A STABILIZING SCHEME FOR THE EXPLICIT TIME-DOMAIN INTEGRAL-EQUATION ALGORITHM

S. Kashyap, M. Burton, and A. Louie
Electronics Countermeasures Section
Defence Research Establishment Ottawa
Ottawa, Canada

ABSTRACT: The stability of the explicit version of the solution of the time-marching electric-field integral-equation continues to depend on the specifics of the application. The design of stability-enhancing schemes seems to be more of an art than a science. A contribution to this art is made with a design that is stable where others are not, and the practical issues of implementation, accuracy, and efficiency in both time and memory are addressed.

1. INTRODUCTION

Use of the method of moments to solve the time-domain electric-field integral equation (EFIE) results in a matrix equation $\mathbf{Ax=b}$, where $\mathbf{x}$ is typically a vector of current densities [1]. The equation is repeatedly solved for $\mathbf{x}$ at progressing intervals of time. In the explicit form of the solution $\mathbf{A}$ is diagonal so its inversion is trivial. In the implicit form of the solution, $\mathbf{A}$ is sparse and, for three-dimensional problems, unbanded as well, so inversion is no longer trivial. In both forms, $\mathbf{A}$ is time-independent, which is a major attraction of solving the EFIE in the time domain. For a more detailed description and comparison of the two forms consult the introductory sections of [2] and [3], and Section 2.3 of [4]. Although each of the two forms of the solution has its merits, the implicit form is increasingly favored among researchers because of its late-time stability. We, however, have not yet abandoned the explicit form because we have found a way to stabilize it.

Most attempts to correct the stability problem have inserted so-called averaging schemes into the process [5,6,7]. None have been successful over a wide range of applications. Sadigh and Arvas [8] show that averaging is equivalent to using a finite impulse-response (FIR) digital filter that has a constant group delay. We have found that their particular implementation is also unstable for some applications. In this paper we present a variation of their instability filter and show it to be effective in all test cases considered. We have not yet found an application that causes late-time instability. Moreover, our stabilization scheme can be coded so that it adds a negligible amount to the overall execution time. To explain the issues affecting the design and behaviour of the new filter we briefly review how the moment method works and behaves in the time domain.

2. MOMENT-METHOD SOLUTION OF THE EFIE IN THE TIME DOMAIN

The basic EFIE states that the tangential component of the scattered field is equal to the negative of the tangential component of the incident field on the surface of a perfect conductor. That is,

$$\vec{E}_{scat} = -\vec{E}_{inc} \tag{1}$$

everywhere on the surface. This is an infinitely complex problem. The moment method reduces the problem to a manageable size by assuming that the current can be approximated by a finite number of "basis functions". The type of basis function used here is known as a "rooftop" function. All the algebraic and geometric properties of a rooftop, except the amplitude J, are known. See Figure 1. J is the "unknown" in the moment-method matrix equation. For more detailed information about rooftops see [1, 9, 10, 11].

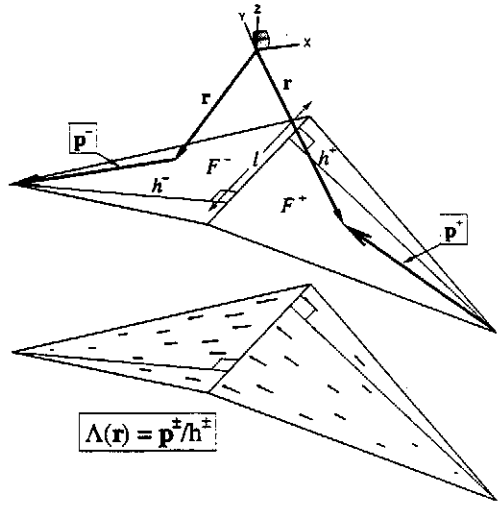

Figure 1 The "rooftop" type of basis function used in the time-domain algorithm.

The vector and scalar potentials $\vec{A}$ and V are typically used to find $\vec{E}_{scat}$ according to

$$\vec{E}_{scat} = -\frac{\partial \vec{A}}{\partial t} - \nabla V \tag{2}$$

where

$$\vec{A}(\vec{r}) = \frac{\mu}{4\pi}\int \frac{\vec{J}(\vec{r}')}{R} dS \tag{3}$$

and

$$V(\vec{r}) = \frac{1}{4\pi\epsilon}\int \frac{\rho(\vec{r}')}{R} dS \tag{4}$$

and $R = |\vec{r} - \vec{r}'|$. In the time domain, the computation of $\vec{E}_{scat}$ at $\vec{r}$ due to a given rooftop looks like

$$-\frac{\mu}{4\pi}\int \dot{J}(\tau)\frac{\vec{\Lambda}}{R} dS dt - \frac{1}{4\pi\epsilon}\nabla\int \frac{\rho(\tau)}{R} dS dt \tag{5}$$

where $\tau = t - R/c$ is called "retarded time". This involves integrals over both space and time. Such integrals have to be computed numerically, are compute-intensive, and present problems as $R \to 0$. To avoid this trouble Rao and Wilton [1] choose to assume that retarded time τ is constant over the entire source rooftop. The computation then becomes

$$\vec{E}_{scat} = -\frac{\mu}{4\pi}\dot{J}(\tau)\int \frac{\vec{\Lambda}}{R} dS - \frac{\rho(\tau)}{4\pi\epsilon}\nabla\int \frac{1}{R} dS \tag{6}$$

