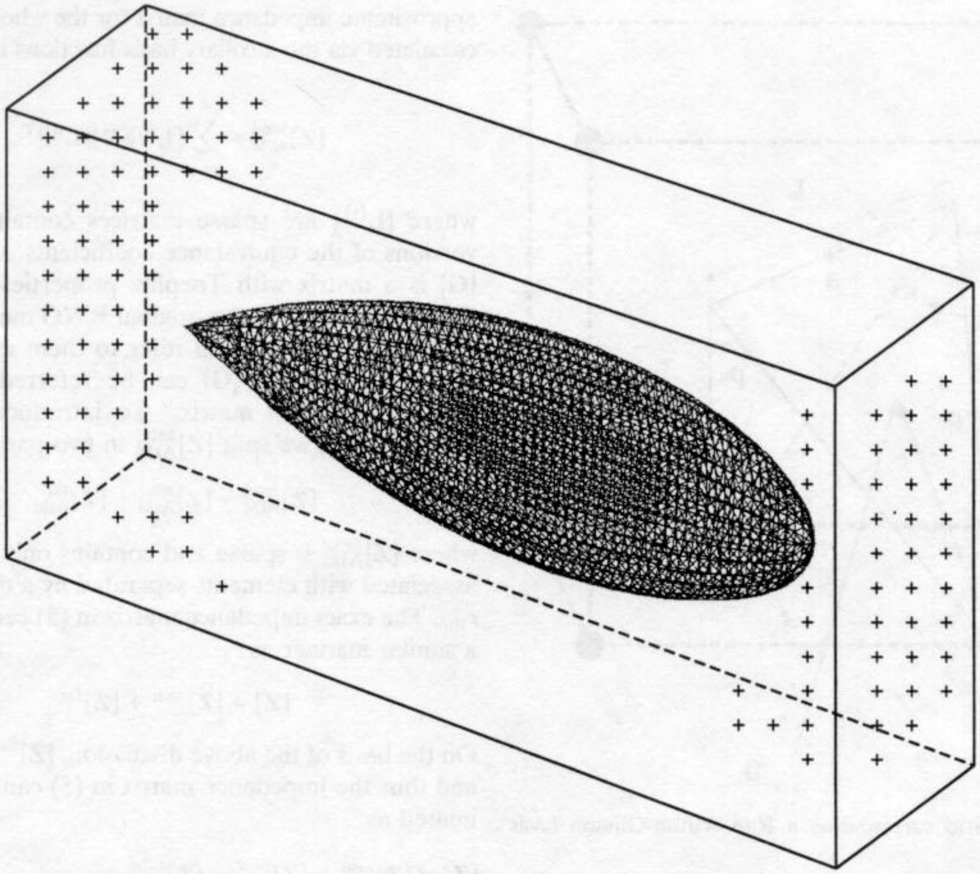

Figure 2. Uniform adaptive integral method (AIM) grid surrounding the scatterer.

where $\delta(z_m^{\pm})$ is a one-dimensional (1-D) delta function and $z_m^{\pm}$ is the local coordinate along the normal to the positive or negative triangles which comprise the RWG diehedral mth patch (see Figure 4). This is an important step to ensure the mathematical rigor of AIM and was omitted from the original presentation given by *Bleszynski et al.* [1994, 1996]. The two sets $\boldsymbol{\psi}_m$ and $\tilde{\mathbf{f}}_m$ can become equivalent by imposing equality of their moments up to order M with respect to the midpoint $\mathbf{r}_a$ of the mth edge. These moments are defined by

$$\mathbf{M}^m_{q_1q_2q_3} \equiv \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \boldsymbol{\psi}_m(\mathbf{r})(x-x_a)^{q_1}(y-y_a)^{q_2} \cdot (z-z_a)^{q_3}\, dx\, dy\, dz$$

$$= \sum_{q=1}^{M^3} (x_{mq}-x_a)^{q_1}(y_{mq}-y_a)^{q_2}(z_{mq}-z_a)^{q_3} \cdot [\Lambda^x_{mq}\hat{\mathbf{x}} + \Lambda^y_{mq}\hat{\mathbf{y}} + \Lambda^z_{mq}\hat{\mathbf{z}}] \qquad (9)$$

$$\tilde{\mathbf{M}}^m_{q_1q_2q_3} \equiv \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \tilde{\mathbf{f}}_m(\mathbf{r})(x-x_a)^{q_1} \cdot (y-y_a)^{q_2}(z-z_a)^{q_3}\, dx\, dy\, dz \qquad (10)$$

where $\mathbf{M}^m_{q_1q_2q_3}$ refers to the moments of the expansion basis on the rectangular grid and $\tilde{\mathbf{M}}^m_{q_1q_2q_3}$ refers to those associated with the original basis. Apart from $\tilde{\mathbf{f}}_m$ itself, the surface divergence of $\tilde{\mathbf{f}}_m$ must be similarly approximated and independently expanded. We will define the moments associated with $\nabla_s \cdot \tilde{\mathbf{f}}_m$ as $D^m_{q_1q_2q_3}$ and $\tilde{D}^m_{q_1q_2q_3}$.

