Practical 3-D Contour/Staircase Treatment of Metals in FDTD

J. Anderson, M. Okoniewski, *Member, IEEE,* and S. S. Stuchly, *Fellow, IEEE*

Abstract—**The standard finite-difference time-domain (FDTD) method incorporates an inaccurate staircase representation of perfect electric conductors. More accurate, contour-type update formulas have been proposed. These approaches suffer from bookkeeping complexity, difficulties in mesh generation, and stability problems. Only simplified special cases have been implemented in three dimensions. A new contour-like FDTD algorithm is presented. Subcell formulas and staircase logic are combined to produce a three-dimensional (3-D) algorithm that is simple, robust, and fully automatic.**

I. Introduction

SUBCELL treatments of curved PEC surfaces in a structured mesh proposed by various authors have been reviewed by Yee [1]. In contour finite-difference time-domain (CFDTD) [2], [3] and related algorithms [4], nodes near metal are updated using subcell formulas derived from the integral form of Faraday's law. Only simplified special cases have been implemented in three dimensions, due to problems of bookkeeping complexity discussed by Taflove [5] and Yee [1]. An automatic three-dimensional (3-D) algorithm was proposed in [6]. Existing CFDTD algorithms are also late-time unstable, due to the nearest neighbor approximation used to compute some E-nodes. A stabilized two-dimensional (2-D) algorithm was presented in [7].

A contour/staircase finite-difference time-domain (FDTD) hybrid, which is simple, fully automatic, and stable is proposed. The algorithm is an improved version of [6], which used a cruder stabilization method. Both the current algorithm and [6] implement Faraday's integral law—a concept proposed by Taflove [5]. However, the algorithm presented here is automatic, 3-D, and stable. In the FDTD hybrid, the mesh is first generated using staircase preprocessing as in ordinary FDTD. Then, in an extended preprocessing stage, nodes using subcell formulas are flagged and required additional geometrical details are stored in auxiliary arrays. During time stepping, most nodes are treated as in ordinary FDTD and flagged nodes are updated using subcell formulas.

II. Mesh Generation

Let $In\_PEC(x, y, z)$ be a function that detects if the point (x, y, z) is physically within a perfect electric conductor. The computational domain is analyzed as a collection of Yee-cubes

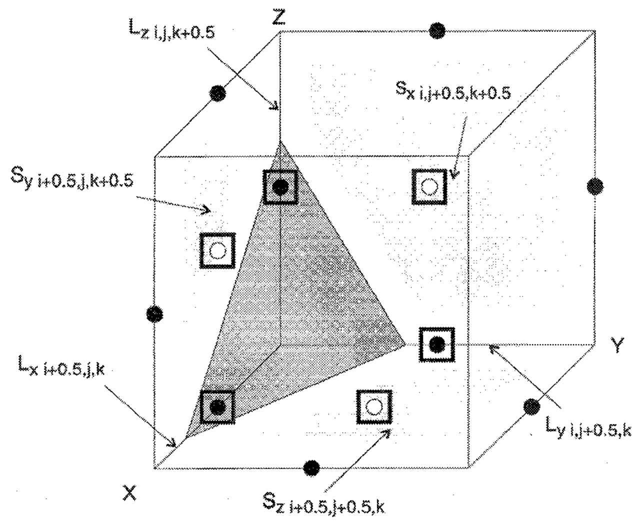

Fig. 1. Surface of a perfect electric conductor intersecting a Yee cube. Metal-free fractions of lines and surfaces are labeled. The filled and open circles represent E and H nodes, respectively. *Edge*-type nodes are boxed.

(Fig. 1), and E and H nodes are of either PEC- or *space*-type (the default). Staircase mesh generation is carried out by examining each cube in the computational domain [8]

IF ($In\_PEC(x, y, z) = true$ at cube center)
THEN flag all associated E and H nodes as PEC-type.