The integrals now have a purely spatial domain and can be efficiently computed from accurate analytical formulas [12]. Computing (6) requires a knowledge of <u>both</u> $J(\tau)$ and $\rho(\tau)$ for all t up to the present. But computing $\rho(\tau)$ would involve integrating the continuity equation, $\dot{\rho} = -\nabla\cdot\vec{J}$, at each time step and possibly storing the results. Rao and Wilton avoid these problems by solving (1) after it has been differentiated with respect to time. Then only $\dot{\rho}$ itself is needed, which for a rooftop is simply

$$\dot{\rho}^{\pm} = -2\frac{J}{\pm h^{\pm}} \tag{7}$$

The time derivative of (6) then becomes

$$\frac{\partial \vec{E}_{scat}}{\partial t} = -\frac{\mu}{4\pi}\ddot{J}\int \frac{\vec{\Lambda}}{R} dS + \frac{2J}{\pm h^{\pm} 4\pi\epsilon}\nabla\int_{F^{\pm}} \frac{1}{R} dS \tag{8}$$

Using a central-difference formula for numerically computing $\ddot{J}$ gives

$$\frac{\partial \vec{E}_{scat}}{\partial t} = -\frac{\mu}{4\pi}\left[\frac{J(t+\Delta t) - 2J(t) + J(t-\Delta t)}{(\Delta t)^2}\right]\int \frac{\vec{\Lambda}}{R} dS + \frac{2J(t)}{\pm h^{\pm} 4\pi\epsilon}\nabla\int_{F^{\pm}} \frac{1}{R} dS \tag{9}$$

where Δt is the time step.

To solve for N unknown amplitudes (1) must be enforced in (at least) N independent ways. In the language of the moment method this is called "testing". Rao and Wilton do their testing by computing path integrals of (1). Each rooftop has a testing path which begins at the centroid of face F^+, ends at the centroid of face F^-, and crosses the centre of the intervening edge. See Figure 1. Since each rooftop has a testing path that is different from all the others there are N independent ways to enforce (1). (This last statement becomes an interesting half-truth at very low frequencies. See [11].)

Consider testing (1) at the k^{th} rooftop at time t. In the basic time-domain algorithm the path integral is computed numerically in a very simple manner: its integrand is sampled only once, at the centre of the rooftop's interior edge, and τ is computed as if R was measured between edge centres. (Inside the integrals, R is still measured between $\vec{r}$ and $\vec{r}'$.) The result is

$$-\frac{\mu}{4\pi}\left[\frac{J(t+\Delta t) - 2J(t) + J(t-\Delta t)}{(\Delta t)^2}\right]\int_{k^{th}} \frac{\vec{\Lambda}}{R} dS + \frac{2J(t)}{\pm h^{\pm} 4\pi\epsilon}\nabla\int_{k^{th}} \frac{1}{R} dS + \int_{\text{all but } k^{th}} \frac{d\vec{E}_{scat}}{dt} dS = -\frac{d\vec{E}_{inc}}{dt} \tag{10}$$

$J(t+\Delta t)$ is the only unknown in this equation if all earlier rooftop amplitudes are known. Hence, the history of the response can be revealed by building on earlier solutions. (At the beginning of a response all earlier solutions are assumed to be zero.) The repeated solution of (10) ever further into the future is called "marching-on-in-time".

Note that, in our frequency-domain solutions of the EFIE, the path integrals are also computed numerically, but their integrands are sampled at the endpoints [9,10].

3. INSTABILITY FILTERS

The basic marching method is notorious for propagating errors into the future <u>and amplifying them</u>. The errors swamp the correct answer very early in the response. To

improve the stability, Rao and Wilton compute the path integrals differently according to the type of potential involved. When integrating the vector potential, $\vec{A}/R$, they sample the integrand at the point where the path crosses the rooftop's interior edge, and retarded time is held constant over the entire rooftop (as in the basic algorithm). When integrating the scalar potential, $1/R$, they sample the integrand at the endpoints of the path, and the retarded time is held constant over only faces. (The penalty for adding the Rao-Wilton approach to the basic algorithm is to increase the required memory. See Section 7 below.)

