01 Aug 2005

A Three-Dimensional FDTD Subgridding Method with Separate Spatial and Temporal Subgridding Interfaces

Kai Xiao

David Pommerenke
Missouri University of Science and Technology, davidjp@mst.edu

James L. Drewniak
Missouri University of Science and Technology, drewniak@mst.edu

Recommended Citation

K. Xiao et al., "A Three-Dimensional FDTD Subgridding Method with Separate Spatial and Temporal Subgridding Interfaces," *Proceedings of the IEEE International Symposium on Electromagnetic Compatibility (2005, Chicago, IL)*, vol. 2, pp. 578-583, Institute of Electrical and Electronics Engineers (IEEE), Aug 2005.
The definitive version is available at https://doi.org/10.1109/ISEMC.2005.1513581

A three-dimensional FDTD subgridding method with separate spatial and temporal subgridding interfaces

Kai Xiao, David J. Pommerenke, and James L. Drewniak
Electromagnetic Compatibility Lab, Department of Electrical and Computer Engineering
University of Missouri – Rolla
Rolla, MO 6549, USA
kai.xiao@umr.edu, davidjp@umr.edu, and drewniak@umr.edu

***Abstract*—The idea of separating the spatial and temporal subgridding interfaces is introduced in this paper. Based on this idea, the spatial and temporal subgridding algorithms can be developed and analyzed separately. The spatial algorithm was given in the previous paper. In this paper, the temporal subgridding algorithm is described and the stability is illustrated by the analytical formulation of a one-dimensional model. An FDTD code that combines the spatial and temporal subgridding algorithms is implemented. Numerical test models are calculated to show the stability and accuracy of the proposed method.**

Keywords-component; FDTD, subgridding, stability

I. INTRODUCTION

A subgridding method is very useful for an FDTD code because it can dramatically reduce the memory and CPU time requirements by introducing fine mesh only within the regions that need better resolution.

When a subgridding algorithm is applied, the whole computational domain will contain sub-domains of smaller cells. As a result, some spatial and temporal field components on the sub-mesh boundaries can not be directly calculated via the normal update process. Estimation of these 'missing' field components is the key issue of a subgridding algorithm. Many papers have been published in this area [1-7]. In general, all these methods use different interpolation and/or extrapolation schemes to couple the field components between the coarse-mesh and the fine-mesh at the subgridding interface.

In [7], a spatial subgridding algorithm is proposed that can handle mesh ratios of 1:2 and 1:3, and the stability was tested numerically. Three key points of this algorithm are:

- Recessed sub-mesh. The sub-mesh boundary is not co-located with the surfaces of the coarse cells, but recessed by a half (mesh ratio is 1:2) or two third (mesh ratio is 1:3) of the size of a coarse cell.
- Symmetric coupling schemes. The interpolation schemes coupling the field components between coarse-mesh and fine-mesh must be symmetric.
- Special treatment to the edge and corner cells. The interpolation pattern needs to be modified for corner and edge cells on the subgridding interface.

Following these requirements will keep the reciprocity of Maxwell's equations in the update equations and ensure the stability consequently. However, this algorithm can only do subgridding in space. In other words, the same time increment, which is limited by the size of the smallest cell, is used both in the main-domain and sub-domains. Apparently, the most efficient subgridding method should use different time increments in sub-meshes. This is especially important when recursive sub-meshes are used.

A full subgridding method actually consists of two sub-algorithms: the sub-space algorithm and the sub-time algorithm. Most of the published algorithms [1-4] do both spatial and time subgridding at the same subgridding interface. Actually, in a more general way, these two types of subgridding interfaces can be separated, as shown in Figure 1. First, a sub-domain is selected, and a smaller time increment is applied. Then, the spatial subgridding is implemented within this sub-time mesh.