On equating $\mathbf{M}^m_{q_1q_2q_3}$ to $\tilde{\mathbf{M}}^m_{q_1q_2q_3}$ and $D^m_{q_1q_2q_3}$ to $\tilde{D}^m_{q_1q_2q_3}$, we obtain four $M^3 \times M^3$ systems that yield the Λ coefficients as a solution. A considerable portion of our effort was focused on the efficient calculation of the moments in (10). Specifically, a fast,

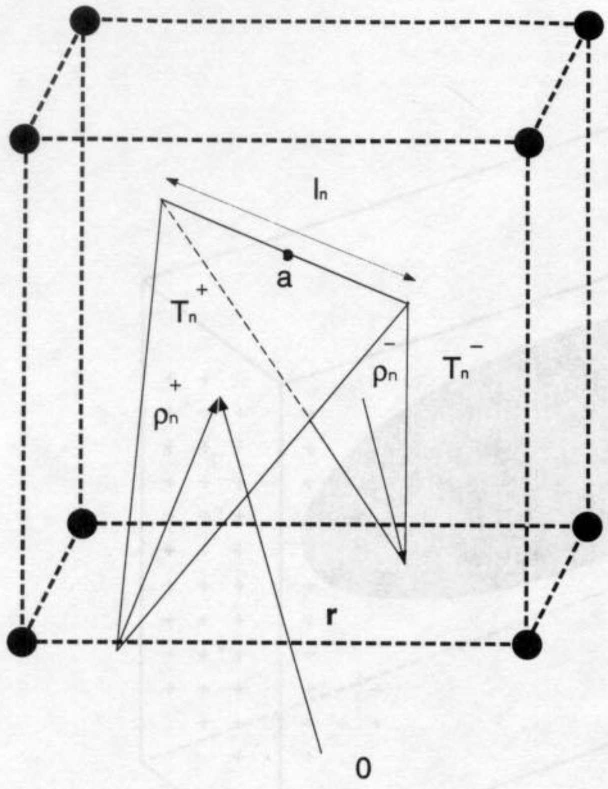

Figure 3. Grid surrounding a Rao-Wilton-Glisson basis function.

closed-form recursive scheme was developed to calculate these moments, and the mathematical details are described in the appendix.

With the introduction of the volumetric basis above, the integral equation (4) can be rewritten as

$$\int_V \mathbf{E}^i \cdot \tilde{\mathbf{f}}_m d^3v = \int_V j\omega\mathbf{A}\cdot\tilde{\mathbf{f}}_m \, d^3v + \int_V \nabla\phi\cdot\tilde{\mathbf{f}}_m \, d^3v \qquad (11)$$

where $\tilde{\mathbf{f}}_m$ represents the volumetric testing function in (8). For $M \to \infty$, the sets $\boldsymbol{\psi}_m$ and $\tilde{\mathbf{f}}_m$ are equivalent, and hence in (11), $\boldsymbol{\psi}_m$ may be used instead of $\tilde{\mathbf{f}}_m$. For practical implementations, though, M must be finite, and therefore only interactions among elements that lie in the far zone of each other can be modeled through the set $\boldsymbol{\psi}_m$. However, since the whole geometry is submerged in the grid, it is inefficient to separate near-field and far-field interactions a priori. It is preferable to calculate all possible interactions using the auxiliary basis functions and afterward replace the near-field interactions by their exact values. It can be shown [*Bleszynski et al.*, 1996] that the approximate impedance matrix for the whole geometry calculated via the auxiliary basis functions is given by

$$[\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{total}} = \sum_{i=1}^{4} [\mathbf{L}^{(i)}][\mathbf{G}][\mathbf{L}^{(i)}]^T \qquad (12)$$

where $[\mathbf{L}^{(i)}]$ are sparse matrices containing scaled versions of the equivalence coefficients Λ in (7) and $[\mathbf{G}]$ is a matrix with Toeplitz properties. Since the $[\mathbf{L}^{(i)}]$ matrices map the original RWG mesh onto the rectangular grid, we will refer to them as "mapping matrices," whereas $[\mathbf{G}]$ can be referred to as the "Green's function matrix." To introduce the exact matrix values, we split $[\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{total}}$ in two parts, as

$$[\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{total}} = [\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{near}} + [\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{far}} \qquad (13)$$

where $[\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{near}}$ is sparse and contains only the entries associated with elements separated by a distance $r < r_{\mathrm{thr}}$. The exact impedance matrix in (5) can be split in a similar manner as

$$[\mathbf{Z}] = [\mathbf{Z}]^{\mathrm{near}} + [\mathbf{Z}]^{\mathrm{far}} \qquad (14)$$