No one has claimed to know why the Rao-Wilton stability-enhancing scheme works. Also, it is by no means totally effective. There remains a wide range of problems that gives unstable results. To widen the set of stable problems we have added what can be called an "instability filter" to the Rao-Wilton algorithm. We provide details of our filter and demonstrate its effectiveness for a wide range of structures. We do not know why ours, or anyone else's stability-enhancing scheme, works. We can only say that ours has given stable results for all applications that we have tried. One of the reviewers notes that we give the impression that instability filters are not well understood and that an improvement has been achieved more or less by chance. We share this impression. In our opinion, the design of instability filters remains an art, not a science.

Figure 2 shows a comparison of two filtering schemes: one designed by Sadigh and Arvas [8], labeled 5SA; and one designed by us [13], labeled 5SK. We grafted each scheme to a marching-on-in-time code, whose design originates with Rao and Wilton [1]. The latter has no stability filter and is labeled 0RW. The two diagrams, 5SA and 5SK, show the sequence of computations on the way to getting a single filtered step into the future for the current crossing a given edge. The letters represent the signed magnitude of that current. Time increases from left to right and the sequence of computations progresses from top to bottom.

The two filtering schemes, 5SA and 5SK, are alike: a march of three steps into the future is needed to compute a single filtered step. The two extra steps supply the necessary data for the 5-term filters to produce a "corrected" delay-compensated output for the first step. The difference between the two schemes is simple: the 5SA scheme does a filtering operation only after the third step; the 5SK scheme does a filtering operation after each of the three steps, and adds an extra filtering operation afterwards. Both schemes use the same filter coefficients: 0.0804, 0.25, 0.3392, 0.25, 0.0804. In the results that follow, output from each of the three different codes is labeled 0RW, 5SA, and 5SK, as appropriate.

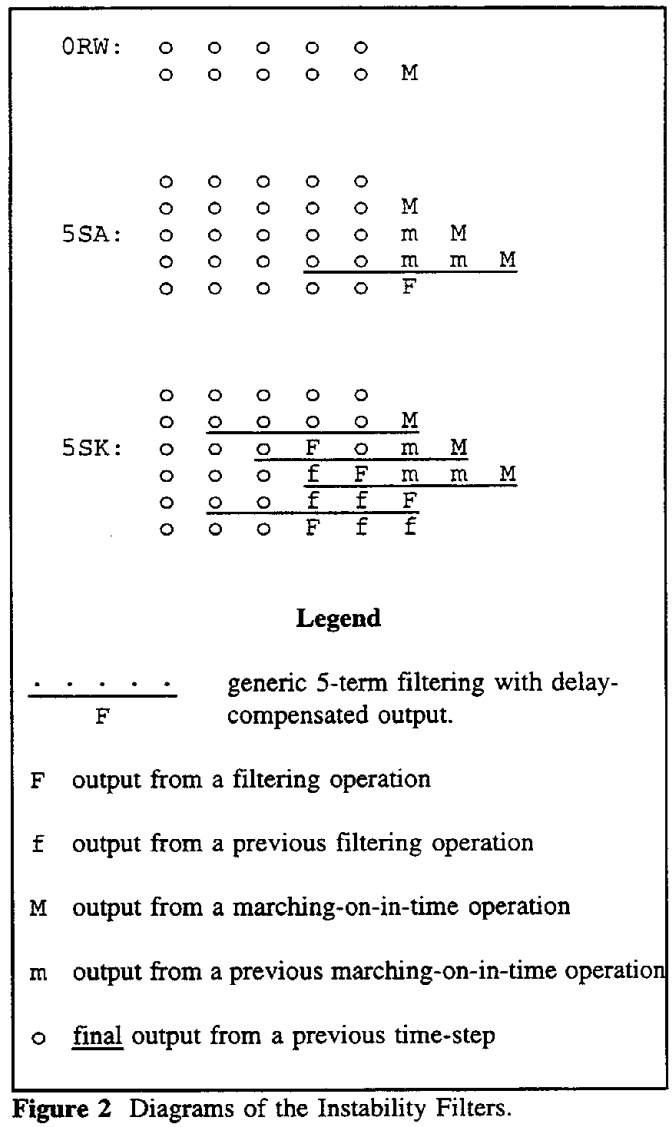

Figure 2 Diagrams of the Instability Filters.

4. RESULTS

Figures 3-5 show the results of our computation for three structures. Figure 3 shows the time-domain response to a Gaussian excitation for a "tray". The tray is divided into 76 triangular faces. It shows that for this structure the Sadigh and Arvas scheme is in fact worse than no filtering at all, and that our filter gives a stable result. Similarly Figures 4 and 5 demonstrate the improved stability for a solid cube and for a trihedral corner reflector, respectively. The cube has 260 triangular faces; the trihedral corner reflector has 768 faces.