E and H nodes are then reexamined to flag *edge*-type nodes, which are updated using special formulas. A node is assigned *edge*-type under the following conditions

1) IF (an E-node is PEC-type) AND (one of the two neighboring collinear E-nodes is *space*-type) THEN the PEC-type node is flagged as *edge*-type.
2) IF (an H node is *space*-type) AND (at least one of the four neighboring coplanar H-nodes is PEC-type) THEN the *space*-type node is flagged as *edge*-type.

Additional coefficients used in the updates of *edge*-type nodes are also calculated and stored. From Fig. 1, it can be seen that E and H nodes are associated with line segments and cube faces, respectively. The metal-free fractions of these line segments and faces are stored in the arrays L_x, L_y, L_z, S_x, S_y, and S_z—associated with E_x, E_y, E_z, H_x, H_y, and H_z, respectively. All of the coefficients are easily computed by e.g., integrating in_PEC over the line or surface of interest.

Manuscript received September 25, 1995.
The authors are with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, B.C. V8W 2Y2, Canada.
Publisher Item Identifier S 1051-8207(96)02067-3.

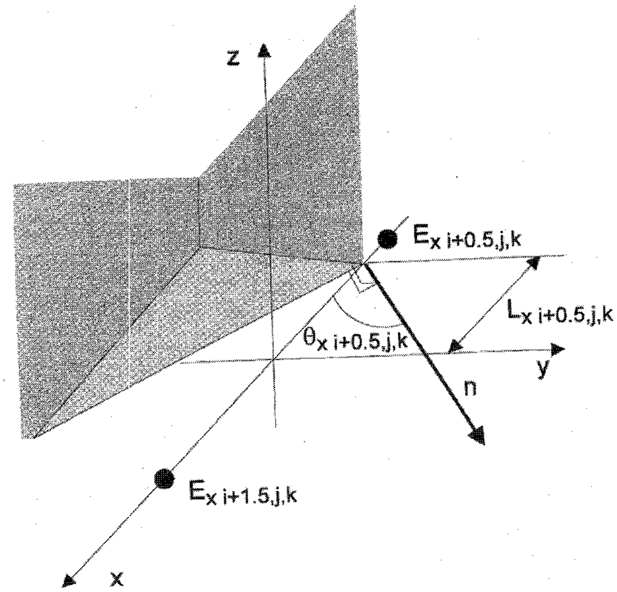

Fig. 2. Geometrical details used in the second-order nearest neighbor approximation.

III. Time Stepping

A. Updates of Edge-Type E Nodes

Late time instability in CFDTD is caused by the updating of certain E nodes using a nearest neighbor approximation. In a subset of nearest neighbor approximations where an auxiliary condition is not satisfied [8], the transfer of information between nodes at infinite speed violates causality. Therefore, in the hybrid algorithm, *edge*-type E nodes are updated with formulas such as $E_{xi+0.5,\,j,\,k} = kE_{xi+0.5\pm1,\,j,\,k}$ where the nodes on the left- and right-hand sides of the equation are *edge*- and *space*-type, respectively. The constant k is automatically zeroed for *edge*-type E nodes where the encircling H nodes are not all of PEC type. (In contour FDTD, k is always unity.)

Alternately, stable running times can be increased by a factor of three to four when the following second-order approximation, incorporating the electrostatic behavior of the field in the vicinity of metal [8], is used

$$k = \frac{L_{xi+0.5,\,j,\,k} + \cos^2(\theta_{xi+0.5,\,j,\,k})(1 + L_{xi+0.5,\,j,\,k})}{L_{xi+0.5,\,j,\,k} + (1 + L_{xi+0.5,\,j,\,k})} \quad (1)$$

where $L_{xi+0.5,\,j,\,k}$ is the metal-free fraction of the line segment of the *edge*-type node to be updated and $\theta_{xi+0.5,\,j,\,k}$ is the angle between the line segment and the normal to the PEC surface (Fig. 2).