On the basis of the above discussion, $[\mathbf{Z}]^{\mathrm{far}} \simeq [\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{far}}$, and thus the impedance matrix in (5) can be approximated as

$$\begin{aligned}[\mathbf{Z}] &\simeq [\mathbf{Z}]^{\mathrm{near}} - [\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{near}} + [\mathbf{Z}]_{\mathrm{AIM}}^{\mathrm{total}} \\ &= [\mathbf{S}] + \sum_{i=1}^{4} [\mathbf{L}^{(i)}][\mathbf{G}][\mathbf{L}^{(i)}]^T \qquad (15)\end{aligned}$$

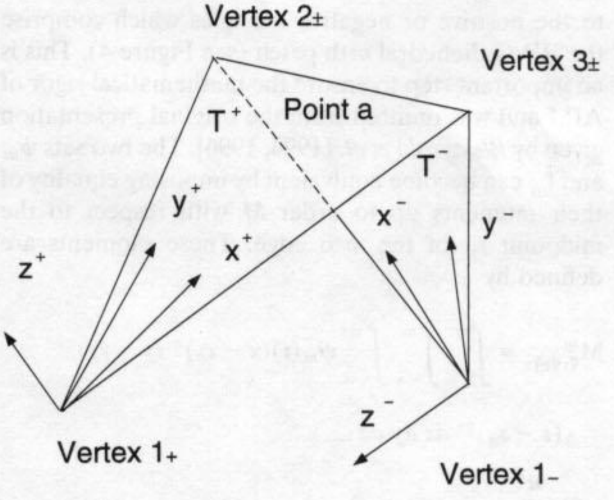

Figure 4. Local coordinate systems for the triangles $T^\pm$ on the scatterer surface.

where [**S**] is a sparse matrix corresponding to the difference between the exact and the AIM-modeled near-field interactions. Of most importance in the formulation of the impedance matrix is that [**G**] is Toeplitz and thus the FFT algorithm can be used to significantly accelerate the calculation of matrix-vector products required for iterative solution of the linear system.

3. Application to Relatively Flat Surfaces

Computational cost estimates show that the memory requirements and complexity of the algorithm depend both on the original number of unknowns N and the total number of grid points N_g. For large N our double precision implementation has a memory requirement of

$$\text{memory} \simeq (368 + 32M^3)N + 314N_g + 16N_{\text{near}} \text{ bytes} \tag{16}$$

and assuming a symmetric biconjugate gradient (BiCG) algorithm employing a radix-2 FFT, the number of complex multiplications per iteration is

$$N_{\text{mult}} \simeq 8M^3N + 540N_g + 120N_g \log_2 N_g + N_{\text{near}} \tag{17}$$

In this, N_{near} is the number of nonzero entries in [**S**], and its value depends on the geometry of the scatterer. For rectangular surfaces it has been shown [*Bleszynski et al.*, 1996] that N_g is asymptotically proportional to $N^{3/2}$. However, N_g is, in general, highly dependent on the geometry. To observe this, let us consider a scatterer of rectangular form whose sides are denoted as a, b, c. If the grid step is chosen as h, then

$$N_g \simeq \frac{abc}{h^3} \tag{18}$$

Assume that the surface of the scatterer is discretized by equilateral triangles of edge length l. The area of each triangle is therefore $S_T = \sqrt{3}l^2/4$, and the number of triangles over the whole surface is

$$N_T \simeq \frac{8\sqrt{3}}{3l^2}(ab + bc + ca) \tag{19}$$

Since the number N of RWG elements is approximately $N \simeq 3N_T/2$, it follows that

$$N_g \simeq \frac{abc}{[4\sqrt{3}(ab + bc + ca)]^{3/2}}\left(\frac{l}{h}\right)^3 N^{3/2} \tag{20}$$