Figures 6 to 9 demonstrate the stability of our scheme for a square plate, an open cube, a sphere, and a torus. The square plate has 180 faces, the open cube has 220 faces, the

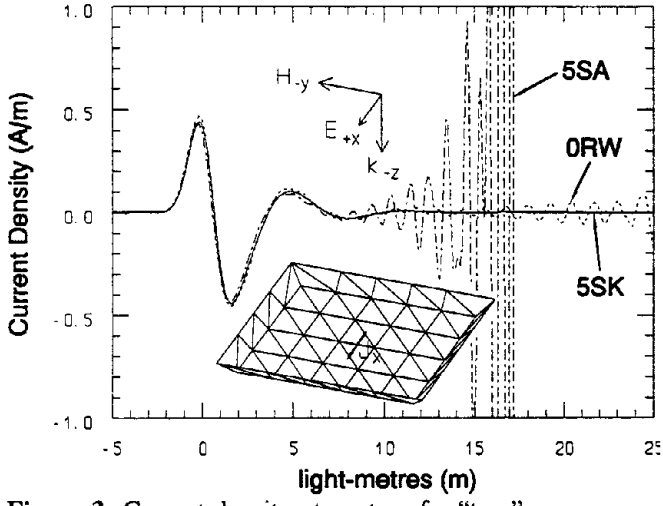

Figure 3 Current density at centre of a "tray".

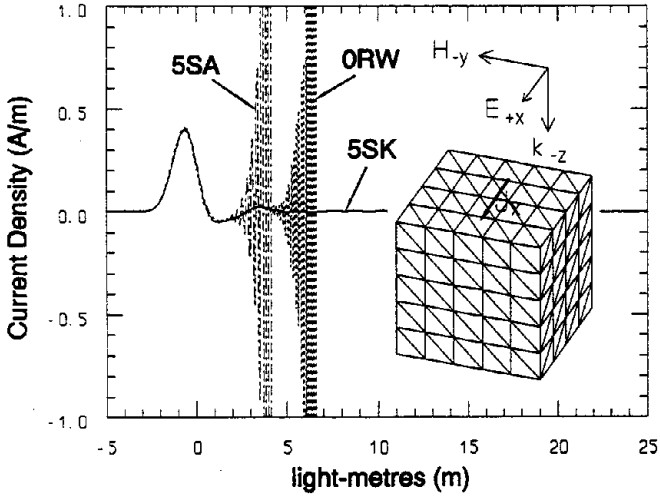

Figure 4 Current density at top centre of a cube.

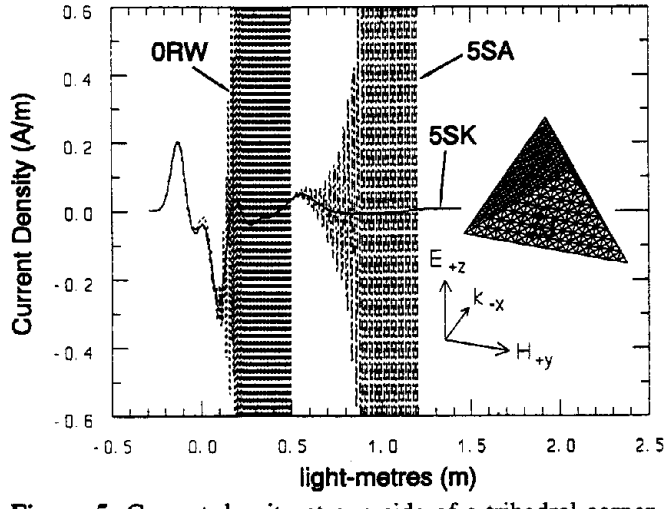

Figure 5 Current density at one side of a trihedral corner.

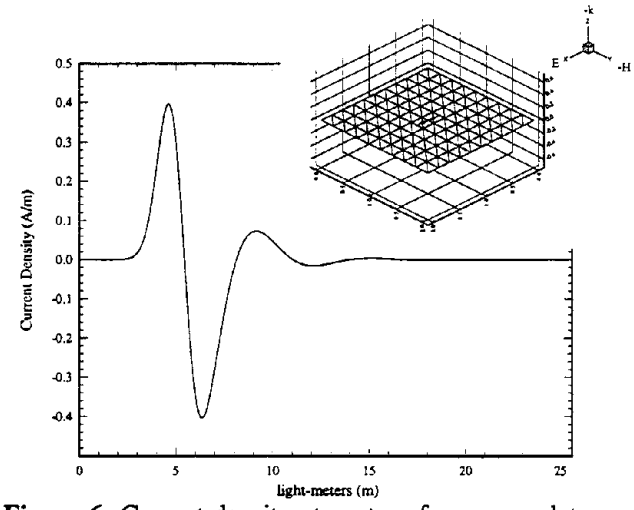

Figure 6 Current density at centre of a square plate.