B. Updates of Edge-Type H Nodes

An edge-type H node has one or more adjacent PEC-type H nodes on the same plane, due to the flagging procedure. A cluster consisting of the edge node under consideration and a neighboring PEC node is analyzed, using the integral form of Faraday's law as advocated by Taflove [2]. There are four possible two-cell clusters, each corresponding to a potential

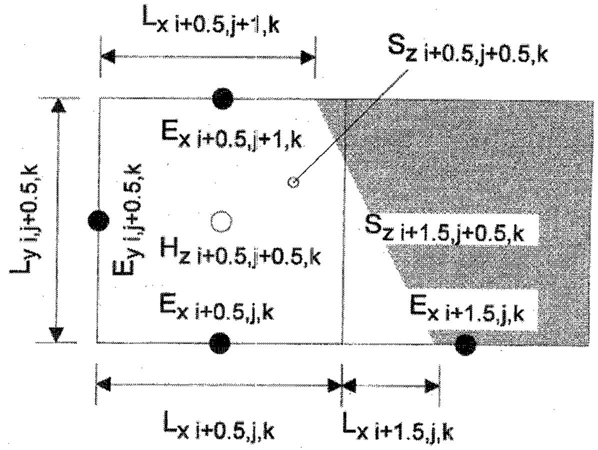

Fig. 3. Realization of Faraday's integral law with available FDTD nodes and contents of auxiliary geometry arrays. Only nonzero quantities are shown.

update formula. For example, updating an H_z edge node, with the cluster extended in the $+x$ direction (Fig. 3), leads to the following expression

$$\begin{aligned} H^{n+0.5}_{zi+0.5,\,j+0.5,\,k} = &\; H^{n-0.5}_{xi+0.5,\,j+0.5,\,k} - \frac{1}{S_{zi+0.5,\,j+0.5,\,k} + S_{zi+1.5,\,j+0.5,\,k}} \frac{\Delta t}{\mu} \\ &\times \left(\frac{L_{yi+2,\,j+0.5,\,k}E^n_{yi+2,\,j+0.5,\,k} - L_{yi,\,j+0.5,\,k}E^n_{yi,\,j+0.5,\,k}}{\Delta x} \right. \\ &- \frac{L_{xi+0.5,\,j+1,\,k}E^n_{xi+0.5,\,j+1,\,k} - L_{xi+0.5,\,j,\,k}E^n_{xi+0.5,\,j,\,k}}{\Delta y} \\ &\left. - \frac{L_{xi+1.5,\,j+1,\,k}E^n_{xi+1.5,\,j+1,\,k} - L_{xi+1.5,\,j,\,k}E^n_{xi+1.5,\,j,\,k}}{\Delta y} \right). \end{aligned} \quad (2)$$

Each E node is weighted by the metal free fraction (L) of its associated line segment, and the right-hand side is also weighted by the metal-free fractions (S) of the two Hnode cells in the cluster. Note that *edge* H nodes are always associated with clusters of exactly two cells, as opposed to other CFDTD formulations. When the *edge* node has multiple PEC-type neighbors, the cluster is formed using neighbor having the smallest fraction free of metal. In the rare circumstance when the extended integration contour of the cluster does not include at least one leg fully within metal, the algorithm instead performs a staircase FDTD update. This occurs mainly when the PEC surface is nearly tangent to the plane of nodes under consideration.