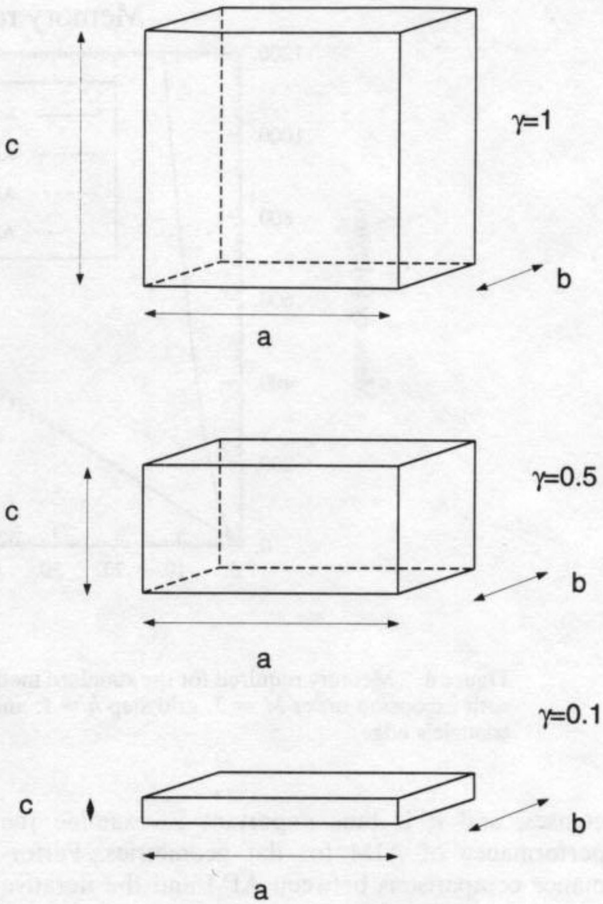

Figure 5. Scatterer shape for various values of the flatness parameter γ.

A similar estimate for N_{near} is

$$N_{\text{near}} \simeq \sqrt{3}\pi\left(\frac{r_{\text{thr}}}{l}\right)^2 N \tag{21}$$

The above analysis, though approximate, shows that when AIM is applied to nearly rectangular surface problems, it has $O(N^{3/2})$ memory requirements and $O(N^{3/2} \log_2 N)$ complexity. Therefore, for a large number of unknowns N, it always performs more favorably than MOM (with an iterative solver), which has $O(N^2)$ complexity and memory requirements.

To demonstrate the dependence of N_g on the relative dimensions of the rectangular scatterer, let $c = \gamma a = \gamma b$, where γ is defined as the "flatness" parameter of the rectangle (see Figure 5). As shown in Figure 5 for small γ, the rectangle is relatively flat and becomes more like a plate as $\gamma \rightarrow 0$. The constant in front of $N^{3/2}$ in (20) becomes smaller as γ de-

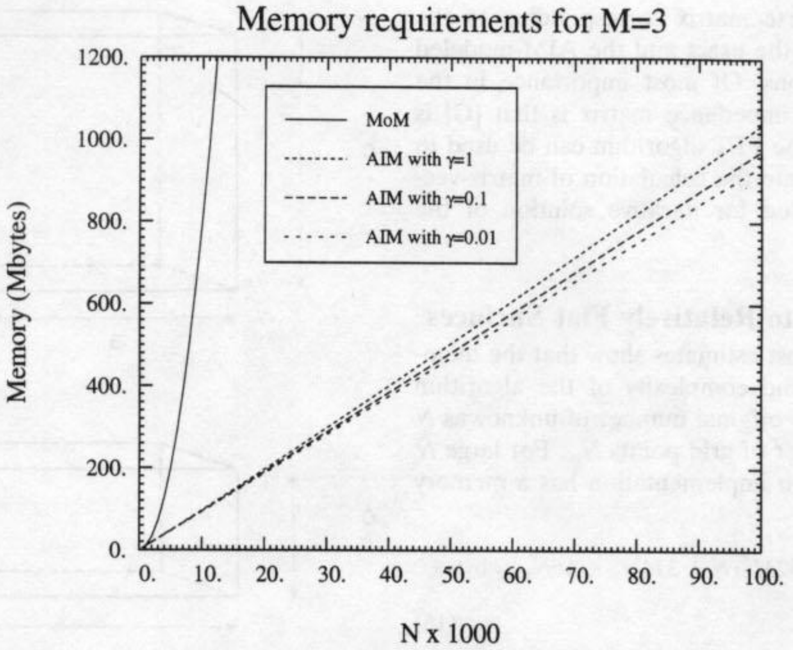

Figure 6. Memory required for the standard method of moments (MOM) and AIM (surface problems with expansion order $M = 3$, grid step $h = l$, and $r_{\text{thr}} = 10l$, where l is the length of the elementary triangle's edge).

creases, and it is thus important to examine the performance of AIM for flat geometries. Performance comparisons between AIM and the iterative MOM for various flatness parameters $h = l$ and $r_{\text{thr}} = 10l$ are given in Figures 6 and 7. In both implementations the BiCG solver was used, and it is clear that AIM is much more efficient than MOM for large N. As can be concluded from (20) the efficiency of AIM increases with the flatness of the scatterer, and for the chosen parameters, AIM is preferrable to MOM in terms of memory and complexity for about $N > 1500$. For larger N the improvement of AIM over MOM cannot be overemphasized. Moreover, for relatively flat surfaces, the AIM grid is also flat, permitting the use of much faster, nearly two dimensional FFTs, which results in a dramatic reduction of the computational cost.