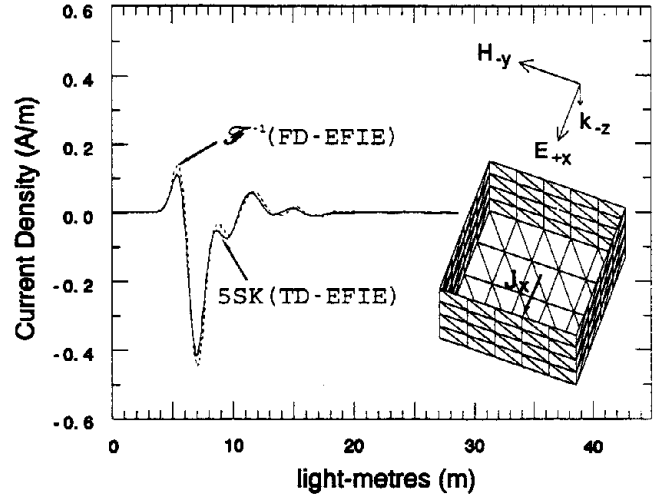

Figure 7 Current density at centre of the bottom of an open cube.

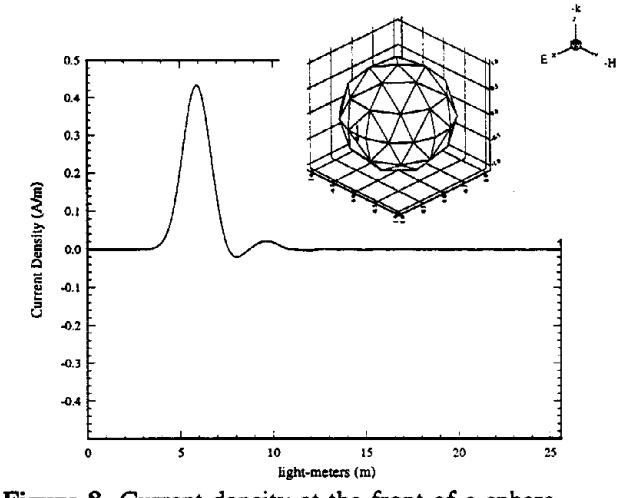

Figure 8 Current density at the front of a sphere.

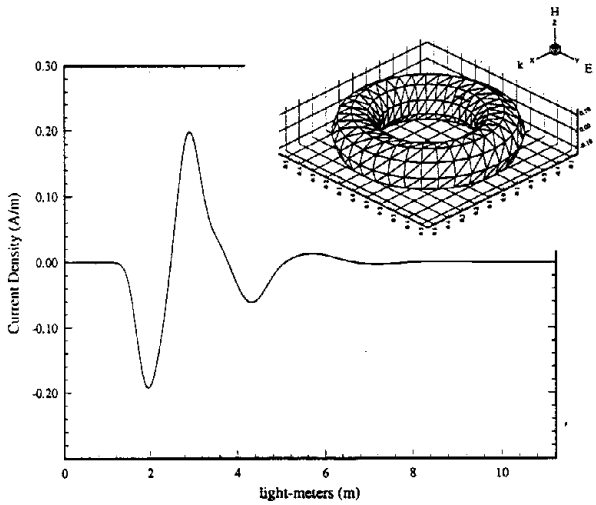

Figure 9 Current density on the wall of a torus.

sphere has 90 faces, and the torus has 630 faces. Figure 7 also shows a comparison with the results obtained by inverse Fourier transformation of the frequency domain EFIE results.

5. REDUCTION OF FILTERING TIME

Both the filtering schemes, 5SA and 5SK, require three time-marching iterations to obtain a single, new, vector of filtered currents. When either is coded simply, this will lead to a tripling of the CPU time compared to the unfiltered, 0RW, scheme. Taking the 5SK scheme as an example, this simple encoding can be represented by the "index space" diagram in Figure 10. The time index varies horizontally; the 5SK

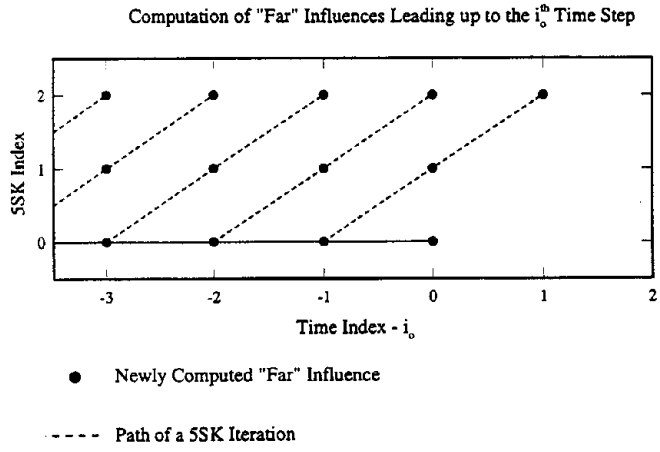

Figure 10 Repetitive computation of far influences in a 5SK stabilization scheme.

index varies vertically. At any given time index there are three concurrent 5SK loops travelling diagonally through index space.