IV. Numerical Results

The contour/staircase hybrid was implemented by appending four additional subroutines to an existing staircase FDTD code. The generality and automatic mesh generation of the staircase FDTD code was preserved. Using a mesh with $\Delta x = \Delta y = \Delta z = 0.05$ m, resonant frequencies of rectangular, cylindrical, and spherical cavities of various sizes were computed using both staircase FDTD and the contour/staircase hybrid. The accuracy of the methods is compared in Tables

TABLE I
RESONANT FREQUENCY OF RECTANGULAR CAVITY

Side Length [m]	f_{res} [Mhz] Analytical	% Error (Staircase)	% Error (Hybrid)
1.00	212.0	-0.09	-0.09
1.02	207.8	+1.92	+0.28
1.04	203.8	+3.93	+0.53
1.06	200.0	-3.70	+0.00
1.08	196.3	-1.88	+0.10
1.10	192.7	-0.05	-0.05

TABLE II
RESONANT FREQUENCY OF CYLINDRICAL CAVITY

Diameter [m]	f_{res} [Mhz] Analytical	% Error (Staircase)	% Error (Hybrid)
1.00	229.5	+2.57	+0.57
1.02	225.0	+1.42	+0.13
1.04	220.7	+3.40	+0.68
1.06	216.5	+3.88	+0.51
1.08	212.5	+1.41	+0.38
1.10	208.7	+1.96	+0.53

TABLE III
RESONANT FREQUENCY OF SPHERICAL CAVITY

Diameter [m]	f_{res} [Mhz] Analytical	% Error (Staircase)	% Error (Hybrid)
1.00	261.9	+1.18	-0.34
1.02	256.7	+1.68	-0.30
1.04	251.8	+1.51	-0.24
1.06	247.0	+1.98	-0.24
1.08	242.4	+1.73	-0.17
1.10	238.0	+1.09	-0.13

I–III. The resonant frequencies of several higher order modes were also computed for all geometries. In all cases, similar gains in accuracy were observed. The additional CPU time required by the hybrid algorithm was less than 10%, since only a fraction of the nodes use the complicated update procedures. The coefficient k was set using the method yielding indefinite late time stability and was verified to 10 000 time steps with Δt at the Courant limit (1000 steps being required for an accurate result).

V. CONCLUSION

A hybrid contour/staircase FDTD algorithm was proposed. The algorithm incorporates existing staircase logic so as to eliminate the bookkeeping complexity of CFDTD in 3-D. Mesh generation remains fully automatic, and late instability can be drastically postponed or eliminated by selecting an appropriate procedure to update the *edge*-type E nodes. The algorithm was implemented by appending several subroutines to an existing staircase FDTD code and was validated by computing the resonant frequencies of rectangular, cylindrical, and spherical cavities.

REFERENCES

[1] K. Yee, "Ten years of evolution of the FDTD-like conformal techniques," in *Proc. 11th Ann. Rev. Progress in Applied Computational Electromagnetics,* Monterey, CA, 1995, pp. 1241–1247.

[2] T. G. Jurgens, A. Taflove, K. Umashankar, and T. G. Moore, "Finite-difference time domain modeling of curved surfaces," *IEEE Trans. Antennas Propagat.,* vol. 40, pp. 357–366, Apr. 1992.

[3] T. G. Jurgens and A. Taflove, "Three-dimensional contour FDTD modeling of scattering from single and multiple bodies," *IEEE Trans. Antennas Propagat.,* vol. 41, pp. 1703–1708, Dec. 1993.

[4] P. Mezzanotte, L. Roselli, and R. Sorrentino, "A simple way to model curved metal boundaries in FDTD algorithm avoiding staircase approximation," *IEEE Microwave Guided Wave Lett.*, vol. 5, pp. 267–269, Aug. 1995.

[5] A. Taflove, *Computational Electrodynamics: The Finite-Difference Time-Domain Method.* Boston: Artech House, 1995, pp. 299–300.

[6] J. Anderson, M. Okoniewski, and S. S. Stuchly, "A simple method for conformal modeling of metal surfaces in FDTD," submitted to *IEEE Trans. Antennas Propagat.,* May 1995.

[7] C. Railton, I. Craddock, and J. Schneider, "An improved locally distorted CPFDTD algorithm with provable stability," accepted for publication in *Electron. Lett.*

[8] J. Anderson, "Conformal treatment of perfect electric conductors in finite difference time domain method," M.A.Sc. thesis, Univ. of Victoria, Oct. 1995.