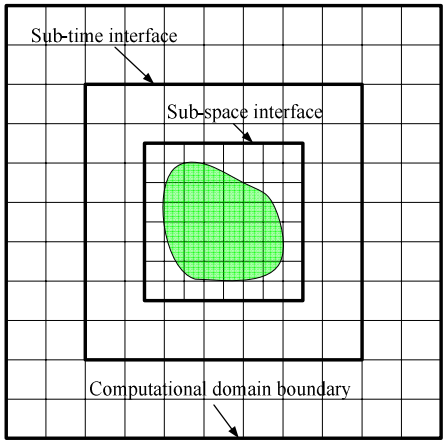

Figure 1. Separation of sub-time and sub-space interfaces

The advantages of such an arrangement are:

- It is necessary for the proposed subgridding algorithm. Because at the subgridding interface cells with reduced size are introduced, if the spatial and temporal subgridding interfaces are collocated, no gain in efficiency will be achieved.

- The sub-time and sub-space algorithms can be separated, and the stability and accuracy of each algorithm can be investigated individually.
- The sub-time algorithm can be applied in other situations, such as for non-uniform mesh, and it may also provide additional flexibility in locating the subgridding interfaces.

Based on this idea, the time subgridding algorithm is implemented and analyzed separately in this paper. Combined with the spatial subgridding algorithm given in [7], a full subgridding method is realized.

Like any other numerical algorithms, the two major concerns of a subgridding algorithm are stability and accuracy. Stability means that any numerical errors introduced in a certain time step, such as floating number errors, will not accumulate with the advance of update process, and accuracy means that the numerical solution can converge to the actual solution.

Two methods are widely used in the stability analysis of the finite difference update equations, the matrix method and the Fourier method [8-10]. The matrix method is used here because it is easy to handle an update process with a subgridding scheme.

The update equations of the FDTD method in matrix form are given by,

$$\begin{cases} \mathbf{H}^{n+\frac{1}{2}} = \mathbf{H}^{n-\frac{1}{2}} - \dfrac{\Delta t}{\mu \Delta l}\mathbf{D}_e\mathbf{E}^n \\ \mathbf{E}^{n+1} = \mathbf{E}^n + \dfrac{\Delta t}{\varepsilon \Delta l}\mathbf{D}_h\mathbf{H}^{n+\frac{1}{2}} \end{cases}, \qquad (1)$$

where $\mathbf{E}$ and $\mathbf{H}$ are one dimensional vectors, representing all the E-field and H-field components, respectively; $\mathbf{D}_e$ and $\mathbf{D}_h$ are the normalized discrete curl operators over $\mathbf{E}$ and $\mathbf{H}$; ε is the permittivity, μ is the permeability, and Δt is the time increment. For simplicity, assume the FDTD model is in a lossless homogeneous media and all the cells are cubic. Since the FDTD method maintains the reciprocity of Maxwell's equations, the two matrix operators have the relation of [5],

$$\mathbf{D}_h = \mathbf{D}_e^{\mathrm{T}}. \qquad (2)$$

Consider the update of $\mathbf{E}$ from time step (n-1) to (n+1). Based on Equation (1), we can eliminate $\mathbf{H}$, and obtain an iterative formula with only $\mathbf{E}$ as the update variable, which has the form of,

$$\mathbf{E}^{n+1} - (2\mathbf{I} - \frac{\Delta t^2}{\varepsilon\mu\Delta l^2}\mathbf{D}^{\mathrm{T}}\mathbf{D})\mathbf{E}^n + \mathbf{E}^{n-1} = 0, \qquad (3)$$

where $\mathbf{I}$ is a unit matrix of the same order of $\mathbf{D}^{\mathrm{T}}\mathbf{D}$. Define $\mathbf{A} = (2\mathbf{Y} - \frac{\Delta t^2}{\varepsilon\mu\Delta l^2}\mathbf{D}^{\mathrm{T}}\mathbf{D})$. Eq. (3) can also be obtained by directly discretizing the wave equation. If all the eigenvalues of matrix $\mathbf{A}$ are real and fall into [-2, 2], the update process will be stable [11]. Since $\mathbf{D}^{\mathrm{T}}\mathbf{D}$ is a semi-positive definite matrix, all its eigenvalues are real numbers and no less than zero. If the time increment is properly selected (according to Courant criteria), all the eigenvalues of $\mathbf{A}$ will satisfy the requirement, and consequently the update process is stable.

