JOURNAL OF COMPUTATIONAL PHYSICS **137,** 299–320 (1997)
ARTICLE NO. CP975802

A Parallel Finite-Volume Runge–Kutta Algorithm for Electromagnetic Scattering

Vineet Ahuja and Lyle N. Long

Department of Aerospace Engineering, Pennsylvania State University, University Park, Pennsylvania 16802
E-mail: lnl@psu.edu

Received May 21, 1996; revised May 27, 1997

A 3D explicit finite volume algorithm has been developed to simulate scattering from complex geometries on parallel computers using structured body conformal curvilinear grids. Most simulations for practical 3D geometries require a large number of grid points for adequate spatial resolution making them suitable to parallel computation. The simulations have been carried out using a multi-block/zonal approach in the message passing paradigm on the SP-2. Each zone is placed on a separate processor and interprocessor communication is carried out using the Message Passing Library/Interface (MPL/MPI). Integration of Maxwell's equations is performed using the four-stage Runge–Kutta time integration method on a dual grid. This method of integrating on a staggered grid gives enhanced dissipative and dispersive characteristics. A scattered field formulation has been used and the Liao boundary condition is used at the outer nonreflecting boundary. The far zone transformation has also been implemented efficiently, using specialized MPL functions to evaluate the far zone scattering results. Results show extremely good comparisons for scattering from the sphere and the ogive with the exact solution and standard FDTD type algorithms. Comparisons for nonaxisymmetric targets like the NASA almond with experimental data has also been found to be extremely good.

1. INTRODUCTION

Scattering problems in electromagnetics have been numerically modelled since the conception of Yee's leapfrog algorithm [1] in 1966. However, the production of staircasing errors [2] coupled with the difficulty in predicting the radar return for low RCS geometries encouraged the development of various finite difference

0021-9991/97 $25.00

[3–5], finite volume [6, 7], and finite element algorithms [8]. The complexity in the shapes of the scatterers being modelled has led to a transition from the use of Cartesian grids. Body-fitted structured grid algorithms are being used because they utilise curvilinear orthogonal grids that map the surface of the body exactly by a transformation between the physical and the computational domain. The development of robust multiblock grid generators has reduced the turnaround time required to generate structured grids around complex configurations. In the present paper we make use of the finite volume approach with the Runge–Kutta time integration method that has been successfully used in aerodynamic [11] and time dependent aeroacoustic type problems [13–15]. The finite volume time domain (FVTD) algorithm is based on the four-stage Runge–Kutta explicit time stepping method on a dual grid. All three electric field components are computed at the same grid points and three magnetic field components are solved on the same dual grid points. The reason for adopting a dual grid based approach is to take advantage of the inherent dissipation and minimize the addition of explicit artificial dissipation (typical of central differencing schemes).

Over the years, there has been a tremendous boost in the computational resources available, both in terms of speed and memory. The onset of parallel computing has made a huge impact on the choice of the algorithm and the type of mesh to be considered. In the recent past, there has been a considerable effort made in utilizing both the data parallel and message passing paradigms to solving FDTD/FVTD problems in computational electromagnetics. In [16] parallelization issues related to the Yee algorithm are discussed for the data parallel paradigm on the CM-5. Parallelization issues have also been discussed previously by the authors in [17] for the CM-200/CM-5 machines. Certain drawbacks of the data parallel paradigm are the overhead costs incurred in performing the near-to-far-field transformation and computation of the outer radiation boundary condition. Shang *et al.* [18] address the domain decomposition strategies for message passing type electromagnetic algorithms. Chris Rowell *et al.* [19] have also incorporated a zone based parallel strategy on the Intel Paragon for their flux difference algorithm. Nguyen and Hutchinson [20] have used the bicharacteristic form of the Maxwell's equations on parallel architectures supporting message passing. In this paper, we present a dual grid based approach where the message passing is reduced due to the fact that only the E-field variables need to be passed across processors, thus making the algorithm very efficient on parallel architectures that support message passing. Implementation of the algorithm has been carried out on the IBM SP-2 using Message Passing Library/Interface (MPL/MPI). The effort in this paper is primarily geared toward solving problems of practical interest in an efficient manner with reasonable turnaround time.

2. NUMERICAL MODEL

Maxwell's equations can be written in integral form as

$$\int\int_V \int \frac{\partial \mathbf{B}}{\partial t}\, dV = -\int\int_S \mathbf{n} \times \mathbf{E}\, dS \tag{1}$$

$$\int\int_V \int \frac{\partial \mathbf{D}}{\partial t}\, dV = \int\int_S \mathbf{n} \times \mathbf{H}\, dS. \tag{2}$$

Since Maxwell's equations are linear for most materials, the total electric and magnetic fields can be split up into their scattered and incident components for these materials and Maxwell's equations should independently hold for each of the above-mentioned components,

$$E^{\text{total}} = E^{\text{incident}} + E^{\text{scattered}}$$
$$H^{\text{total}} = H^{\text{incident}} + H^{\text{scattered}}.$$

As is defined in [5] the incident field here is assumed to travel in free space throughout the domain of the problem and the total field is supposed to propagate in free space outside the scatterer and in the media of the scatterer when propagating within the scatterer. The scattered field is defined as the resultant of the incident fields subtracted from the total fields. Scattered fields emanate from the scattering object when the incident waves impinge upon them. The above equations are solved only for the scattered fields as the analytical solution of the incident field is known. A dual grid approach is taken with the electric and magnetic field components being solved on separate grids. All the E-field components lie on the same grid point and all the H-field components lie on the same dual grid point. However, the dual grid points are offset from the regular or E-field grid points. Each dual grid point is constructed by taking the average of the coordinates of the corners of each E-field cell. As has been previously shown by Noack and Anderson [7], this type of staggering exploits the duality of Maxwell's equations since the time derivative of the electric field is directly dependent on the curl of the magnetic field and vice versa. Therefore a typical H-field cell has an E-field placed at its centroid so all the numerical fluxes for that cell are computed using the H-field variables at the edges and those are used to update the E-field at the centroid. Time integration is carried out using the four-stage Runge–Kutta time marching method:

$$\mathbf{Q}_i^m = \mathbf{Q}_i^n - \gamma_m\, \Delta t\, \mathbf{R}_i^{m-1}, \quad m = 1, 2, 3, 4,$$
$$\mathbf{Q}_i^{n+1} = \mathbf{Q}_i^{m=4}.$$

The time step is denoted by n and each stage of the Runge-Kutta method by m, where the coefficients are $\gamma_m = \frac{1}{4}, \frac{1}{3}, \frac{1}{2}, 1$, respectively:

$$\mathbf{Q_1} = \begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix}, \quad \mathbf{Q_2} = \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix}. \tag{3}$$

The residuals $\mathbf{R_i}$ are defined as

$$\mathbf{R_1} = \frac{1}{\Delta V}(\Delta\mathbf{F} + \Delta\mathbf{G} + \Delta\mathbf{K}) \tag{4}$$

$$\mathbf{R_2} = \frac{1}{\Delta V_d}(\Delta\mathbf{L} + \Delta\mathbf{M} + \Delta\mathbf{N}), \tag{5}$$

where

$$\Delta\mathbf{F} = \frac{1}{\mu}(\mathbf{S}^{\xi}_{i+1,j+1/2,k+1/2} \times \mathbf{E}^{\xi}_{i+1,j+1/2,k+1/2} - \mathbf{S}^{\xi}_{i,j+1/2,k+1/2} \times \mathbf{E}^{\xi}_{i,j+1/2,k+1/2})$$

$$\Delta\mathbf{G} = \frac{1}{\mu}(\mathbf{S}^{\eta}_{i+1/2,j+1,k+1/2} \times \mathbf{E}^{\eta}_{i+1/2,j+1,k+1/2} - \mathbf{S}^{\eta}_{i+1/2,j,k+1/2} \times \mathbf{E}^{\eta}_{i+1/2,j,k+1/2})$$

$$\Delta\mathbf{K} = \frac{1}{\mu}(\mathbf{S}^{\zeta}_{i+1/2,j+1/2,k+1} \times \mathbf{E}^{\zeta}_{i+1/2,j+1/2,k+1} - \mathbf{S}^{\zeta}_{i+1/2,j+1/2,k} \times \mathbf{E}^{\zeta}_{i+1/2,j+1/2,k}).$$

In the above expressions $\mathbf{S}^{\xi}$, $\mathbf{S}^{\eta}$, $\mathbf{S}^{\zeta}$ represent the projected surface areas of constant ξ, η, ζ faces, respectively. It should be noted that the electric field points used in the evaluation of the above fluxes lie in the center of the cell faces and are not computed directly from the integration process. Instead, they are extrapolated from the electric field points comprising the corners of the cell faces.

