Aalborg Universitet

AALBORG UNIVERSITY

Efficient Calculation of Near Fields in the FDTD Method

Franek, Ondrej

Efficient Calculation of Near Fields in the FDTD Method

O. Franek[1]

Abstract – **When calculating frequency-domain near fields by the FDTD method, almost 50 % reduction in memory and CPU operations can be achieved if only E-fields are stored during the main time-stepping loop and H-fields computed later. An improved method of obtaining the H-fields from Faraday's Law is presented and shown to be orders of magnitude more accurate than the straightforward approach, at no additional cost. The method is validated by comparison with analytical solution of a traveling wave, and on simulations of dipole and patch antennas.**

1   INTRODUCTION

The finite-difference time-domain (FDTD) method [1] is widely used in electromagnetics thanks to its property of encompassing wide range of frequencies in one simulation run, via pulse excitation. Desired quantities in frequency domain are then computed from time domain response using fast Fourier transform (FFT). If we demand the complete electromagnetic field distribution in the whole computational domain (e.g. near fields around an antenna, inside human body, etc.), storing time domain responses for every field component in the FDTD grid would lead to an excessive amount of memory needed. Therefore, the field values at selected frequencies are rather calculated on-the-fly using discrete Fourier transform (DFT) algorithm [2].

Still, the requirements on memory during the simulation are quite high: apart from the essential 6 three-dimensional *real* arrays for each of the field components (E_x, E_y, E_z, H_x, H_y, H_z), another 6 arrays of *complex* numbers for the fields need to be allocated, *times number of frequencies*. In the end, the total memory requirements will be several times higher than for the basic FDTD algorithm. Similarly, the DFT operations on the field arrays will substantially add to the total CPU operations in each time step, and considerably extend the simulation time.

In this paper we suggest to alleviate the memory and CPU requirements in such a scenario by storing and applying DFT to electric field only – the magnetic field is then cheaply calculated after the FDTD loop is finished. Here we note, however, that using a straightforward approach based on Faraday's Law is accompanied by a non-negligible error. Instead, we propose a modified formula based on the temporal discretization of the FDTD method which gives superior results at no additional cost, and, on certain conditions, it produces field values *exactly* the

same as those obtained by applying DFT on H-field directly.

2   THEORY

Let us define the DFT (denoted by a hat) of the time-domain electric field E as

$$\hat{E}(f) = \int_{-\infty}^{+\infty} E(t)\mathrm{e}^{-\mathrm{j}\omega t}\mathrm{d}t \approx \Delta t \sum_{n=0}^{N-1} E(n\Delta t)\mathrm{e}^{-\mathrm{j}\omega n\Delta t} \quad (1)$$

As the magnetic field H is shifted by half of the FDTD time step Δt, its DFT will accordingly be

$$\hat{H}(f) = \int_{-\infty}^{+\infty} H(t)\mathrm{e}^{-\mathrm{j}\omega t}\mathrm{d}t \approx$$

$$\approx \Delta t \sum_{n=0}^{N-1} H[(n+0.5)\Delta t]\mathrm{e}^{-\mathrm{j}\omega(n+0.5)\Delta t} \quad (2)$$

Here, $\omega = 2\pi f$ where f is the desired frequency, N is the length of the time-domain sequence, and $\mathrm{j}=\sqrt{-1}$. Furthermore, bold typeface means that the symbol represents a vector anywhere in the computational domain.