When a spatial subgridding algorithm is applied to an FDTD model, the interpolation pattern for the field coupling scheme at the subgridding interface will be included into $\mathbf{D}_e$ and $\mathbf{D}_h$. If the field coupling schemes are constructed properly so that the relationship in Eq. (2) still holds, the subgridding algorithm will be stable.

Another important property of a numerical algorithm is efficiency. Ideally, an algorithm can generate accurate results in a very short time. However, there is often a compromise between accuracy and efficiency. The aim of an FDTD subgridding method is to improve the modeling efficiency. The introduction of interpolation scheme to estimate the missing field components, however, will inevitably introduce numerical reflections at the subgridding interface, which will harm the accuracy of the results. From an engineering point of view, if the errors can be controlled within an acceptable range, the algorithm is practical.

In section II, the time subgridding algorithm is described, and its stability is analyzed using the matrix method based on a one-dimensional problem. Numerical examples are also given to show its stability and accuracy. In section III, the spatial and time algorithms are combined together, and results of the full subgridding method, the spatial subgridding method, and uniform mesh are compared. At the end, the conclusions are given.

II. TIME SUBGRIDDING ALGORITHM

When a full sub-time algorithm is applied to an FDTD model, the time increments in the coarse mesh and fine mesh are different. We call the domain using coarse time step as main-time domain, and the domain using sub-time step as sub-time domain. Since the field components in the sub-time domain need to be updated multiple times during one coarse time step in the main-time domain, there are missing field components in the sub-time steps at the time subgridding interface. How to estimate these missing field components is the key issue of a time subgridding algorithm.

Different schemes have been proposed by different authors. In [2] and [4], the 2nd order Taylor expansion is used to extrapolate the sub-time field components. However, in [4] an estimation-correction scheme is also applied by averaging the extrapolated results with that from the standard Yee update. A simple scheme for a time increment ratio of 1:2 was proposed in [5]. It assumes that the transverse H-field at the subgridding interface is constant between time steps.

To our knowledge, the algebraic formulation for stability analysis of temporal subgridding algorithm is not available in the publication. It is evident that the high order extrapolation

schemes yield a smoother field waveform in time domain. On the other hand, introducing extrapolation in the update process also increases the risk of late-time instability. It is often found that late-time instability occurs after trying different extrapolation schemes. The scheme given in [5] was tested to be stable. In this paper, the scheme is extended to the temporal subgridding ratios other than two, and the formulation for stability analysis is derived and analyzed.

The sub-time coupling scheme is shown in Figure 2. for the time increment ratios of 1:2 and 1:3. The basic idea is that the unknown H-field components in the sub-time grid on the subgridding interface are set to be equal to the closest available H-field components in main-time mesh. The time co-located E-fields in the sub-time mesh then will be coupled back directly to the main-time mesh.

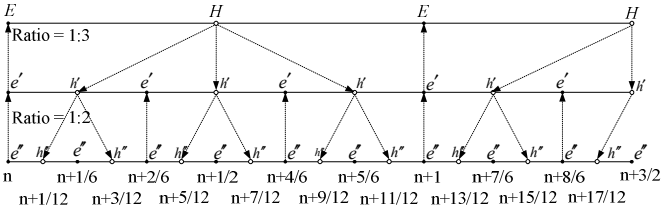

Figure 2. Field coupling scheme for the time subgridding algorithm

For illustrating the chain of thought used in the stability analysis, a one dimensional transmission line example is used here. Assume that the transmission line is lossless and shorted at the both ends, e.g. voltages at the terminal nodes are zero. Figure 3 shows the space-time diagram of a finite difference model the transmission line. The *V* and *I* in upper case denote the voltage and current nodes in the main-time grid, and the *v* and *i* are the nodes in sub-time grid. Assume that the number of voltage nodes in main-time grid is N, and the number in sub-time grid is M. The update variables (voltages and currents) in the main-time and sub-time grids can be written in vector form as given in Eq. (4). Since the update variables V_N and i_0 at the interface are coupled from v_1 and I_N, respectively, they are not included in the vectors.

$$\begin{aligned} \mathbf{V} &= (V_1, V_2, \cdots, V_{N-1}) \\ \mathbf{I} &= (I_1, I_2, \cdots, I_N) \\ \mathbf{v} &= (v_1, v_2, \cdots, v_M) \\ \mathbf{i} &= (i_1, i_2, \cdots, i_M) \end{aligned} \tag{4}$$

At the temporal subgridding interface, the field coupling scheme for this model are:

- From main-time grid to sub-time grid, the coupling scheme is $i_0^{n+\frac{1}{4}} = i_0^{n+\frac{3}{4}} = I_N^{n+\frac{1}{2}}$.
- From sub-time grid to main-time grid, the coupling scheme is $V_N^n = v_1^n$.