Kordulla and Vinokur [21] have proposed an efficient method of calculating cell volumes for three-dimensional flow predictions. The volume ΔV is calculated by partitioning the hexahedron cell into five tetrahedra and computing the volume of each tetrahedron separately. This decomposition helps avoid gaps and overlaps in computing cell volumes and is computationally very efficient.

The residual $\mathbf{R_2}$ is used to evaluate the E-field components and can be computed in a similar manner:

$$\Delta\mathbf{L} = -\frac{1}{\varepsilon}(\mathbf{Sd}^{\xi}_{i+1,j+1/2,k+1/2} \times \mathbf{H}^{\xi}_{i+1,j+1/2,k+1/2} - \mathbf{Sd}^{\xi}_{i,j+1/2,k+1/2} \times \mathbf{H}^{\xi}_{i,j+1/2,k+1/2}) \tag{6}$$

$$\Delta\mathbf{M} = -\frac{1}{\varepsilon}(\mathbf{Sd}^{\eta}_{i+1/2,j+1,k+1/2} \times \mathbf{H}^{\eta}_{i+1/2,j+1,k+1/2} - \mathbf{Sd}^{\eta}_{i+1/2,j,k+1/2} \times \mathbf{H}^{\eta}_{i+1/2,j,k+1/2}) \tag{7}$$

$$\Delta\mathbf{N} = -\frac{1}{\varepsilon}(\mathbf{Sd}^{\zeta}_{i+1/2,j+1/2,k+1} \times \mathbf{H}^{\zeta}_{i+1/2,j+1/2,k+1} - \mathbf{Sd}^{\zeta}_{i+1/2,j+1/2,k} \times \mathbf{H}^{\zeta}_{i+1/2,j+1/2,k}). \tag{8}$$

The expressions $\mathbf{Sd}^{\xi}$, $\mathbf{Sd}^{\eta}$, $\mathbf{Sd}^{\zeta}$ represent the projected surface areas of constant ξ, η, ζ faces of the dual grid respectively. It must be pointed out the subscripts i, j, k used in the calculation of the above fluxes are subscripts on the dual grid.

2.1. Dispersion and Dissipation Characteristics

Most central differencing algorithms require the addition of some sort of explicit artificial dissipation to damp out spurious numerical high frequency waves. Figure 1 shows the dissipation characteristics for a conventional nonstaggered second-

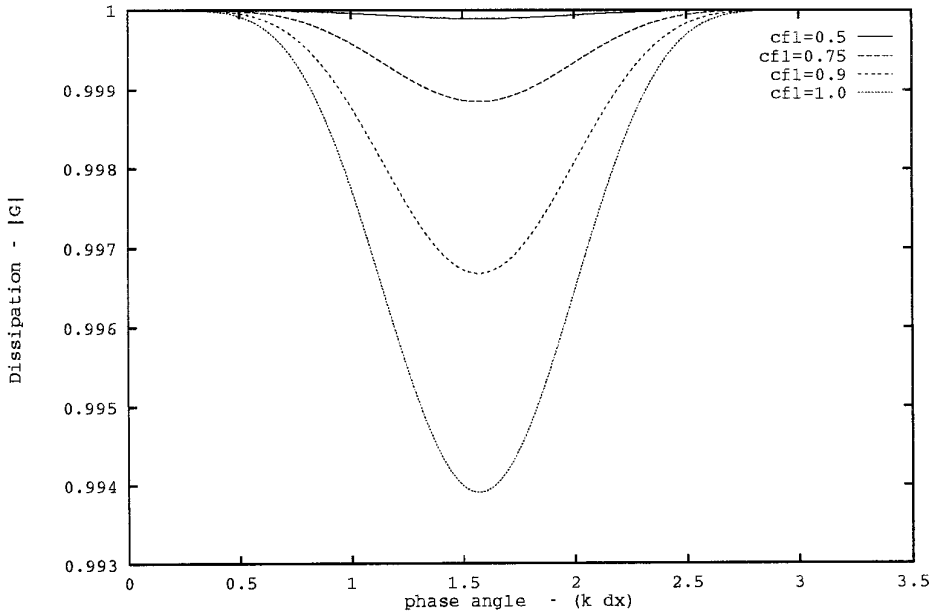

FIG. 1. Dissipation for a Runge–Kutta with central differencing.

order central differencing in space and four-stage Runge–Kutta time integration for the one-dimensional wave equation. It is evident that there is little or no damping of the high wavenumber components, thereby requiring the addition of explicit artificial dissipation.

However, for a spatially staggered scheme the dissipation characteristics with a four-stage Runge–Kutta method are quite different. This has been analysed with the help of a one-dimensional system of hyperbolic equations in [22]:

$$\frac{\partial}{\partial t}\begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} 0 & c \\ c & 0 \end{bmatrix} \frac{\partial}{\partial x}\begin{bmatrix} u \\ v \end{bmatrix} = 0. \tag{9}$$

The u and v variables were staggered in space so that the system can be written as

$$\frac{\partial}{\partial t} U_j = \begin{bmatrix} 0 & 0 \\ \dfrac{c}{\Delta x} & 0 \end{bmatrix} U_{j-1} + \begin{bmatrix} 0 & \dfrac{c}{\Delta x} \\ -\dfrac{c}{\Delta x} & 0 \end{bmatrix} U_j + \begin{bmatrix} 0 & -\dfrac{c}{\Delta x} \\ 0 & 0 \end{bmatrix} U_{j+1}, \tag{10}$$

where

$$U = \begin{bmatrix} u \\ v \end{bmatrix}.$$

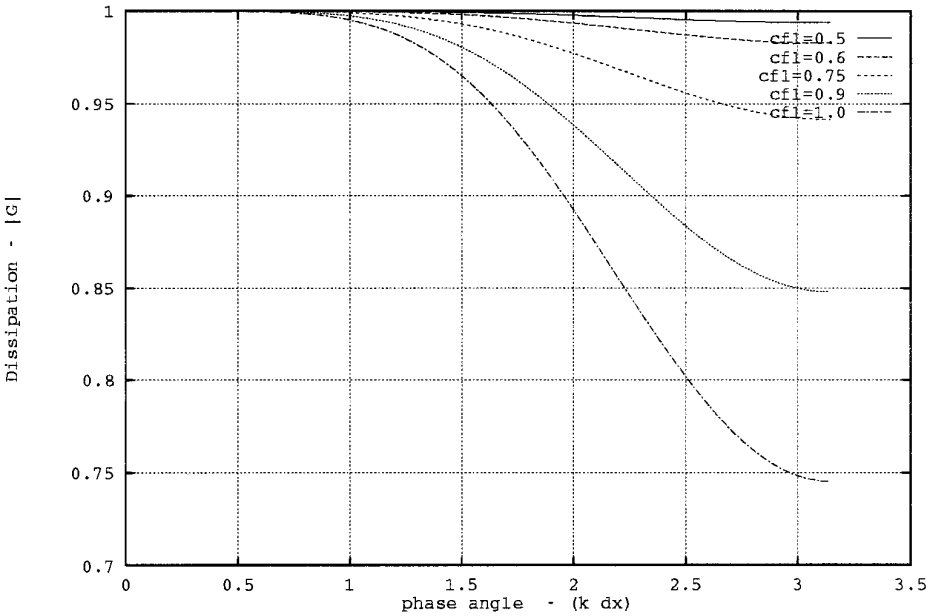

FIG. 2. Dissipation for a staggered system.