The E- and H-fields are related by Faraday's Law which in time domain is given by

$$-\nabla_\mathrm{G} \times E = M_i + \mu\frac{\partial H}{\partial t} \quad (3)$$

where M_i is the impressed magnetic current density, μ is the permeability and $\nabla_\mathrm{G} \times$ denotes the discrete Yee grid curl operator. In frequency domain, the H-field can be obtained from the E-field by

$$\hat{H} = \frac{\mathrm{j}}{\omega\mu}(\nabla_\mathrm{G} \times \hat{E} + \hat{M}_i) \quad (4)$$

whereas the time derivative has been replaced by $\mathrm{j}\omega$. This way we can save half of the memory resources dedicated to the frequency domain data by storing only the frequency domain E-fields during the FDTD run and calculating the H-fields once the main loop is finished. The only other data needed is the array of μ, which is usually given by a look-up table as long as there is limited number of materials in the computational domain, and the impressed sources M_i at the particular frequency.

However, calculating the H-fields by (4) is burdened by a non-negligible error owing to the fact that we did not take into account the time discretization of the FDTD method. Rewriting

---

[1] APNet Section, Department of Electronic Systems, Faculty of Engineering and Science, Aalborg University, Niels Jernes Vej 12, DK-9220 Aalborg Ø, Denmark, e-mail: of@es.aau.dk, tel.: +45 9940 9837, fax: +45 9815 1583.

Faraday's Law (3) for the n-th time step in the FDTD method we obtain

$$-\nabla_G \times \boldsymbol{E}(n\Delta t) = \boldsymbol{M}_i(n\Delta t) +$$
$$+ \boldsymbol{\mu} \frac{\boldsymbol{H}((n+0.5)\Delta t) - \boldsymbol{H}((n-0.5)\Delta t)}{\Delta t} \qquad (5)$$

Next, we apply the DFT as defined by (1) to both sides of (5). Let us assume that the excitation of the FDTD simulation is done by a time limited impulse for which $\boldsymbol{H}(-0.5\Delta t) = 0$ and the values at the point of terminating the simulation are decayed practically to zero, $\boldsymbol{H}[(N-0.5)\Delta t] \approx 0$. The frequency domain H-field following definition (2) can then be expressed as

$$\hat{\boldsymbol{H}} = \frac{j\Delta t}{2\boldsymbol{\mu}\,\sin(\omega\Delta t/2)}(\nabla_G \times \hat{\boldsymbol{E}} + \hat{\boldsymbol{M}}_i) \qquad (6)$$

This formula brings significant improvement in accuracy in comparison with (4), as will be shown in section 4, and at no additional cost.

The ratio between the original formula (4) and the corrected formula (6) is given by

$$\frac{\hat{\boldsymbol{H}}_{(4)}}{\hat{\boldsymbol{H}}_{(6)}} = \operatorname{sinc}(\omega\Delta t/2) = \operatorname{sinc}(\pi S/N_\lambda) \qquad (7)$$

it is real (no phase shift) and dependent on frequency and time step. Alternatively, it is proportional to the Courant number S [3] and inversely proportional to the number of cells per free-space wavelength N_λ. This means that in the limit of infinite grid resolution both formulations are identical, but for practical resolutions the modified formula (6) has usually an advantage of several orders of magnitude lower error.

3 RESOURCES

The proposed method reduces the amount of stored frequency domain near fields to one half by completely eliminating the need to store the DFT of the H-field. Correspondingly, the CPU operations related to these fields are also eliminated.

3.1 Memory

During an FDTD simulation, six field components (E_x, E_y, E_z, H_x, H_y, H_z) per one cell are stored. In most cases the number of materials in the computational domain is limited and one additional value per cell is then used as a pointer into a material look-up table. To obtain field distribution at a particular set of frequencies, via the on-the-fly DFT [2], we again need to store six field components, but as a complex number (2 values) and for each frequency. Total memory for this scenario is then proportional to

$$M_{\text{orig}} \sim 6 + 1 + 6 \times 2 \times N_f \qquad (8)$$

where N_f is the number of frequencies.

If we entirely omit the frequency domain H-fields during the simulation, the memory requirements decrease to

$$M_{\text{reduced}} \sim 6 + 1 + 3 \times 2 \times N_f \qquad (9)$$

For one frequency, this results in approx. 32 % reduction of memory, but for multiple frequencies the savings approach 50 %, as shown in Fig. 1.