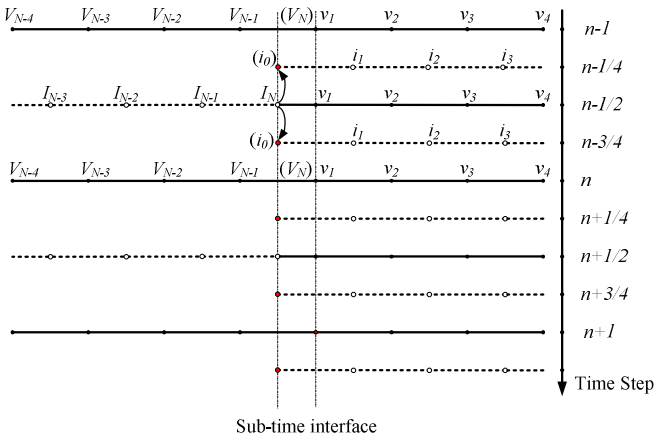

Figure 3. Sub-time update process for a 1-D transmission line model

The update equations at time step n for the main-time grid in matrix format have the form of,

$$\begin{cases} \mathbf{V}^n = \mathbf{V}^{n-1} - \Delta t \alpha \mathbf{D} \mathbf{I}^{n-\frac{1}{2}} \\ \mathbf{I}^{n+\frac{1}{2}} = \mathbf{I}^{n-\frac{1}{2}} - \Delta t \beta (-\mathbf{D}^{\mathrm{T}} \mathbf{V}^n + \mathbf{P} \mathbf{v}^n) \end{cases} \tag{5}$$

where Δt is the time increment in main-time grid; $\alpha = \dfrac{L}{\Delta l}$ and $\beta = \dfrac{C}{\Delta l}$ are the constant update coefficients, where *L* and *C* are the per unit length parameters and Δl is the size of the mesh; **D** is the discrete difference operator; **P** is the matrix that couples the voltage from sub-time grid to main-time grid at the interface. The matrix **D** and **P** are given by,

$$\mathbf{D} = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \\ & & & & -1 \quad 1 \end{pmatrix}_{(N-1)\times N} \tag{6}$$

$$\mathbf{P} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \end{pmatrix}_{N\times M} \tag{7}$$

Similarly, the update equations in matrix form for sub-time grid can be written as,

$$\begin{cases} \mathbf{v}^{n-\frac{1}{2}} = \mathbf{v}^{n-1} - \dfrac{\Delta t}{2}\alpha(\widetilde{\mathbf{D}}\mathbf{i}^{n-\frac{3}{4}} + \mathbf{Q}\mathbf{I}^{n-\frac{1}{2}}) \\ \mathbf{i}^{n-\frac{1}{4}} = \mathbf{i}^{n-\frac{3}{4}} - \dfrac{\Delta t}{2}\beta(-\widetilde{\mathbf{D}}^{\mathrm{T}}\mathbf{v}^{n-\frac{1}{2}}) \end{cases} \tag{8}$$

where $\widetilde{\mathbf{D}}$ is the difference operator matrix, and $\mathbf{Q}$ is the matrix that couples the current on the time interface from the main-time grid to the sub-time grid. They have the form of,

$$\widetilde{\mathbf{D}} = \begin{pmatrix} 1 & & & & \\ -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{pmatrix}_{M\times M} \tag{9}$$

$$\mathbf{Q} = \begin{pmatrix} 0 & \cdots & 0 & -1 \\ 0 & \cdots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{pmatrix}_{M\times N} \tag{10}$$