By analysing the solution of this system for a single harmonic,

$$U_j^n = \hat{U}_j^n e^{i\phi}, \tag{11}$$

and inserting it in Eq. (10) a relationship is obtained for the time discretization,

$$\hat{U}_j^{n+1} = G(\phi)\hat{U}_j^n, \tag{12}$$

where G is called the amplification matrix and for the system in Eq. (10) using four-stage Runge–Kutta becomes

$$G = \begin{bmatrix} 1 - 2\sigma^2\left(1 - \frac{\sigma^2}{3}\sin^2\left(\frac{k\Delta x}{2}\right)\right)\left(\sin^2\left(\frac{k\Delta x}{2}\right)\right) & \sigma\left(1 - \frac{2\sigma^2}{3}\sin^2\left(\frac{k\Delta x}{2}\right)\right)(e^{ik\,\Delta x} - 1) \\ \sigma\left(1 - \frac{2\sigma^2}{3}\sin^2\left(\frac{k\Delta x}{2}\right)\right)(1 - e^{-ik\,\Delta x}) & 1 - 2\sigma^2\left(1 - \frac{\sigma^2}{3}\sin^2\left(\frac{k\Delta x}{2}\right)\right)\sin^2\left(\frac{k\Delta x}{2}\right) \end{bmatrix},$$

where σ is the CFL number and can be defined as $c\Delta t/\Delta x$. The eigenvalues of the matrix G are

$$\lambda_{\pm} = \frac{1}{18}\left(18 - 36\sigma^2\sin^2\left(\frac{k\Delta x}{2}\right) + 12\sigma^4\sin^4\left(\frac{k\Delta x}{2}\right) \pm \sqrt{-144\sigma^2 + \sigma^2\cos^2(k\Delta x)\sin^2\left(\frac{k\Delta x}{2}\right)}\right).$$

The dissipation characteristics obtained from the spectral radius $|\lambda_+|$ of the amplification matrix G using the four-stage Runge–Kutta time integration is plotted against the phase angle $(k\Delta x)$ in Fig. 2. The low wavenumber components propagate without

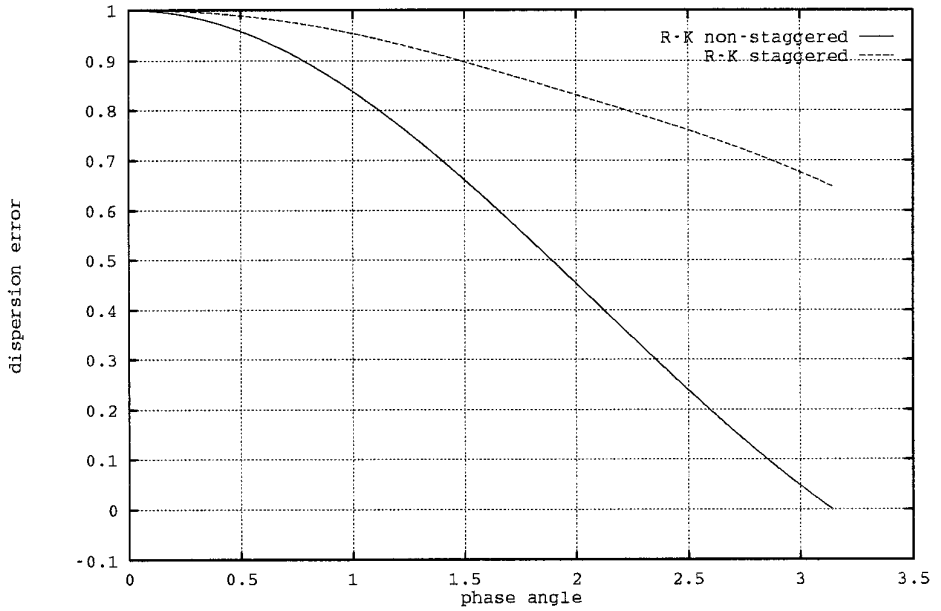

FIG. 3. A comparison of the dispersion characteristics.

being damped. However, the high frequency components are damped due to the grid induced dissipation caused by the staggering of the spatial derivatives. In most cases, this amount of grid induced dissipation is enough to suppress any high frequency numerical oscillations that may arise. These dissipation characteristics are quite commonly seen with implicit and explicit methods that utilize central differencing and have added artificial dissipation terms. Apart from improving the dissipation characteristics, spatial staggering is seen to have an impact on the dispersion characteristics too. In Fig. 3 a comparison is made between the dispersion error obtained for the one-dimensional wave equation on a nonstaggered grid using second-order central differences and a system of equations on a staggered grid for various phase angles ($k\Delta x$) at a CFL number of 1. The dispersion error for the staggered system is obtained from

$$\frac{\phi}{\phi_e} = \frac{1}{\sigma k\, \Delta x} \tan^{-1} \frac{\text{Imag}\, \lambda}{\text{Real}\, \lambda}. \tag{13}$$

Four-stage Runge–Kutta time integration has been used in both cases. It is seen that there is significant improvement of the staggered system over the nonstaggered equation and that the staggered approach can be utilised to resolve 12 cells per wavelength at 99% accuracy compared to the 20 cells per wavelength needed for a standard second-order central differencing algorithm. It should be pointed out that the above analysis is for a one-dimensional case on a uniformly spaced mesh with perfect staggering. In reality, however, body-fitted grids are usually clustered and may have imperfections like skewed cells or cells with high aspect ratios. Also, the staggering used in the algorithm is not completely perfect in three dimensions.

All the above reasons contribute in the production of spurious diffusion [23] for the high wavenumber components that are not completely damped by the grid-induced dissipation, hence the need for explicitly adding artificial dissipation. When required, fourth-order dissipation [24] is added,

$$D\mathbf{Q} = D_\xi\mathbf{Q} + D_\eta\mathbf{Q} + D_\zeta\mathbf{Q}, \tag{14}$$

where $D_\xi\mathbf{Q}$, $D_\eta\mathbf{Q}$, and $D_\zeta\mathbf{Q}$ are the fourth-order dissipation operators in the ξ, η, and ζ directions, respectively, and can be explicitly written as

$$D_\xi\mathbf{Q} = d_{i+1/2,j,k} - d_{i-1/2,j,k}, \quad D_\eta\mathbf{Q} = d_{i,j+1/2,k} - d_{i,j-1/2,k}, \quad D_\zeta\mathbf{Q} = d_{i,j,k+1/2} - d_{i,j,k-1/2},$$

where the above $d_{i\pm1/2,j,k}$ can be represented as

$$d_{i+1/2,j,k} = \frac{(\Delta V_d)_{i-1/2,j,k}}{\Delta t}\,\Omega(\mathbf{Q}_{i+2,j,k} - 3\mathbf{Q}_{i+1,j,k} + 3\mathbf{Q}_{i-1,j,k} - \mathbf{Q}_{i-2,j,k}).$$

The coefficient Ω is a constant whose value is $\frac{1}{256}$; hence inserting the expressions for artificial dissipation in Eq. (5) and rewriting it gives

$$\mathbf{R_2} = \frac{1}{\Delta V_d}(\Delta\mathbf{L} + \Delta\mathbf{M} + \Delta\mathbf{N} - D\mathbf{Q}). \tag{15}$$

2.2. Boundary Conditions

We have to consider the boundary conditions at the surface of the scatterer and the radiation condition at the outer boundary. In both cases we have only the E-field points located at the boundaries. The H-field points lie half a cell away from the boundaries and are considered as interior points. Hence, we have to determine the boundary conditions for only the E-field components.

2.2.1. *Surface Boundary Condition*

At the surface of a perfectly conducting scatterer we know that the tangential components of the total electric field are equal to zero,

$$\mathbf{n}_\eta \times \mathbf{E}^{\text{total}} = 0,$$

where $\mathbf{n}_\eta$ is the unit normal to the surface of the conductor.