Almost all of the CPU-time overhead can be recovered with careful coding to take advantage of the following subtle detail: the 5SK filtering scheme alters only recent entries in the history book. Through retarded time, this effectively alters the scattered field from nearby rooftops — the scattered field from far rooftops is not changed. So only the near influences need to be refreshed within each of the three concurrent loops in index space. The far influence needs to be computed only once, since it can be reused in the other two loops. Figure 11 shows how this looks in 5SK index

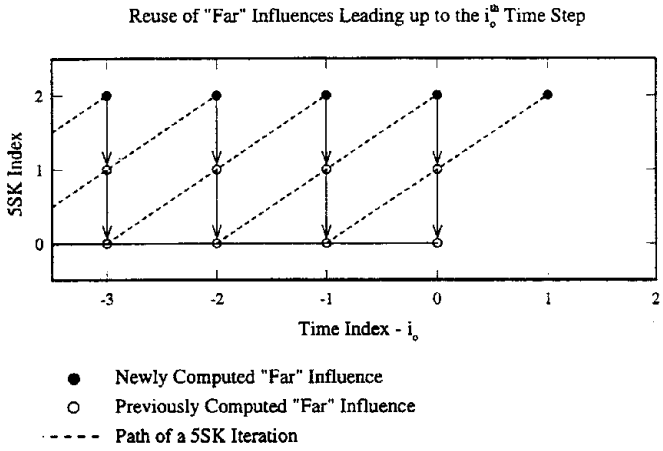

Figure 11 Re-use of previously computed far influences in a 5SK stabilization.

space. The far influence computed at a given time index is used at the same time index in the other two 5SK loops. Note that each iteration of the instability filter uses two different sets of remembered far-influences. This scheme results in a recovery of almost all of the CPU-time overhead attributed to filtering.

6. EXECUTION TIME

A common application of computational electromagnetics is to find the response due to narrow-pulse (wide-spectrum) illumination from a fixed source. Under these conditions we can make a meaningful comparison between the execution times of time-domain and frequency-domain EFIE codes since, for real signals, the number of data points on the time-domain side of a fast Fourier transform is twice the number on the frequency-domain side. At each time step, a marching-on-in-time algorithm needs of the order of N_e^2 computations to solve for all N_e unknown current functions on the body. (N_e is the number of interior edges in the tessellation of the surface.) At each frequency step, a frequency-domain code needs of the order of N_e^3 computations to solve for all N_e unknown currents at each frequency step. So, as the size of the problem increases, it

should be faster to use a time-domain code, by a factor of the order of N_e. For these reasons we asked for, and received free of charge, a copy of the code that had been written to prove the Rao-Wilton algorithm [1].

The code used by Rao and Wilton used almost no memory — it recomputed everything it needed at every step in the march. This was an extravagant use of CPU time for all but the smallest of problems. We extensively modified the code to reduce the execution time, but the core-memory requirements became excessive — about four times what was needed by the frequency-domain code [10] that we use. Also, the promise of quickly obtaining narrow-pulse responses failed — the CPU time was about the same as that needed to do an equivalent spectral sweep with our frequency domain code. It turned out that this disappointing performance was due not to the algorithm but to fundamental inefficiencies built into the original code, which was written solely to prove the Rao-Wilton algorithm on small problems. The code was not designed for efficient use of computer resources when solving large practical problems, and our modifications could not hide its academic heritage. This was our motivation to write a completely new time-domain EFIE code. It uses none of the code from Rao and Wilton but faithfully implements every feature of their algorithm. The result is labeled 0RW in this report. (As noted in Section 5 above, the execution times of the our 0RW and 5SK codes are practically the same.)

Figure 12 compares the execution times of our frequency-domain and (5SK) time-domain codes. The frequency domain code is always slower by at least an order of magnitude. For $N_e > 2000$, it is slower by a factor of about $N_e/100$. The speed advantage of the time-domain solution is obvious for the case of narrow-pulse (wide-band) illumination. However, the speed advantage would not be so obvious for the opposite case of continuous-wave (narrow-band) illumination. Then, the time-domain solution must run until all transients have vanished. This may actually take longer than the frequency-domain solution, especially if the illumination is near resonance. It can be argued that this is an unfair comparison since the frequency-domain solution gives only the late-time result; it does not tell what happened on the way.