Considering the voltage update from time step (n-1) to (n+1) in main-step grid, we can eliminate **I** based on (5) and obtain the iterative formula, as given in Eq. (11), where $\mathbf{Y}_{N-1}$ is a (N-1) order unit matrix.

$$\mathbf{V}^{n+1} - (2\mathbf{Y}_{N-1} - \Delta t^2\alpha\beta\mathbf{D}\mathbf{D}^{\mathrm{T}})\mathbf{V}^n - \mathbf{V}^{n-1} - \Delta t^2\alpha\beta\mathbf{D}\mathbf{P}\mathbf{v}^n = 0 \tag{11}$$

Similarly, we can derive that the iterative formula in sub-time domain, as given in (12).

$$\mathbf{v}^{n+1} + (2\mathbf{Y}_M - \mathbf{R}^2 - \frac{\Delta t^2}{2}\alpha\beta\mathbf{R}\mathbf{Q}\mathbf{P})\mathbf{v}^n + \mathbf{v}^{n-1} + \frac{\Delta t^2}{2}\alpha\beta\mathbf{R}\mathbf{Q}\mathbf{D}^{\mathrm{T}}\mathbf{V}^n = 0 \tag{12}$$

where $R = 2\mathbf{Y}_M - \dfrac{\Delta t^2}{4}\alpha\beta\widetilde{\mathbf{D}}\widetilde{\mathbf{D}}^{\mathrm{T}}$; $\mathbf{Y}_M$ is an M-order unit matrix.

It needs to stress that, from time step (n-1) to (n+1), the update in sub-time grid will be performed for four update cycles. From (12), however, it is seen that all the *v* at sub-time steps can be eliminated. We found that it may not be possible to obtain such a concise iterative formula for the coupling schemes that apply extrapolation. Combining (11) and (12), an iterative formula of the whole update process can be derived, as given in (13),

$$\begin{bmatrix} V^{n+1} \\ v^{n+1} \end{bmatrix} + \begin{bmatrix} -(2\mathbf{Y}_{N-1} - \Delta t^2\alpha\beta\mathbf{D}\mathbf{D}^{\mathrm{T}}) & -\Delta t^2\alpha\beta\mathbf{D}\mathbf{P} \\ \dfrac{\Delta t^2}{2}\alpha\beta\mathbf{R}\mathbf{Q}\mathbf{D}^{\mathrm{T}} & 2\mathbf{Y}_M - \mathbf{R}^2 - \dfrac{\Delta t^2}{2}\alpha\beta\mathbf{R}\mathbf{Q}\mathbf{P} \end{bmatrix} \begin{bmatrix} V^n \\ v^n \end{bmatrix} + \begin{bmatrix} V^{n-1} \\ v^{n-1} \end{bmatrix} = 0 \tag{13}$$

From convenience, we define that,

$$\mathbf{A} = \begin{bmatrix} -(2\mathbf{Y}_{N-1} - \Delta t^2\alpha\beta\mathbf{D}\mathbf{D}^{\mathrm{T}}) & -\Delta t^2\alpha\beta\mathbf{D}\mathbf{P} \\ \dfrac{\Delta t^2}{2}\alpha\beta\mathbf{R}\mathbf{Q}\mathbf{D}^{\mathrm{T}} & 2\mathbf{Y}_M - \mathbf{R}^2 - \dfrac{\Delta t^2}{2}\alpha\beta\mathbf{R}\mathbf{Q}\mathbf{P} \end{bmatrix} \tag{14}$$

Eq. (13) has the similar form as Eq. (3). As discussed in the introduction section, if the absolute value of each eigenvalue of the matrix **A** is real and no more than two, the update process is stable. According to matrix spectral radius theorem (See Appendix A), the maximum absolute value of the eigenvalues of a matrix is bounded by any norm of the matrix.