4. Parallelization of the FFT and Numerical Results

Profiling the AIM code showed that a very large portion (up to 75%) of the total CPU time was consumed by the FFT. Although the AIM system solution is faster than MOM, the major bottleneck of the AIM algorithm is filling [S] in (15), a process which may also utilize the FFT. These observations were not unexpected and point to a need for parallelization. For general cases the algorithm has been parallelized via domain decomposition [*Bleszynski et al.*, 1996], an approach that is better suited for geometries that contain large sections of free space between the scatterer and the outer bound of the AIM grid. As can be realized, domain decomposition is fairly complicated and difficult to implement. Alternatively, for nearly flat scatterers the FFT is almost two dimensional, and thus there is little waste in unknowns since the free-space region is minimal. Consequently, speedup via parallelization can be readily achieved by simply parallelizing the FFT subroutines employed at each iteration step of the solver.

Our parallelization steps for the FFT were the following:

1. The array with dimensions n_x, n_y, n_z is distributed into planes, equally divided among all available processors of total number p.

2. One-dimensional FFTs are performed along each of the two dimensions x and y, and this is done n_z/p times for each node concurrently since each node contains n_z/p planes.

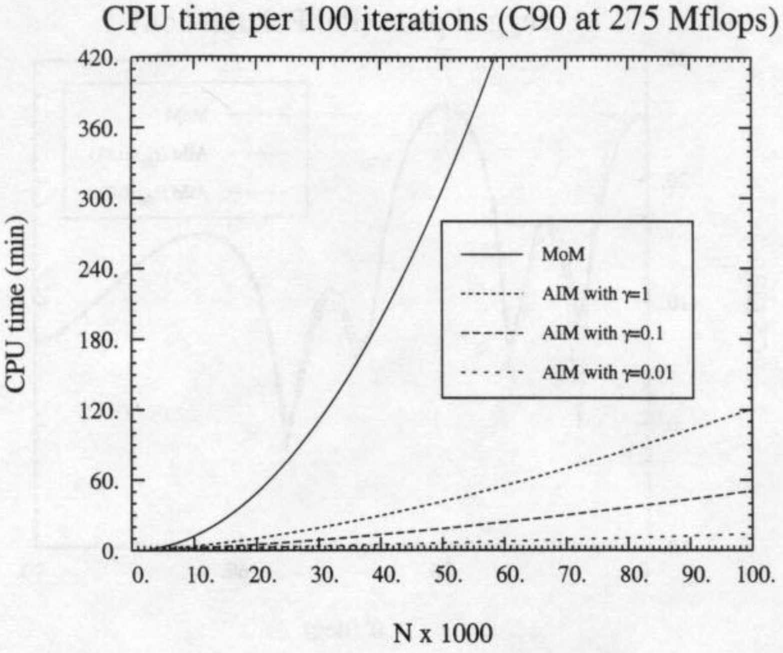

Figure 7. CPU time for standard MOM and AIM (surface problems with expansion order $M = 3$, grid step $h = l$, and $r_{\text{thr}} = 10l$).

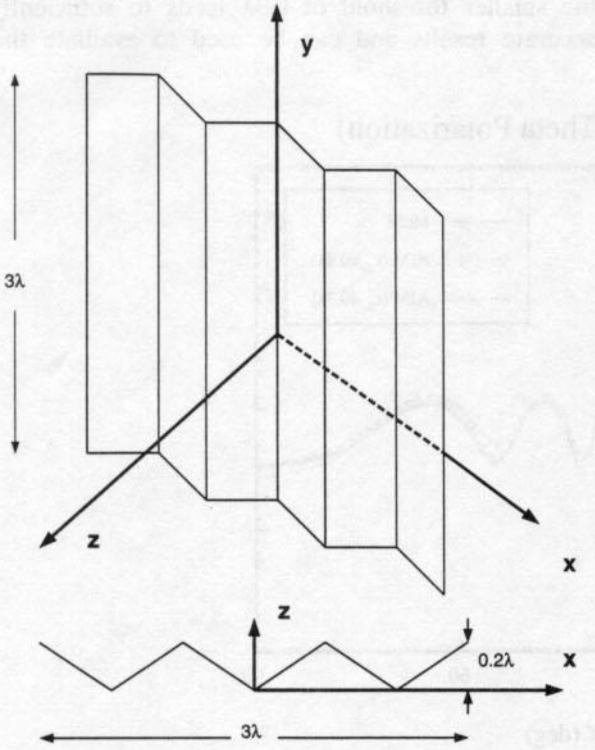

Figure 8. The jagged plate geometry.

3. The array is transposed, while still residing on all processors.

4. One-dimensional FFTs are then performed concurrently on all nodes along the third dimension.

5. At this point the transpose of the Fourier transform is available, and hence an additional transposition is required to complete the FFT routine.