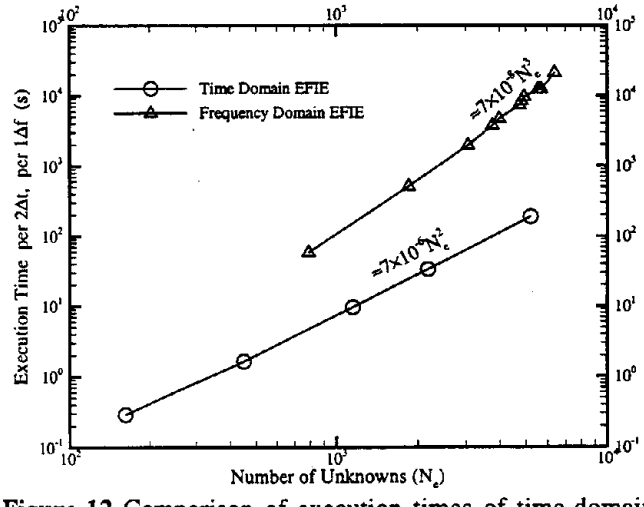

Figure 12 Comparison of execution times of time-domain and frequency-domain EFIE codes for the case of narrow-pulse illumination from a fixed source.

Note that, for both time-domain and frequency-domain solutions of the EFIE, the tessellation restricts the highest frequency in the spectrum of the illumination. For frequency-domain solutions, the commonly accepted restriction is $f_{max} < c/10\Delta R_{min}$, where c is the speed of light and ΔR_{min} is the smallest distance between the centroids of any two faces of the tessellation. For the explicit form of time-domain solutions there is the Courant condition $c\Delta t > \Delta R_{min}$. Another restriction is $f_{max} < 1/10\Delta t$, so that the central-difference formula for time derivatives will be accurate enough for the given illumination. Combining this with the Courant condition gives the same restriction as for frequency-domain solutions of the EFIE.

7. MEMORY REQUIREMENTS

The memory that we discuss here is core memory, not the archival memory that stores, for later off-line analysis, the computed rooftop amplitudes at each time step or at each frequency step. The size of core memory stays fixed as the program runs; archival memory grows. For large problems, the frequency-domain code's core memory is almost entirely taken up by the impedance matrix; which is full, square, and complex. The number of unknown rooftop amplitudes is nearly equal to the number of interior edges N_e in the tessellated surface of the body. The number of single-precision real words of memory in the impedance matrix is $2N_e^2$. (Two real words are needed to represent a complex number.) Also, for a triangular tessellation, the number of faces N_f and the number of edges are related by $N_e \approx 1.5N_f$. So, for the frequency-domain (FD) code:

$$\text{FD memory} \approx 4.5N_f^2$$

For large problems the basic time-domain code's memory requirement is exactly the same. N_e^2 words are needed for the mutual influences between N_e rooftops and another N_e^2 for the time delays at which these influences occur. But when the Rao-Wilton stability enhancing scheme is added, the mutual influences due to $\bar{\Lambda}/R$ and $1/R$ must be kept in separate arrays because their different delays demand that they be multiplied by different rooftop amplitudes from the history book. As with the frequency-domain code, about

$4.5N_f^2$ words are needed for describing the mutual $\vec{\Lambda}/R$ influences (and their delays); but an extra $2N_f^2$ words are needed for describing the $1/R$ influences (and their delays). So, for the time-domain (TD) codes 0RW and 5SK:

$$\text{TD memory} \approx 6.5N_f^2$$

None of this extra memory would be needed if the marching algorithm were stable. Note that the 0RW and 5SK codes use nearly the same amount of memory even though the 5SK code has an instability filter added.

8. ACCURACY

Comparison of the two curves in Figure 7 shows that the results of the time-domain code differ from the results of the frequency-domain code. Our experience shows that the time-domain codes (both our 0RW and 5SK codes, and Rao's and Wilton's original code too) are not as accurate as our frequency domain EFIE code and that the error increases with frequency. The accuracy improves with finer triangulation of the surface, but this requires more computation time. Thus our statement that the time-domain code is $N_e/100$ times faster than the frequency domain code will be too optimistic if the same accuracy is desired.

The probable explanation for the poor high frequency accuracy is that mutual influences are computed by assuming a constant retarded time over faces and rooftops. The corresponding process in the frequency domain would be to assume a constant retarded phase $-jkR$. But this is not done — the frequency-domain code actually performs a numerical integration to account for the effect of the variation of the retarded phase over each face. This difference between the two codes would be most pronounced when the retarded time, or the retarded phase, was most variable, that is, at high frequencies. Another contribution to the high frequency errors is the low-pass filtering effect of the instability filter.