The flux at the face that lies on the surface of the body can be exactly calculated by using the above condition and the analytically computed incident fields:

$$\mathbf{S}_\eta \times \mathbf{E}^{\text{scatt}} = -\mathbf{S}_\eta \times \mathbf{E}^{\text{incident}}.$$

Since the flux can be calculated directly the values of the E-field components on the surface of the conductor are needed only to solve for the fluxes of the perpendicular faces. To compute the fluxes on the faces perpendicular to the surface of the scatterer, the E-field components are required half a grid point above the surface

of the scatterer. These values are procured by applying Gauss's electric law to the cell adjacent to the surface of the scatterer. Two assumptions are taken into account here. First, the cell is assumed to be completely in free space; i.e., the lower surface of the cell is a small distance ε away from the surface of the scatterer. Second, the tangential components of the electric field on this lower surface are assumed to be the same as that on the surface of the scatterer. In this case, we use Gauss's electric law for a charge-free region such as free space in the integral form as

$$\int\int\int_V (\nabla \cdot \mathbf{D})\, dV = 0 \tag{16}$$

or

$$\int\int_S (\mathbf{n} \cdot \mathbf{D})\, dS = 0. \tag{17}$$

We need to apply Gauss's electric law to each cell that lies adjacent to the boundary. Using the above criterion the normal components of the electric field on the boundary can be approximated as

$$E^{\text{total}}_{x\,\text{normal}} = \frac{S^{\eta}_x(T)}{([S^{\eta}_x]^2 + [S^{\eta}_y]^2 + [S^{\eta}_z]^2)} \tag{18}$$

$$E^{\text{total}}_{y\,\text{normal}} = \frac{S^{\eta}_y(T)}{([S^{\eta}_x]^2 + [S^{\eta}_y]^2 + [S^{\eta}_z]^2)} \tag{19}$$

$$E^{\text{total}}_{z\,\text{normal}} = \frac{S^{\eta}_z(T)}{([S^{\eta}_x]^2 + [S^{\eta}_y]^2 + [S^{\eta}_z]^2)}, \tag{20}$$

where T is the Gauss divergence law summation over all the faces of the cell. The importance in implementing the boundary condition in such a rigorous manner is to ensure that the Maxwell's divergence equations are satisfied and no spurious solutions are generated at the boundary as is shown by Jiang *et al.* [25].

2.3. Far Field Boundary Condition

Most body conformal grid algorithms in electromagnetics [4–6] use a simple outer boundary condition that either solves the one-dimensional wave equation directly or is based on the method of characteristics in Cartesian coordinates. These boundary conditions are analogous to the Mur-type [26] boundary condition that is frequently used in computational electromagnetics. They generally work well when the outgoing wave is a plane wave travelling perpendicular to the boundary. In order to deal with waves that are incident to the boundary at arbitrary angles of incidence the Liao [27] formulation of transmitting boundary has been incorporated.

3. FAR ZONE TRANSFORM

The feasibility of schemes for Maxwell's equations depends on its ability to accurately predict the radar cross section (RCS) from targets of practical interest.

The implementation of the far zone is briefly discussed here because it is critical to the efficient parallelization of the algorithm. Since the estimation of RCS requires the intensity of the scattered wave at infinity, far beyond the outer boundary of the computational domain, the solution at the far field is usually calculated from the one in the near field with the help of a Green's function. This method of estimating the far field solution, based on a near to far field transformation, is analogous to the Kirchhoff method that is used to predict the linear acoustic far field in aerodynamic noise-related problems [29]. In that case, a control surface is defined in such a manner that it encompasses all nonlinear flow effects and noise sources, although the surface itself must lie in the linear region.

In [5] the authors have discussed the implementation of the far field solution in finite difference algorithms in great detail for both time harmonic solutions and wideband excitations. In this section we have used the method developed in [30] and modified it for the FVTD algorithm. It should be mentioned that, although the method developed in [30] is quite generic, the discussion of its application was limited to the Yee algorithm. The whole concept of the far zone transform is based on the fact that there exists a closed surface around the scatterer, where the electric and magnetic scattered surface currents, $\mathbf{j}_s$ and $\mathbf{m}_s$, can be calculated:

$$\begin{aligned} \mathbf{j}_s &= \mathbf{n} \times \mathbf{H}, \\ \mathbf{m}_s &= \mathbf{E} \times \mathbf{n}. \end{aligned} \tag{21}$$

In [30] the authors show that the retarded potentials $\mathbf{w}$ and $\mathbf{u}$ can be computed from the scattered surface currents using the formulas

$$\mathbf{w}[t - (\vec{r} \cdot \hat{r})/c] = \left(\frac{1}{4\pi c}\right) \frac{\partial}{\partial t} \left\{ \int \int_s \mathbf{j}_s(t)\, ds \right\} \tag{23}$$

$$\mathbf{u}[t - (\vec{r} \cdot \hat{r})/c] = \left(\frac{1}{4\pi c}\right) \frac{\partial}{\partial t} \left\{ \int \int_s \mathbf{m}_s(t)\, ds \right\}, \tag{24}$$

where

t = the elapsed time,

$\hat{r}$ = is the unit vector to the far zone field point,

$\vec{r}$ = is the vector to the source point of integration.

Assuming that the surface η = const denotes the integration surface for the far zone transform the equations can be recast in discretized form as

$$\mathbf{u}[t - (\vec{r} \cdot \hat{r})/c] = \left(\frac{1}{4\pi c}\right) \left((\mathbf{E} \times \mathbf{S}^{\eta})^{n+1} - (\mathbf{E} \times \mathbf{S}^{\eta})^{n} \right). \tag{25}$$

At each time step the contribution of each cell surface that makes up the integration surface η = const is calculated and placed in its corresponding $\mathbf{w}$ bin whose index is computed by subtracting the time delay factor from the elapsed time and dividing

the result by the time step Δt. It must be pointed out here that the right-hand side of Eq. (25) represents the flux terms and are already computed during the Runge–Kutta stages and need not be recomputed.

The retarded potential **w** has to be computed in a slightly different manner since the magnetic field points do not lie on the surface η = const. Instead, there exists a magnetic field point half a cell above and half a cell below the surface. Therefore we can take the average of the time derivatives of the cell above and below and compute the retarded potential **w**:

$$\mathbf{w}[t - (\vec{r}\cdot\hat{r})/c] = \left(\frac{1}{4\pi c}\right)\left(\mathbf{S}^{\eta} \times \frac{(\mathbf{H}^{n+1} - \mathbf{H}^{n})_{\eta+1/2} + (\mathbf{H}^{n+1} - \mathbf{H}^{n})_{\eta-1/2}}{2}\right). \quad (26)$$

The far zone scattered fields can be computed from the above potentials as

$$E_{\theta} = -Zw_{\theta} - u_{\phi}, \quad E_{\phi} = -Zw_{\phi} + u_{\theta},$$

where Z is the impedance of free space. The far zone scattered fields are then Fourier transformed and divided by the Fourier transform of the incident pulse to obtain the scattering cross section, which can be defined as

$$\sigma = \lim_{r'\to\infty} 4\pi(r')^2 \left|\frac{E^{\text{scatt}}(r')}{E^{\text{inc}}}\right|^2. \quad (27)$$

4. PARALLELIZATION ISSUES AND PERFORMANCE

Some of the features that make this algorithm suitable for parallelization are its explicit character, the narrow stencil size, and the dual grid-based approach. In [17] these features were exploited in a data parallel manner for the CM-200/CM-5 machines. Although the implementation was quite efficient, there was an overhead computing outer radiation boundary conditions and the far zone transformation. In [17] an efficient send-with-add type structure was used in the implementation of the far zone transformation it was still found to be quite expensive due to the inherently serial nature of the transformation.

Parallelization of the algorithm was performed on the SP-2 using the MPL/MPI. The computer that was used for the message passing algorithm is the IBM 9076 Scalable Power parallel system (SP-2) which is a tightly clustered group of RISC system/6000 processors. Parallelization was effected by domain decomposition techniques and the algorithm was made efficient by minimizing the communication-to-computation ratio and balancing the workload effectively by partitioning the data evenly.

4.1. Implementation in MPL/MPI

Parallelization of the algorithm was performed utilizing the single program multiple data (SPMD) model wherein the same program is executed on all processors.