3.2 CPU

The number of multiplications performed by a CPU running the basic FDTD algorithm is 9 per cell for electrically lossy materials [1]. The DFT adds another two multiplications per field component (complex) per frequency, resulting in

$$C_{\text{orig}} \sim 9 + 6 \times 2 \times N_f \qquad (10)$$

The proposed method reduces the number of multiplications connected with the DFT to half

$$C_{\text{proposed}} \sim 9 + 3 \times 2 \times N_f \qquad (11)$$

Again, for one frequency the benefit is around 29 %, but if many frequencies are required the reduction in multiplication CPU operations converges to 50 % (see Fig. 1).
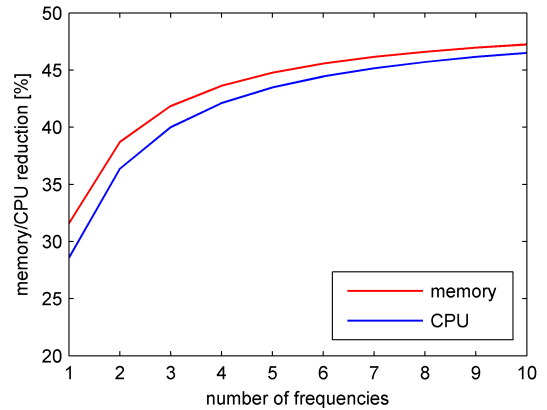

Figure 1: Reduction in required memory and CPU multiplication operations with respect to full six-component DFT.

4 VALIDATION

The differences between the H-field computation approaches are demonstrated on an example of a traveling plane wave for which analytical solution is available, and also on a full-wave dipole and a patch antenna.

4.1 Traveling plane wave

The field is invariant along the x and y axes, whereas the wave is traveling in the z direction. The cell size is 10 mm and the time step is given by the Courant factor $S = 1/\sqrt{3}$. Three methods of obtaining

the H-field are compared with the analytical solution: the original method of full six-component DFT computation by (2) (with no reduction in resources); from Faraday's Law by (4); and by the proposed method (6). Magnitudes of the H-field with corresponding percentage errors are presented in Tables 1 and 2, at frequencies 3 GHz and 6 GHz, respectively.

Method	H [mA/m]	error [%]
Analytical	2.65	–
Eq. (2)	2.65	-1.04×10^{-5}
Eq. (4)	2.64	-0.54
Eq. (6)	2.65	-3.39×10^{-6}

Table 1: H-field in a plane wave and error with respect to the analytical solution, $\Delta z = 10$ mm, $f = 3$ GHz.

Method	H [mA/m]	error [%]
Analytical	2.65	–
Eq. (2)	2.65	6.79×10^{-3}
Eq. (4)	2.60	-2.17
Eq. (6)	2.65	7.31×10^{-3}

Table 2: H-field in a plane wave and error with respect to the analytical solution, $\Delta z = 10$ mm, $f = 6$ GHz.

In both cases, the original and the proposed methods give very small error, not exceeding 0.01 % even when the mesh is very coarse and there are only 5 cells per wavelength (Table 2). On the other hand, the uncorrected formula (4) can produce error of more than 2 %, which is generally growing with coarser mesh, as suggested by (7).

It should be noted that the results from Eq. (2) and (6) are different only due to the term $\boldsymbol{H}\left[(N-0.5)\Delta t\right]$ which in real simulations is very small although not entirely zero as previously assumed. However, if this term is taken into account, the two results can be made identical. Fortunately, doing so does not pose any serious difficulties, since the term is actually the last value of the time-marching FDTD sequence and is present in memory at the end of the simulation, right when the H-field is about to be evaluated.