Communication time consumption for a transposition across several processors is significant, and therefore in an optimized code any unnecessary transposition must be avoided. If the FFT output and input data must be in the same format, though, the data must be redistributed after the FFT calculation, resulting in a total of two transpositions.

The parallelized version of the code was tested on the IBM SP2 for the jagged plate geometry of Figure 8. The size of the plate was 3λ × 3λ × 0.2λ, implying a fairly flat FFT grid. We chose a rectangular grid spacing of 0.05λ for all directions and a moment expansion order of $M = 3$. For this discretization the original number of unknowns was $N = 3108$ and the number of grid points was $N_g = 32768$. The monostatic radar cross section (RCS) for this configuration is plotted in Figures 9 and 10 for the $\phi = 0°$ cut ($0 \leq \theta \leq 90°$) using 2° steps for both polarizations. Two

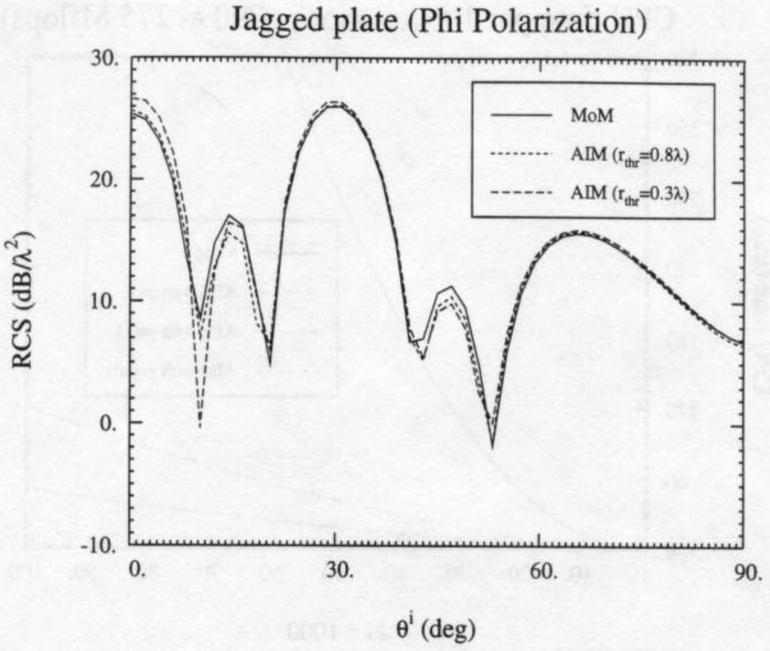

Figure 9. Monostatic radar cross section (RCS) for the jagged plate geometry (ϕ polarization).

sets of AIM results are shown: one where the near-field threshold was $r_{\text{thr}} = 0.8\lambda$ and the other for $r_{\text{thr}} = 0.3\lambda$. From (16), (17), and (21), both memory and complexity increase with r_{thr}, and thus a smaller r_{thr} is highly desirable, provided the accuracy of the solution is maintained. As seen from Figures 9 and 10 the smaller threshold of 0.3λ leads to sufficiently accurate results and can be used to evaluate the

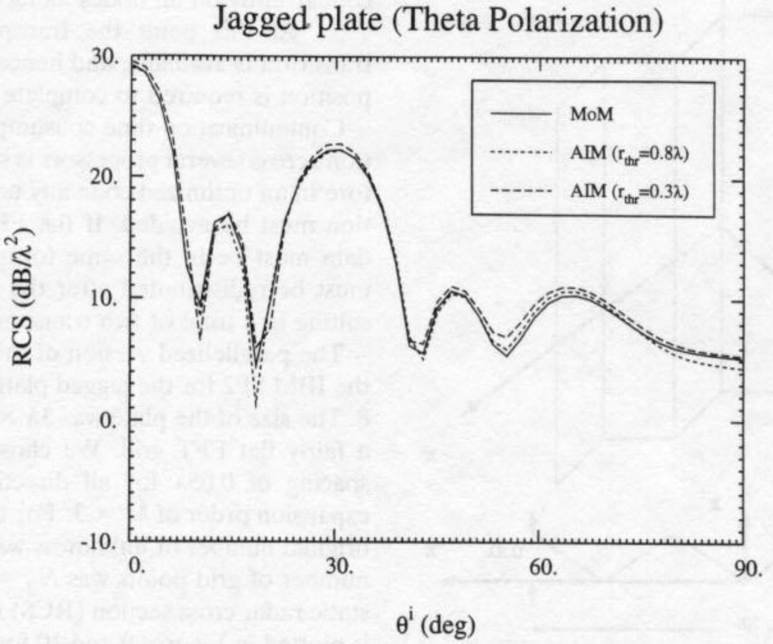

Figure 10. Monostatic RCS for the jagged plate geometry (θ polarization).