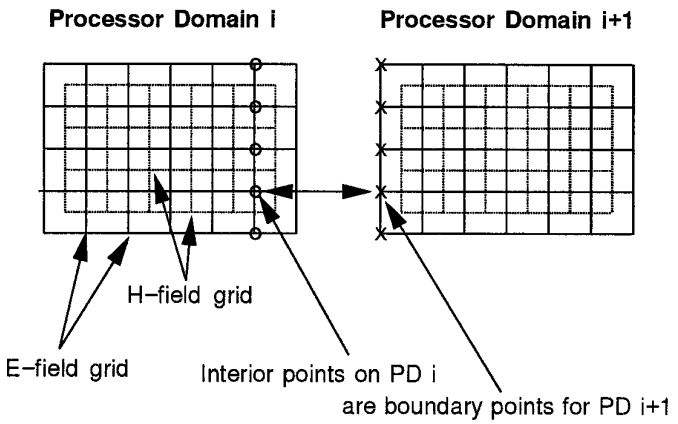

FIG. 4. Processor domain decomposition.

However, each processor has its identifier processor number and this permits MPL/MPI to have processor-based conditional statements.

Each processor has a certain part of the grid located on it. For the sake of brevity, let us call this the processor domain (PD). In most cases, the PD is synonymous with a zone of the grid. Since the grid for the algorithm discussed in the previous section consists of the E-field grid and the H-field dual grid, a PD for our purposes is a combination of the two. Message passing is carried out between two adjacent PD's by creating pseudo boundaries or ghost cells between them. As is evident from Fig. 4 the interior cell values (E-fields in this case) are passed from one processor to its neighboring processor and they constitute the pseudo boundary conditions for the neighboring processor and vice versa. Each processor has its own processor domain to work with. Each processor domain is either bounded by pseudo boundaries or a combination of physical and pseudo boundaries. Domain decomposition in allocating PD's is very important and careful consideration has to be paid to issues of load balancing, synchronization, distribution of boundaries over processors, and the amount of message passing. The domain decomposition strategy used in most cases presented in this paper is that of 1D parallelization. For example, for the case of the ogive the PD consisted of (ni $\times$ jmax $\times$ kmax) grid points, where ni refers to the number of *i*-planes in the PD. As has been pointed out in [18] this sort of partitioning achieves near perfect load balancing. Since only nearest neighbor information is required by the algorithm for flux computation, message passing between two contiguous domains is now only (3 $\times$ jmax $\times$ kmax) E-field variables. The dual grid formulation provides the added advantage of having to pass only the E-field variables across processors. As is evident in Fig. 4 the interior cell values from one processor are passed to the neighboring processor and they act as the pseudo boundary values for the neighboring processor and vice versa. This leads to an overlap of one layer of cells between processors and is really the only overhead incurred in parallelizing the code.

The far zone transformation was implemented by identifying a far zone surface

TABLE I
Performance of Codes in μs/Node/Timestep

Algorithm	Computer	Paradigm	μs/node/timestep
FDTD	RS/6000	Serial	14.0
FVTD	RS/6000	Serial	94.0
FDTD	CM-5	Data parallel	0.6
FDTD w/o farzone	CM-5	Data parallel	0.1
FVTD	CM-5	Data parallel	19.92
FVTD w/o farzone	CM-5	Data parallel	17.32
FVTD	SP-2 (4-node)	Message passing	12.76
FVTD	SP-2 (16-node)	Message passing	3.99

and all nodes transferred their contribution of the retarded potentials to the master node, which then combined the potentials and calculated the far zone scattered fields.

4.2. Performance Results

In this section the performance of the FVTD algorithm will be compared to the FDTD algorithm [5] in terms of the computational time required.

Table I shows the time taken per grid point per iteration for both the FVTD and FDTD codes. In the data parallel mode the FDTD algorithm is 33.2 times faster than the FVTD algorithm on a per cell per timestep basis. This is attributed to the fact that the FDTD is run on Cartesian grids, whereas the FVTD algorithm is used on curvilinear grids and the additional cost of the transformation from the physical to the computational grid is incurred while computing the numerical fluxes. Also, the FVTD algorithm has a four-stage time integration process which is a further overhead. The numbers in Table I can be misleading because of two reasons: First, the FVTD algorithm requires fewer cells per wavelength than the FDTD algorithm. Second, the curvilinear nature of the FVTD grid requires far less cells in the computational domain than is required by the stair-stepped Cartesian grid needed for the FDTD algorithm. Table II shows the number of grid points required by the two algorithms to adequately resolve the scattering wave patterns for an ogive shaped target.

Here, it can be clearly seen that approximately 20 times more grid cells are required by the FDTD algorithm over the FVTD algorithm. This shows that even

TABLE II
Grid Point Comparison for the Ogive

Algorithm	Grid dimensions	Total grid points
FDTD	$400 \times 100 \times 100$	4 Million
FVTD	$200 \times 31 \times 32$	198,400

though the FDTD algorithm is much faster than the FVTD algorithm, for certain problems the FVTD approach may still be very efficient.

It is also found from the data presented in Table I that the message passing paradigm is better suited to the FVTD algorithm over the data parallel mode. A 16-node SP-2 execution of the FVTD code was five times faster than a 32-node run on the CM-5, although the peak speeds of these two machines are roughly equal. This indicates that the FVTD algorithm may be more efficient as a message passing based algorithm.

5. RESULTS

5.1. Scattering from a Sphere

The first test case considered here is that of the scattering of a wideband pulse from a perfectly conducting sphere. The grid used for this case is an O–O type grid and consisted of $37 \times 46 \times 73$ grid points. The sphere is of radius 0.1 m. The incident pulse is represented by the derivative of a Gaussian and can be explicitly written as

$$f(t) = -2\alpha(\tau - \beta\,\Delta t)e^{-\alpha(\tau-\beta\,\Delta t)^2}, \tag{28}$$

where τ is the time delay parameter,

$$\tau = t - (\vec{r}\cdot\hat{r})/c,$$

and

$$0 < \tau < 2\beta\,\Delta t,$$

$$\alpha = \left(\frac{4}{\beta\,\Delta t}\right)^2,$$

and β is a user-defined parameter that dictates the width of the pulse. The value of β used for the FVTD run in this case was 756. It should be mentioned at this point that the FVTD results are being compared to the results obtained with the PSU-FDTD code [5] that employs the Yee algorithm. An equivalent β of 64 was used in the FDTD run. (The FDTD run was made with a cell size of 0.01 m or 10 cells per radii). The difference in the β value can be attributed to the difference in the time step, because the value of $\beta\,\Delta t$ has to be the same for both codes. Since the value of β defines the spectral bandwidth it has to be chosen judiciously so as not to create any unnecessary noise. The highest frequency for which the RCS is computed corresponds to a wavelength roughly equal to the radius of the sphere that would be resolved by 14 cells per wavelength. It should be noted here that the FDTD grid consisted of 262,144 grid cells, which is almost double the FVTD grid of 124,246 grid points.