We select the time increment in the main-step grid to the upper limit of the value given by Courant criteria, e.g. $\Delta t = \dfrac{\Delta l}{\sqrt{LC}}$. Based on (6), (7), (9), and (10), it can be shown that the infinity-norm of **A** is two. For example, a typical matrix **A** is given in Eq. (15), where N = 4 and M = 6.

$$\mathbf{A} = -\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{\frac{7}{8}} & \frac{1}{4} & \mathbf{\frac{13}{16}} & \frac{1}{16} & 0 & 0 & 0 \\ 0 & 0 & \mathbf{\frac{1}{8}} & \mathbf{\frac{11}{16}} & \frac{3}{8} & \frac{3}{4} & \frac{1}{16} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{16} & \frac{3}{4} & \frac{3}{8} & \frac{3}{4} & \frac{1}{16} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{16} & \frac{3}{4} & \frac{3}{8} & \frac{3}{4} & \frac{1}{16} \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{16} & \frac{3}{4} & \frac{3}{8} & \frac{3}{4} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{16} & \frac{3}{4} & \frac{5}{16} \end{bmatrix} \tag{15}$$

It can be seen from (15), all the elements of the matrix are negative. The matrix is symmetric except the elements in bold, which are related to the temporal subgridding scheme. It is easy to obtain that the infinity norm of the matrix is 2, which is a necessary condition for a stable scheme. Since matrix **A** is not symmetric any more, it is not easy to prove that all the eigenvalues are real analytically. Instead, we calculated all the eigenvalues with enumerating M and N, and found all the eigenvalues are real numbers when M and N are set up to 500.

The process of the derivation of matrix **A** can also be extended to three dimensions.

Although a complete proof of the stability is not available, the iterative formula in (13) provides the possibility of the stability analysis of a temporal subgridding scheme, and it may also helpful for developing new temporal subgridding scheme.

For 3-D case, a test model, a scattered problem of a lossless dielectric sphere in a cavity with an incident plane wave source, is calculated to show the stability of the temporal subgridding.

The computational domain is cubic with the dielectric sphere $(\varepsilon_r = 4)$ in the center. There are forty cubic cells along each axis, and each cell has the size of $3\,cm \times 3\,cm \times 3\,cm$. The dielectric sphere is fully enclosed in the sub-time domain, and a temporal subgridding ratio of 3:1 is used. An E-field probe is placed at the center of the sphere, and the recorded E-field waveform is given in Figure 4. It can be seen from Figure 4 that, the field is oscillating but the amplitude is bounded. After one million time steps in the sub-time grid, there is no sign of instability observed.

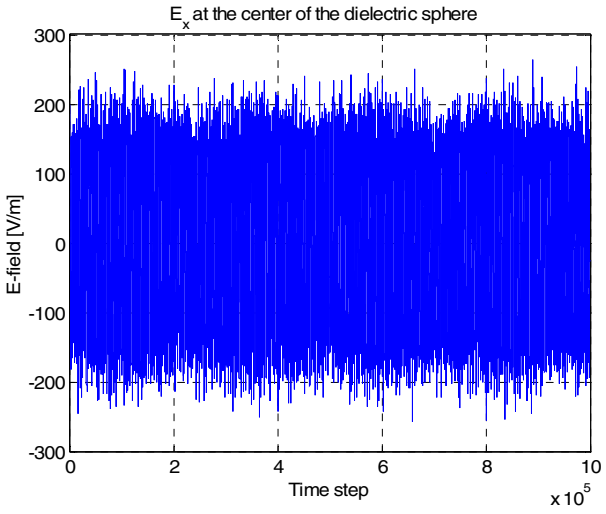

Figure 4. The E_x at the center of the dielectric sphere

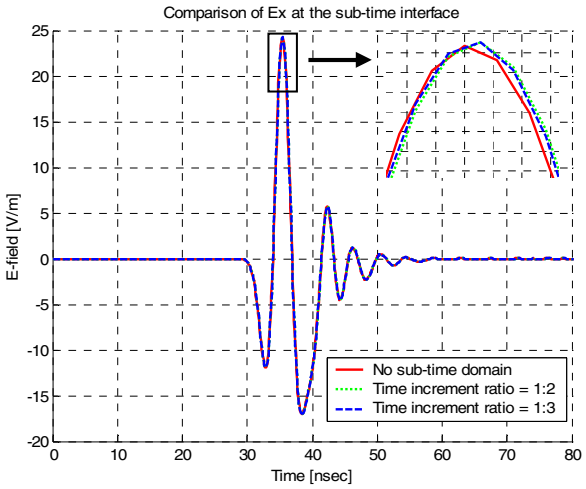

Figure 5. Comparison of E-field at the sub-time interface

III. FULL SUBGRIDDING METHOD

An FDTD code with implementing the spatial and temporal subgridding algorithm and their combination was developed. The stability of the spatial and temporal algorithms was tested separately in [7] and this paper, respectively. The stability of the full subgridding algorithm that combines the spatial and temporal schemes was also tested numerically, and stable results were obtained.