Table 1. CPU Time Comparisons for the Scatterer of Figure 8 With Zero Thickness

Number of Processors	Total Time, hours	Matrix-Vector Product Time, s	Percent of FFT in Total Time
1 (serial MOM)	22.3	5.42	0
1 (serial AIM)	29.2	5.36	70
2	30.4	5.64	69
4	18.7	3.12	55
8	13.8	2.04	45
16	11.2	1.48	29

performance of the AIM code. Tables 1 and 2 sum up the CPU time and storage comparisons between AIM and MOM, where in this case the jagged plate was assumed to have zero thickness. In terms of memory the AIM savings are dramatic, especially for smaller thresholds r_{thr}. In terms of CPU time, each matrix-vector product of the serial AIM algorithm is comparable to MOM, but significant speedup is gained as the number of processors is increased. For small but nonzero plate thickness, though, the advantages of AIM over MOM become much more impressive. For a similar jagged plate with nonzero thickness, the number of RWG unknowns is approximately doubled, resulting in quadruple memory and CPU time requirements for MOM. On the other hand, since the number of AIM grid points N_g remains the same, the corresponding AIM CPU time is virtually unaffected, and even the serial AIM code becomes 4 times faster than MOM. The AIM storage requirements are only slightly increased (see Table 3 for the thicker plate), whereas those of MOM grow beyond the capacity of conventional computational platforms.

5. Summary and Conclusions

In this paper we demonstrated the use and advantages of the adaptive integral method (AIM) for scattering by perfectly conducting and relatively flat but possibly irregular surfaces. The AIM concepts were specifically presented in a simplified manner for this geometry, and care was taken to justify certain mathematical assumptions and details neglected by *Bleszynski et al.* [1996]. Our investigation of AIM for relatively flat scatterers showed that the flat FFT grids provide dramatic reduction in complexity and memory requirements, and thus AIM is particularly efficient for such structures. We also developed closed-form expressions for the CPU time and storage requirements of AIM and included actual AIM performance data for a parallel version of the AIM scattering code.

Table 2. Memory Comparisons for the Scatterer of Figure 8 with Zero Thickness

Method	Memory, Mb
MOM	77.3
AIM ($r_{\text{thr}} = 0.8\lambda$)	28.94
AIM ($r_{\text{thr}} = 0.3\lambda$)	17.19

Table 3. Memory Comparisons for the Scatterer of Figure 8 With Nonzero Thickness

Method	Memory, Mb
MOM	309.2
AIM ($r_{\text{thr}} = 0.8\lambda$)	74.41
AIM ($r_{\text{thr}} = 0.3\lambda$)	27.41

Appendix: The Moments of the Basis Functions

Here we will demonstrate how the moments defined in (10) can be efficiently calculated using closed-form expressions, thus avoiding time-consuming numerical integrations.

Let the global coordinate system be denoted by (x, y, z), and the vertices $i = 1, 2, 3$ of a given triangle have coordinates (x_i, y_i, z_i). By transforming to local simplex coordinates [*Silvester and Ferrari*, 1991; *Anastassiu*, 1997], we have ($\mathbf{r}$, $\mathbf{r}_i$ are the position vectors to these points)

$$\mathbf{r} - \mathbf{r}_1 = (x_{21}\xi + x_{31}\eta)\hat{\mathbf{x}} + (y_{21}\xi + y_{31}\eta)\hat{\mathbf{y}} + (z_{21}\xi + z_{31}\eta)\hat{\mathbf{z}} \tag{22}$$

$$\mathbf{r} - \mathbf{r}_a = \mathbf{r} - \mathbf{r}_1 + \mathbf{r}_1 - \mathbf{r}_a = (x_{21}\xi + x_{31}\eta + x_{1a})\hat{\mathbf{x}} + (y_{21}\xi + y_{31}\eta + y_{1a})\hat{\mathbf{y}} + (z_{21}\xi + z_{31}\eta + z_{1a})\hat{\mathbf{z}} \tag{23}$$

where we have used the notation $x_{rs} \equiv x_r - x_s$, $y_{rs} \equiv y_r - y_s$, and $z_r - z_s$. To calculate the moments, it is sufficient to evaluate the integral

$$J^{(Q)} = \int\int_{T_0} (\lambda_{10}^{(0)}\xi + \lambda_{01}^{(0)}\eta) \prod_{q=0}^{Q} (\mu_{10}^{(q)}\xi + \mu_{01}^{(q)}\eta + \mu_{00}^{(q)})\, d\xi\, d\eta \tag{24}$$

where Q is the sum of the moment indices and T_0 is an isosceles triangle of unit lateral side. For example, to calculate the x component of $\tilde{\mathbf{M}}_{210}$, we let $Q = 2 + 1 + 0 = 3$ and

$$\lambda_{10}^{(0)} = x_{21}, \quad \lambda_{01}^{(0)} = x_{31}$$

$$\mu_{10}^{(0)} = 0, \mu_{01}^{(0)} = 0, \quad \mu_{00}^{(0)} = 1$$

$$\mu_{10}^{(1)} = x_{21}, \quad \mu_{01}^{(1)} = x_{31}, \quad \mu_{00}^{(1)} = x_{1a} \tag{25}$$

$$\mu_{10}^{(2)} = x_{21}, \quad \mu_{01}^{(2)} = x_{31}, \quad \mu_{00}^{(2)} = x_{1a}$$

$$\mu_{10}^{(3)} = y_{21}, \quad \mu_{01}^{(3)} = y_{31}, \quad \mu_{00}^{(3)} = y_{1a}$$