Near zone (i.e., points within the computational space) and far-field comparisons of the FVTD and FDTD solutions were made in order to ascertain the validity of

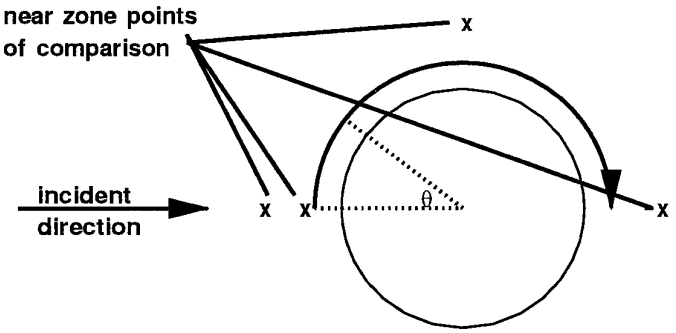

FIG. 5. Points of near zone comparison for the sphere.

the FVTD algorithm. For the near zone comparisons, two points were chosen in the backscatter direction at distances of 0.02 m and 0.04 m from the surface of the sphere. One point was sampled at an angle of 90° from the direction of incidence and another point of comparison was chosen in the shadow region behind the sphere at an angle of 180° (Fig. 5). All four near zone comparisons were found to be in excellent agreement [22]. Figure 6 shows the near zone comparisons of the E_z component at two locations. The comparison in the first plot in Fig. 6 is taken at a location 0.04 m from the surface of the sphere in the backscatter direction and the second is at an angle of $\theta = 180°$. Both figures clearly indicate that the FVTD algorithm captures the creeping wave behind the main component of the reflected wave. Figure 7 is a comparison of the backscatter cross section against frequency of the FVTD and FDTD algorithms and the exact solution. The exact solution for the monostatic radar cross section can be written as

$$\sigma_{3d} = \lim_{r\to\infty} 4\pi r^2 \frac{|\mathbf{E}^s|^2}{|\mathbf{E}^i|^2} = \frac{\lambda^2}{4\pi}\left|\sum_{n=1}^{\infty} \frac{(-1)^n(2n+1)}{\hat{H}_n^{(2)'}(kR)\hat{H}_n^{(2)}(kR)}\right|^2 \tag{29}$$

and

$$\hat{H}_n^{(2)'}(kR) = \frac{\partial \hat{H}_n^{(2)}(kR)}{\partial(kR)},$$

where

$\hat{H}_n^{(2)}(kR)$ is the spherical Hankel function of the second kind,

λ is the wavelength,

k is the wavenumber, and

R is the radius of the sphere.

The comparison between the exact solution and the FVTD solution is exceptionally good, especially at low frequencies and it is seen to be an improvement over the FDTD solution.

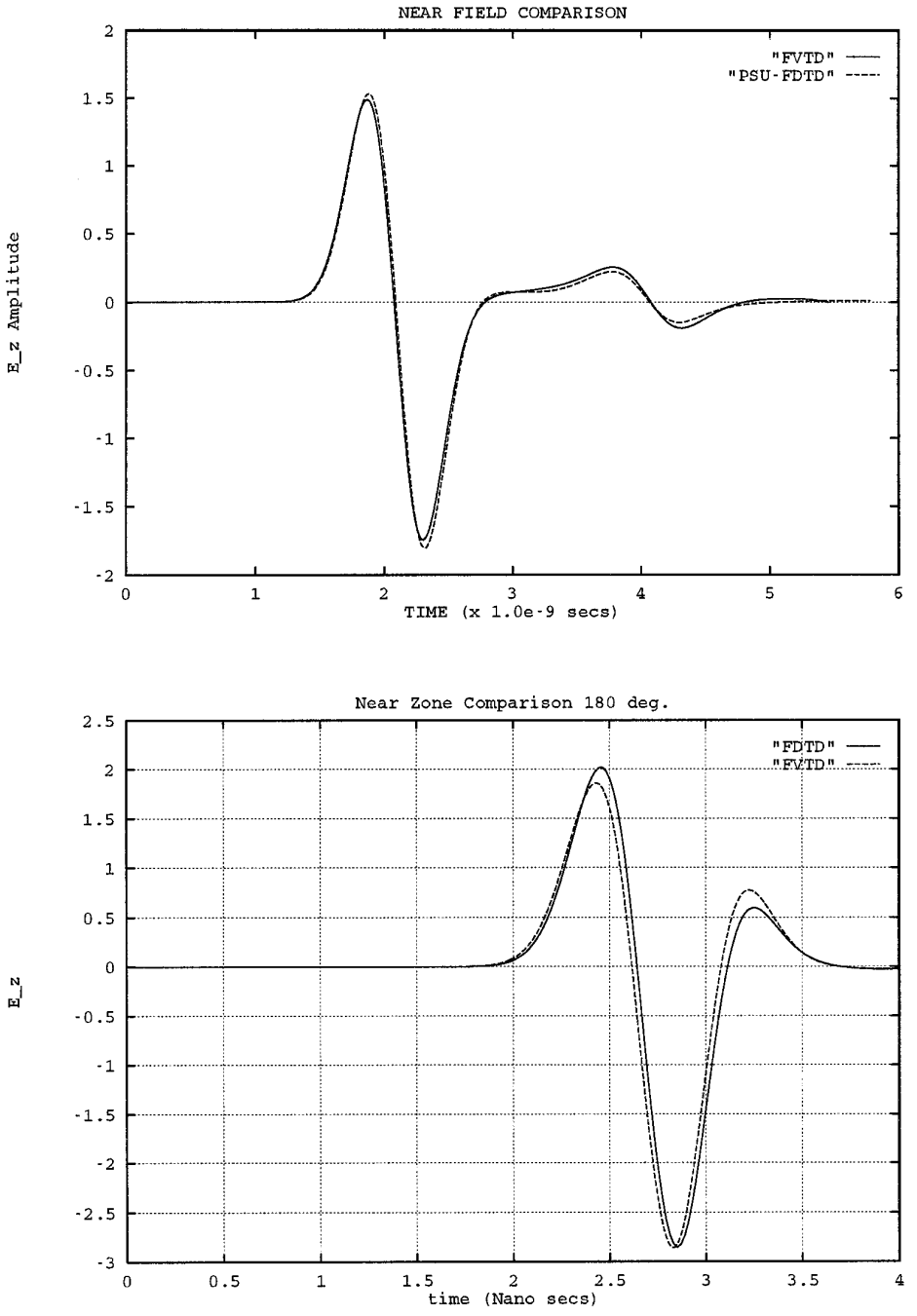

FIG. 6. Scattered field comparison for the sphere in the backscatter and shadow regions.

5.2. Scattering from an Ogive

The shaping of aerodynamic vehicles based on electromagnetic and radar considerations is an important aspect of the design process. Shaping the nose of an aeroplane/missile is of primary importance in reducing the effective area that is

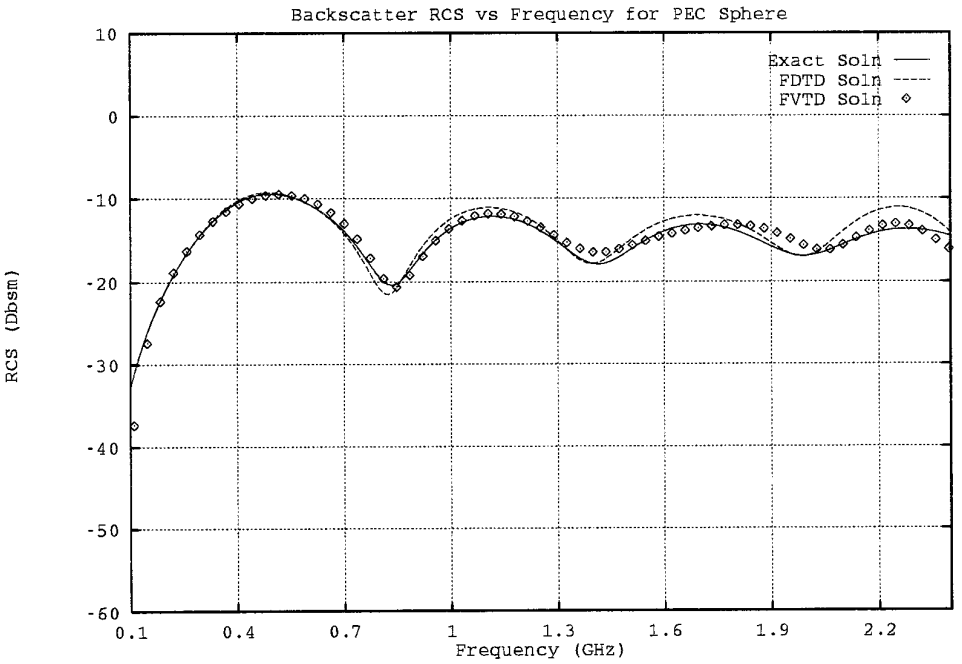

FIG. 7. RCS for the sphere.

visible to the radar. Most nose sections are ogive-shaped, although some may be spheroids or ellipsoids. This makes it important to study the scattering effects and to evaluate the RCS for ogives at nose-on incidence. Therefore, the next test case investigated is that of a metallic ogive. The ogive has a half angle of 22.62° and an aspect ratio of 5:1. The length of the ogive is 10 in. and is a standard EMCC (electromagnetic code consortium) benchmark test. The grid consisted of 200 × 31 × 32 cells and is clustered near the leading and trailing edges and close to the surface of the scatterer. The grid is generated by first constructing a planar grid and rotating it about the axis of symmetry. In order to resolve the singular point at the tip of the ogive, special cells have to be constructed around the points lying on the axis of symmetry. These cells are formed from the dual grid points that lie adjacent to the axis of symmetry. Needless to say, these special cells are $(n + 2)$-sided polyhedrons, where n is the number of grid cells in the rotational direction. A zonal approach is taken in solving this problem and the entire computational domain is spread out over eight processors of the IBM SP-2 equally.