The accuracy of the full subgridding algorithm was tested using the model shown in Figure 6, which consists of a dipole source, which is implemented with two PEC stabs connected with a lumped 50-ohm voltage source excited with a Gaussian pulse. Absorbing boundary conditions are used. In the first calculation, no sub-grid was used. The calculation was then repeated with the presence of a sub-mesh in free-space. A field probe was placed right before the sub-mesh. The results were compared in Figure 7. The spatial and temporal subgridding ratios are both three.

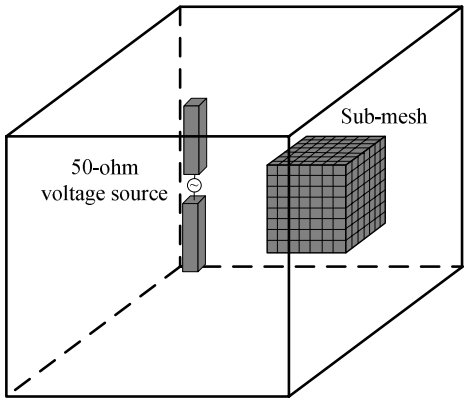

Figure 6. An FDTD mdoel for stability test of the subgridding method. (The sub-mesh is created in free space.)

It shows in Figure 7, the E-field waveforms agree very well. The difference between them with and without the sub-mesh provides information on the reflection expressed in the time domain, as given in Figure 8. From Figure 8, we can see that reflection caused by the full subgridding scheme is only slightly bigger that that by the pure spatial subgridding scheme.

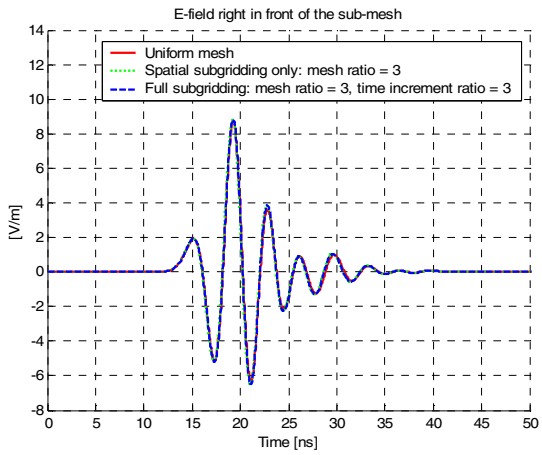

Figure 7. Comparison of E-field with and without sub-mesh.

When a sub-time interface is introduced, numerical reflections at the interface will be inevitable. To test how the reflections may compromise the accuracy of the simulation result, the similar dielectric sphere scattering problem is calculated. However, instead of placing the sphere in a cavity, absorbing boundary conditions were used. Two cases with the time increment ratio of two and three are calculated. The results are compared in Figure 5, from which it can be seen that

the three curves agree very well. When zooming into a peak, it can be observed that the curves have a small offset in time.

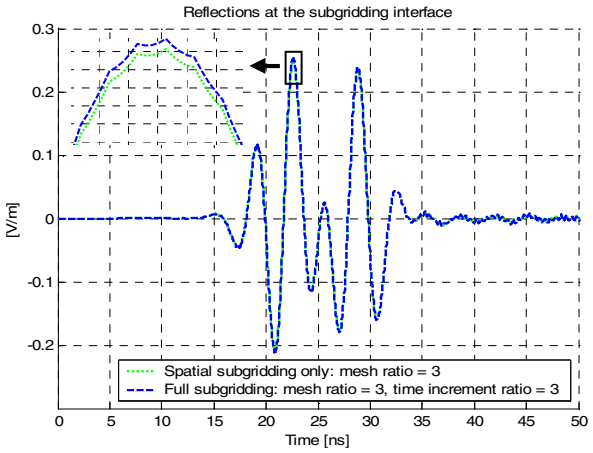

Figure 8. Reflection at the subgridding interface.