The integral $J^{(Q)}$ can be evaluated recursively in the following way: For $Q = 0$ we have

$$J^{(0)} = \frac{\lambda_{10}^{(0)} + \lambda_{01}^{(0)}}{6} \tag{26}$$

and since the integrand of $J^{(Q)}$ is the polynomial

$$P^{(Q)} = \sum_{i=0}^{Q+1} \sum_{j=0}^{Q+1} \lambda_{ij}^{(Q)} \xi^i \eta^j \tag{27}$$

it follows that

$$J^{(Q)} = \sum_{i=0}^{Q+1} \sum_{j=0}^{Q+1} \lambda_{ij}^{(Q)} \frac{i!j!}{(i+j+2)!} \tag{28}$$

Finally, the integrand of $J^{(Q+1)}$ is the polynomial

$$\begin{aligned} P^{(Q+1)} &= P^{(Q)}(\mu_{10}^{(Q+1)}\xi + \mu_{01}^{(Q+1)}\eta + \mu_{00}^{(Q+1)}) \\ &= \sum_{i=0}^{Q+2} \sum_{j=0}^{Q+2} \lambda_{ij}^{(Q+1)} \xi^i \eta^j \end{aligned} \tag{29}$$

where

$$\lambda_{ij}^{(Q+1)} = \lambda_{i-1,j}^{(Q)} \mu_{10}^{(Q+1)} + \lambda_{i,j-1}^{(Q)} \mu_{01}^{(Q+1)} + \lambda_{i,j}^{(Q)} \mu_{00}^{(Q+1)} \tag{30}$$

$$i > 0, j > 0$$

$$\lambda_{ij}^{(Q+1)} = 0 \qquad i \le 0 \text{ or } j \le 0 \tag{31}$$

References

Anastassiu, H. T., Electromagnetic scattering from jet engine inlets using analytical and fast integral equation methods, Ph.D. thesis, Univ. of Mich., Ann Arbor, 1997.

Bleszynski, E., M. Bleszynski, and T. Jaroszewicz, A fast integral equation solver for electromagnetic scattering problems, in *Proceedings of the International Symposium on Antennas and Propagation*, pp. 416–419, Inst. of Electr. and Electron. Eng., New York, 1994.

Bleszynski, E., M. Bleszynski, and T. Jaroszewicz, AIM: Adaptive integral method compression algorithm for solving large-scale electromagnetic scattering and radiation problems, *Radio Sci.*, *31*, 1225–1251, 1996.

Harrington, R. F., *Time-Harmonic Electromagnetic Fields*, McGraw Hill, New York, 1961.

Peters, T. J., and J. L. Volakis, Application of the conjugate gradient FFT method to scattering from thin planar material plates, *IEEE Trans. Antennas Propag.*, *36*, 518–526, 1988.

Rao, S. M., D. R. Glisson, and A. W. Wilton, Electromagnetic scattering by surfaces of arbitrary shape, *IEEE Trans. Antennas Propag.*, *30*, 409–418, 1982.

Sarkar, T. K., E. Arvas, and S. M. Rao, Application of FFT and the conjugate gradient method for the solution of electromagnetic radiation from electrically large and small conducting bodies, *IEEE Trans. Antennas Propag.*, *34*, 635–640, 1986.

Silvester, P. P., and R. L. Ferrari, *Finite Elements for Electrical Engineers*, Cambridge Univ. Press, New York, 1991.

Volakis, J. L., and K. Barkeshli, Applications of the conjugate gradient FFT method to radiation and scattering, in *PIER 5: Application of Conjugate Gradient Method to Electromagnetics and Signal Analysis*, chap. 6, pp. 159–239, Elsevier, New York, 1991.

H. T. Anastassiu, S. Bindiganavale, M. Smelyanskiy, and J. L. Volakis, Radiation Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan, 1301 Beal Avenue, Ann Arbor, MI 48109-2122. (e-mail: hristosa@engin.umich.edu; bssunil@engin.umich.edu; msmelyan@engin.umich.edu; volakis@engin.umich.edu)

(Received March 10, 1997; revised July 15, 1997; accepted July 28, 1997.)