4.2 Dipole antenna

The dipole is 270 mm long and represents a full-wave dipole at 900 MHz in a very coarse resolution – the dipole spans only 9 FDTD cells of 30 mm. Fig. 2 displays the H-field along the dipole in the distance of 15 mm (half a cell). Despite the coarse resolution there is an excellent agreement between the proposed efficient method and the full six-component method with no reduction in memory or CPU usage. The dynamic range of the simulation (ratio of the peak field value to the maximum difference) is 127 dB.
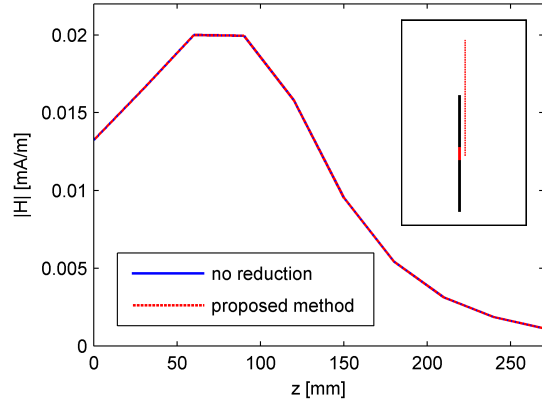

Figure 2: Magnetic field H along the dipole antenna at 900 MHz for the proposed method (6) and for (2) with no reduction of fields.

4.3 Patch antenna

The patch antenna has dimensions 15.7×15.7 mm with a 3.14 mm wide feed line. The substrate has dimensions 31.4×31.4 mm, it is 1.57 mm thick with relative permittivity 2, and it is backed by a metallic ground plane. The FDTD cell size is 1.57 mm, which gives approx. $N_\lambda = 33$ at frequency 5.8 GHz.

Fig. 3 shows the magnitude of the H-field along a line from the center of the patch towards the side, as depicted by the dotted red line in the inset of Fig. 3. The field obtained by (6) agrees again very well with the original method, with dynamic range 88 dB.
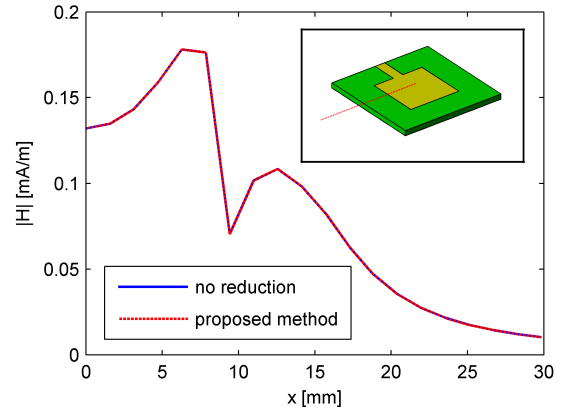

Figure 3: Magnetic field H over the surface of the patch antenna at 5.8 GHz for the proposed method (6) and for (2) with no reduction of fields.

5   CONCLUSION

It has been demonstrated that omitting one set of field components, in our case the H-fields, when computing the near fields via DFT can bring considerable savings in memory and CPU resources

of the FDTD method, reaching nearly 50 % for multiple frequencies. It has also been found that the H-fields subsequently obtained from the derived formula are of the same accuracy as with the direct approach. The proposed method therefore allows simulations of larger structures with the same amount of computer memory, less CPU resources, and, at the same time, no compromise in quality of results.

Acknowledgment

References

[1] A. Taflove, and S. C. Hagness, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*, Artech House, 3rd ed., 2005.

[2] C. M. Furse, and O. P. Gandhi, "Why the DFT is faster than the FFT for FDTD time-to-frequency domain conversions," In *IEEE Microwave and Guided Letters*, vol. 5, no. 10, Oct. 1995.

[3] J. B. Schneider, and C. L. Wagner, "FDTD Dispersion Revisited: Faster-Than-Light Propagation," In *IEEE Microwave and Guided Letters*, vol. 9, no. 2, Feb. 1999.