9. DISCUSSION

An exponentially growing oscillation at the input of a digital filter will produce an exponentially growing output even if the oscillation is out of band. In other words, putting off the filtering until the march is over will not remove the unstable signal. The marching algorithm must operate on filtered data as it proceeds, to keep the unstable part from feeding on itself. Both filtering schemes do this, but the 5SK scheme is better at controlling the unstable signal. We think this is because the 5SK scheme operates on more filtered data.

A comparison of the early-time results of the three marching methods shows how little (or how much) the true signal is corrupted by the filtering schemes. Some of the smaller local extrema can change by as much as thirty percent. The 5SA and 5SK schemes agree more with each other than with the 0RW scheme.

We do not believe that we have eliminated the stability problem, but that we have extended the applicability of the time-domain EFIE solution to a larger class of problems. There are probably combinations of illumination, tessellation, geometry, and Δt that will lead to unstable solutions.

10. CONCLUSIONS

We have presented a new filtering scheme for improving the stability of explicit, time-marching, solutions of the EFIE. The limited number of cases presented here demonstrate that the scheme is stable for a wider range of problems than previous schemes. We have shown a way to practically eliminate the CPU-time overhead caused by such filtering schemes. We have also shown that the time domain code is less accurate than the frequency-domain code, and takes 50% more memory. However, for large number of unknowns N_e, it is $N_e/100$ times faster than the frequency-domain code for the case of narrow-pulse illumination from a fixed source.

11. REFERENCES

[1] S. M. Rao and D. R. Wilton, "Transient scattering by conducting surfaces of arbitrary shape," ***IEEE Trans. Antennas and Propagation***, vol.39, no.1, pp. 56-61, Jan. 1991.

[2] S.M. Rao and T.K. Sarkar, "Transient analysis of electromagnetic scattering from wire structures utilizing an implicit time-domain integral-equation technique," ***Microwave And Optical Technology Letters***, vol. 17, no. 1, Jan. 1998, pp. 66-69.

[3] S.M. Rao and T.K. Sarkar, "Time-domain modeling of two-dimensional conducting cylinders utilizing an implicit scheme — TM incidence," ***Microwave And Optical Technology Letters***, vol. 15, no. 6, Aug. 20, 1997, pp. 342-347.

[4] S.P. Walker, "Developments in time-domain integral-equation modeling at Imperial College," ***IEEE Antennas and Propagation Magazine***, vol. 39, no. 1, Feb. 1997, pp.7-19.

[5] A. G. Tijhuis, "Toward a stable marching-on-in-time method for two-dimensional electromagnetic scat-

tering problems," ***Radio Sci.***, vol. 19, no. 5, pp.1311-1317, Sep.-Oct. 1984.

[6] B. P. Rynne, "Instabilities in time marching methods for scattering problems," ***Electromagnetics***, vol. 6, pp.129-144, 1986.

[7] P. D. Smith, "Instabilities in time marching methods for scattering: cause and rectification," ***Electromagnetics***, vol. 10, pp.439-451, 1990.

[8] A. Sadigh and E. Arvas, "Treating the instabilities in marching-on-in-time method from a different perspective," ***IEEE Trans. Antennas and Propagation***, vol.41, no.12, pp. 1695-1702, Dec. 1993.

[9] S. M. Rao, D. R. Wilton, and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," ***IEEE Trans. Antennas Propagation***, vol. AP-30, no. 3, May 1982, pp. 409-418.

[10] S. U. Hwu and D. R. Wilton, ***JUNCTION CODE USER'S MANUAL, Electromagnetic Scattering and Radiation by Arbitrary Configurations of Conducting Bodies and Wires***, Technical Report Number 87-18, Applied Electromagnetics Laboratory, Department of Electrical Engineering, University of Houston, 1989.

[11] M. Burton and S. Kashyap, "A study of a new moment-method algorithm that is accurate to very low frequencies," ***Applied Computational Electromagnetics Society Journal***, Vol. 10, No. 3, 1995, pp. 58-68.

[12] D. R. Wilton, S. M. Rao, A. W. Glisson, D. H. Schaubert, O. M. Al-Bundak, and C. M. Butler, "Potential integrals for uniform and linear source distributions on polygonal and polyhedral domains," ***IEEE Trans. Antennas Propagation***., vol. AP-32, no. 3, March 1984, pp. 276-281.

[13] S. Kashyap, M. Burton, and A. Louie, "Stabilizing the time-marching EFIE algorithm," ***IEEE Antennas and Propagation Society International Symposium 1995***, vol. 2, pp. 1033-1036.