IV. CONCLUSION

In this paper, the idea of separating the spatial and temporal subgridding interfaces is proposed. Based on this idea, the spatial subgridding and temporal subgridding algorithm can be developed and investigated separately, which can reduce the complexity of the implementation and analysis. It also provides users with the more flexibility in locating the sub-domains and choosing the spatial and temporal subgridding ratios. Additionally, the temporal subgridding can be applied in other situations, such as non-uniform meshing.

A simple temporal subgridding algorithm is used, and its stability is analyzed based on a one-dimensional transmission line model. A 3-D model, a dielectric sphere in a cavity illuminated with a plane wave, was calculated to show the stability of the temporal subgridding algorithm. After one million time steps in the sub-time grid, no sign of instability was observed. Accuracy of the temporal subgridding algorithm was also tested. Results show that the reflections at the sub-time interface can cause a very small offset of the field curves in time domain.

An FDTD code with the combination of the spatial and temporal subgridding algorithms was implemented. The numerical reflection caused by the full subgridding scheme is only slightly larger than that of the pure spatial subgridding algorithm.

V. APPENDIX

A. *Matrix Spectral Radius Theorem* [11]

Let $(\lambda_1, \lambda_2, \cdots, \lambda_n)$ be the (real or complex) eigenvalues of a matrix **A**. Define $\rho(\mathbf{A}) := \max(|\lambda_i|)\ i = 1,2,\cdots n$, where $\rho(\mathbf{A})$ is called the spectral radius of the matrix **A**.

Theorem: Spectral radius of a matrix **A** is less of any form of the norm of the matrix.

REFERENCES

[1] M. Okoniewski, E. Okoniewska, and M. A. Stuchly, "Three-dimensional subgridding algorithm for FDTD," IEEE Trans. Antenna and Propagat., vol. 45, pp. 422-429, March, 1997

[2] M. W. Chevalier, R. J. Luebbers, and V. P. Cable, "FDTD local grid with materials transverse," IEEE Trans. Antenna and Propagat., vol. 45, pp 411-421, March, 2001

[3] M. J. White, Z. Yun, and M. F. Iskander, "A new 3-D FDTD multigrid technique with dielectric traverse capabilities," IEEE Trans. Microwave Theory Tech., vol. 49, pp. 422-430, March, 2001

[4] N. Chavannes, Local Mesh Refinement Algorithm for Enhanced Modeling Capabilities in the FDTD Method, Ph.D. dissertation, Swiss Federal Institute of Techology, Zurich, 2002

[5] P. Thoma and T. Weiland, "A consistant subgridding scheme for the finite difference time domain method," Int. Journal of Numerical Modelling: Electronic Networks, Devices and Fields, vol. 9, pp. 359-374, Sept.-Oct., 1996

[6] O. Poděbrad, M. Clemens, and T. Weiland, "New flexible subgridding scheme for the finite integration technique," IEEE Trans. Magnetics, vol. 39, pp1662-1665, May, 2003

[7] K. Xiao, D. Pommerenke, and J. Drewniak, "A three dimensional FDTD subgridding algorithm based on interpolation of current density," Proceeding of IEEE EMC Symposium '04 Santa Clara, CA. vol. 1, pp. 118-123, August, 2004

[8] G. D. Smith, Numerical Solution of Parital Differential Equations, Oxford University Press, New York and London, 1965

[9] Aslak Tveito and Ragner Winther, Introduction to Partial Differential Equations – A Computational Approach, Springer-Verlag, New York 1991

[10] A. Taflove, Computational Electromagnetics: The Finite-Difference Time-Domain Method, Artech House, Boston-London, 1995

[11] S. D. Gedney, and J. Roden, "Numerical Stability of Non-orthogonal FDTD Methods," IEEE Trans. on Antenna and Propagation, vol. 48, No.2, Feb., 2000

[12] Peter Lancaster and Miron Tismenetsky, The Theory of Matrices: Second Edition with Applications, Academic Press, Inc., 1985