The far zone integration surface is located four grid points from the surface of the ogive. A wideband Gaussian pulse is incident normal to the leading edge and is defined as

$$f(t) = e^{-\alpha(\tau-\beta\Delta t)^2}, \tag{30}$$

where β is chosen such that the frequency content of the incident pulse extends up to 5 GHz. The direction of propagation of the incident pulse is parallel to the

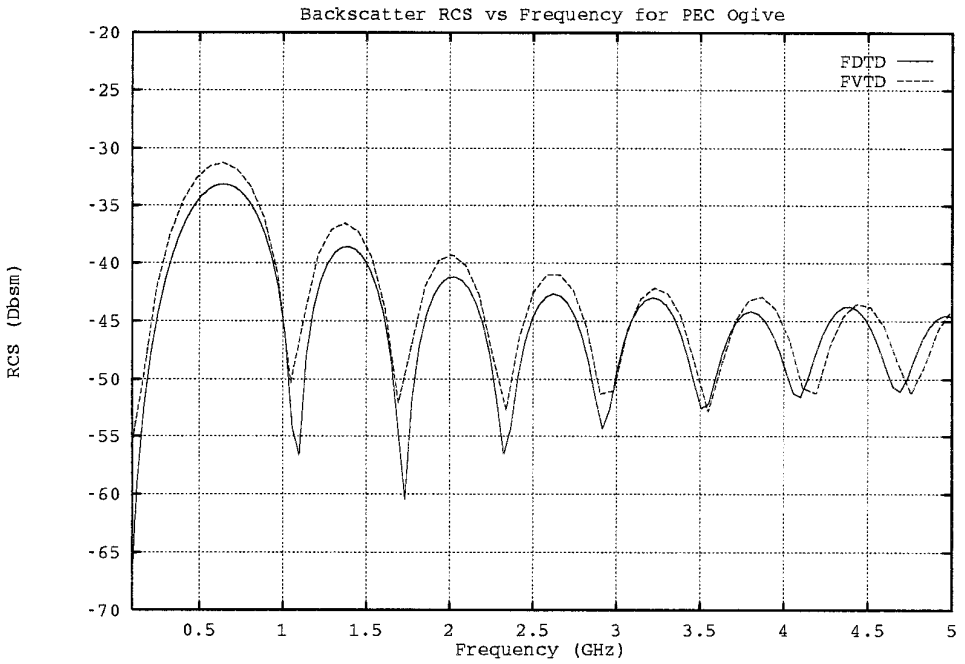

FIG. 8. RCS for the ogive.

major axis of the ogive. In Fig. 8 the co-polarized RCS is plotted versus frequency for the FVTD and FDTD algorithms. In order to adequately resolve the sharp corner at the tip of the ogive the FDTD simulation required up to four million grid points. The FDTD solution has a resolution of 80 cells per wavelength at 5 Ghz, whereas the corresponding FVTD solution has been obtained with a resolution of only 20 cells per wavelength at the upper end of the frequency spectrum. The reason for such a high resolution FDTD solution was the need to resolve the sharp edge adequately with a stair-stepped grid. The comparison shown in Fig. 8 indicates that the FVTD and the FDTD RCS data maintain the same trends with variation in frequency and are in extremely good agreement in almost the entire frequency range that has been resolved.

5.3. Scattering from the Almond

One of the most important EMCC benchmark targets is the NASA almond. It is important partly because of the nature of its geometry and, also, because it remains one of the more difficult bodies for which to procure an accurate RCS. Looking at it from a cross-sectional view, it resembles a circular-arc airfoil. However, it has a finite thickness that is rapidly varying in the spanwise direction. The almond is 9.936 in. long and its geometrical specifications are defined in [31]. Two grids were employed for all the cases run for the almond. One of the grids has $198 \times 20 \times 57$ grid points and the other consists of $198 \times 45 \times 57$ grid points (see Fig. 9). A sinusoidal clustering of grid points was used along the rotational direction.

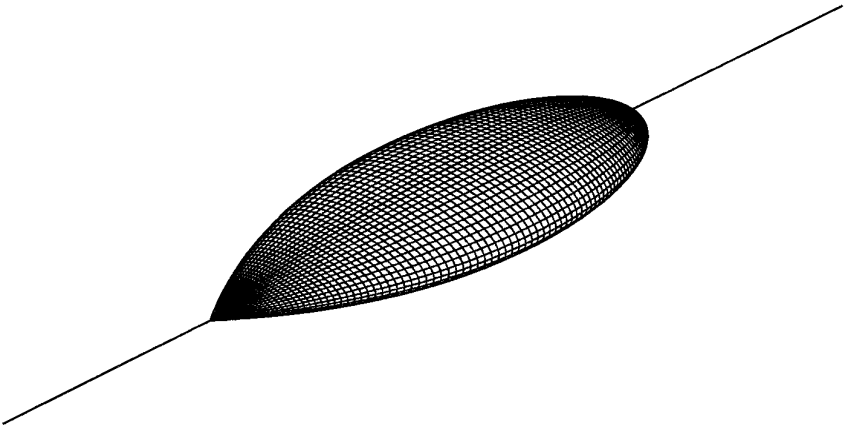

FIG. 9. Surface grid for the NASA almond.

The clustering was necessary to avoid getting high aspect ratio cells. One-dimensional domain decomposition was employed in all cases along the longest dimension. The computational domain was spread across seven processors equally on the IBM SP-2. The far zone integration surface was defined six cells away from the surface of the almond. Due to the nature of the grid used for this test case, odd–even decoupling was observed. It was suppressed by adding fourth-order artificial dissipation, as previously discussed in Section 3. Scattering from the almond was investigated for all angles of incidence from 0° to 180° at intervals of 15° at a frequency of 1.19 GHz.

The monostatic RCS computed was compared to experimental results obtained for vertical/vertical polarization by Woo *et al.* [31]. For experimental measurements the metallic almond target was made of aluminium. The experimental results have been plotted by sweeping through the rotational ϕ direction at 5° intervals. A comparison of the FVTD and experimental results is shown in Fig. 10. Each FVTD point in Fig. 10 represents a FVTD simulation for the angle of incidence represented by ϕ on the abcissa scale. The FVTD computational results are in very good agreement with the experimental observations and are mostly within a 2-db agreement margin at all angles, except those at 45° and 135°, where the algorithm seems to underpredict and overpredict the measured RCS, respectively.

6. CONCLUSIONS

A time domain algorithm has been presented to solve Maxwell's equations over generic body conformal grids for three-dimensional geometries. The dual nature of Maxwell's equations are well exploited in that the algorithm uses a dual mesh. The staggered nature of the algorithm gives it better dispersion and dissipation characteristics. With a second-order staggered scheme resolution normally provided by fourth-order nonstaggered schemes can be obtained. In most cases, the dissipation provided by the grid staggering precludes the need to explicitly add any artificial dissipation. The message passing approach provided more control and flexibility in

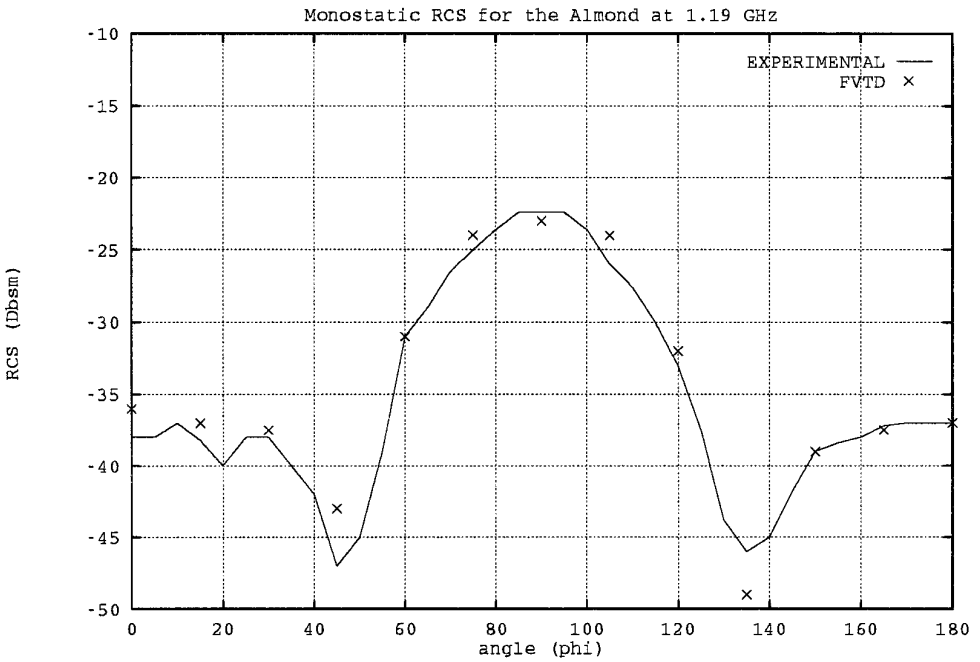

FIG. 10. RCS for the NASA almond—VV polarization.

distributing data over the computational resource and load balancing. The far field transformation, too, was better suited to a message passing environment. The dual nature of the grid aided in parallelization, since only one of the fields had to be communicated between processors. RCS backscatter results for the sphere and the ogive are in excellent agreement with the exact solution and solutions obtained using the standard FDTD Yee-type algorithm. Comparisons of results obtained for scattering from the almond are in good agreement with experimental observations.

REFERENCES

1. K. S. Yee, Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, *IEEE Trans. Antennas Propag.* **AP-14**(3), (1966).
2. R. Holland, Pitfalls of staircase meshing, *IEEE Trans. Electromagn. Compat.* **38**(12), (1993).
3. A. Taflove and K. R. Umashankar, The finite-difference time-domain method for electromagnetic scattering and interaction problems, *Journal of Electromagnetic Waves and Applications* **1,** 243 (1987).
4. P. M. Goorjian, Algorithm development for Maxwell's equations for computational electromagnetism, AIAA Paper 90-0251, Reno, Nevada, Jan. 1990.
5. K. S. Kunz and L. J. Luebbers, *The Finite-difference Time Domain Method for Electromagnetics,* CRC Press, Boca Raton, FL, 1993.
6. V. Shankar, A gigaflop performance algorithm for solving Maxwell's equations of electromagnetics, AIAA Paper 91-1578, Honolulu, June 1991.
7. R. W. Noack and D. A. Anderson, Time domain solutions of Maxwell's equations using a finite volume formulation, AIAA Paper 92-0451, Reno, Nevada, Jan. 1992.

8. N. K. Madsen and R. W. Ziolkowski, A three-dimensional modified finite volume technique for Maxwell's equations, *Electromagnetics* **10,** 147 (1990).
9. A. Shostko, Three dimensional parallel unstructured grid generation, AIAA Paper 94-0418, Reno, Nevada, Jan. 1994.
10. R. Lohner and P. Parikh, Three dimensional grid generation by the advancing front method, *Int. J. Numer. Methods Fluids* **8,** (1988).
11. Z. Weinberg and L. N. Long, A massively parallel solution of the three dimensional Navier–Stokes equations on unstructured adaptive grids, AIAA Paper 94-0760, Reno, Nevada, Jan. 1994.
12. T. J. Barth and D. C. Jespersen, The design and application of upwind schemes on unstructured meshes, AIAA Paper 89-0366, Reno, Nevada, Jan. 1989.
13. Y. Ozyoruk and L. N. Long, A new efficient algorithm for computational aeroacoustics on parallel computers, *J. Comput. Phys.* **125**(1), 135 (1996).
14. T. S. Chyczewski and L. N. Long, A higher order accurate parallel algorithm for aeroacoustic applications, AIAA Paper 94-2265, Colorado Springs, Colorado, June 1994.
15. A. Bangalore, P. J. Morris, and L. N. Long, A nonlinear disturbance equation method for supersonic jet noise, AIAA Paper 96-1728, State College, PA, May 1996.
16. Z. M. Liu, A. S. Mohan, T. A. Aubrey, and W. R. Belcher, Parallelized FDTD for antenna radiation pattern calculations, in *11th Annual Review of Progress in Applied Computational Electromagnetics, Monterey, CA, March 1995.*
17. V. Ahuja and L. N. Long, A FVTD algorithm for Maxwell's equations on massively parallel machines, in *11th Annual Review of Progress in Appl. Comput. Electromagnetics, Monterey, CA, March 1995.*
18. J. S. Shang, D. A. Calahan, and B. Vikstorm, Performance of a finite volume CEM code on multicomputers, AIAA Paper 94-0236, Reno, Nevada, Jan. 1994.
19. C. Rowell, V. Shankar, W. F. Hall, and A. Mohammadian, Advances in time-domain CEM using massively parallel architectures, in *11th Annual Review of Progress in Applied Computational Electromagnetics, Monterey, CA, March 1995.*
20. B. T. Nguyen and S. A. Hutchinson, The upwind leapfrog algorithm scheme for 3-D electromagnetic scattering and its implementation on two massively parallel computers, in *10th Annual Review of Progress in Applied Computational Electromagnetics, Monterey, CA, March 1994.*
21. W. Kordulla and M. Vinokur, Efficient computation of volume in flow predictions, *AIAA Journal* **21**(6), 917 (1983).
22. V. Ahuja, *A Finite Volume Time Domain Method for Maxwell's Equations on Parallel Computers,* Ph.D. dissertation, The Pennsylvania State University, 1995.
23. R. Vichnevetsky, Wave propagation analysis of difference schemes for hyperbolic equations: A review, *Int. J. Numer. Methods Fluids* **7** (1987).
24. A. Jameson, W. Schmidt, and E. Turkel, Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes, AIAA Paper 81-1259, Palo Alto, California, June 1981.
25. B. Jiang, J. Wu, and L. A. Povinelli, The origin of spurious solutions in computational electromagnetics, *J. Comput. Phys.* **125,** 104 (1996).
26. G. Mur, Absorbing boundary conditions for finite-difference aproximation of the time-domain electromagnetic field equations, *IEEE Trans. Electromagn. Compat.* **EMC-23,** 1073 (1981).
27. Z. P. Liao, H. L. Wong, G. P. Yang, and Y. F. Yuan, A transmitting boundary for transient wave analysis, *Sci. Sinica* **28**(10), 1063 (1984).
28. V. Ahuja and L. N. Long, A message passing finite volume algorithm for Maxwell's equations on parallel machines, AIAA Paper 95-1967, San Diego, California, June 1995.
29. Y. Ozyoruk and L. N. Long, Computation of sound radiating from engine inlets, *AIAA Journal* **34**(5), (1996).
30. R. J. Luebbers, K. S. Kunz, M. Schneider, and F. Hunsberger, A finite-difference time-domain near zone to far zone transformation, *IEEE Trans. Antennas Propag.* **39**(4), (1991).

31. A. C. Woo, H. T. G. Wang, and M. J. Schuh, Benchmark radar targets for the validation of computational electromagnetics programs, *IEEE Antennas Propag. Magazine* **35**(1), (1993).
32. W. M. Chan, User guide for hypgen—version 1.3, 1993.
33. A. E. Fuhs, *Radar Cross Section Lectures, Am. Inst. Aeronaut. Astronaut.,* New York, 1988.