A Leapfrog Alternating-Direction Hybrid Implicit-Explicit FDTD Method

A. Van Londersele, D. De Zutter, D. Vande Ginste
Department of Information Technology (INTEC)
Electromagnetics (EM) Group
Ghent University, Belgium
arine.vanlondersele@ugent.be, daniel.dezutter@intec.ugent.be, dries.vande.ginste@intec.ugent.be

Abstract—A novel curl splitting technique is proposed which makes the leapfrog ADI-FDTD method better equipped for configurations requiring a higher resolution in only one or two dimensions. The proposed method, which is given the name “leapfrog ADHIE-FDTD”, hybridizes implicit and explicit update equations such that the time step is solely bounded by the spatial steps in preferred dimensions. This weak conditional stability is rigorously proven and numerically validated.

I. INTRODUCTION

The quest for Finite-Difference Time-Domain (FDTD) solvers that are able to tackle large multiscale problems in a reasonable time span, comes down to finding clever tricks to overcome the notorious Courant stability limit, which relates the maximum allowed time step to the cell size. Thereto, numerous (partially) implicit methods have been developed, one of the most successful being the Alternating-Direction-Implicit (ADI) method, which requires the solution of a set of low-rank tridiagonal systems that scale with only one dimension. The traditional ADI method has evolved from a split-step update scheme [1] to a more efficient one-step leapfrog update scheme [2], but the core of the algorithm has always relied on an unconditionally stable method, i.e. without intrinsic time step bound. Here, we propose a new type of curl splitting that results in a so-called Hybrid Implicit-Explicit (HIE) scheme [2], but the core of the algorithm has always relied on the state-space-based approach described in [3]. The update matrices A and

\begin{equation}
\begin{pmatrix}
A
\end{pmatrix} = \begin{pmatrix}
\left(1+\frac{2\pi n}{\Delta x}\right) I + \left(\frac{\Delta t}{\Delta x}\right)^2 C_1 C_1^T & -\Delta \tau C \\
0 & I + \left(\frac{\Delta t}{\Delta x}\right)^2 C_2^T C_2
\end{pmatrix}
\begin{pmatrix}
Z^{-1} e^{n+1} \\
h^{n+0.5}
\end{pmatrix}
\end{equation}

where \( \sigma \) is the electrical conductivity, \( Z = (\mu/\varepsilon)^{1/2} \) the wave impedance, \( \Delta \tau = c \Delta t \) the time step rescaled by the phase velocity \( c = (\mu\varepsilon)^{-1/2} \), and \( \alpha \in [0,1] \) a tunable parameter. In the extreme case that \( \alpha \) is infinitely large, (2) is the conventional FDTD method. The smaller \( \alpha \), the larger the perturbation introduced by the ADI method and the larger the numerical error. However, as we will see in the next section, a smaller \( \alpha \) yields a larger maximum allowed time step. Note that \( C_1 C_1^T \) and \( C_2^T C_2 \) are tridiagonal matrices which only contribute to the updates of \( e_z \) and \( h_z \). In other words, the actual implementation of (2) comprises:

1) the explicit update of \( h_x \) and \( h_y \)
2) the implicit update of \( h_z \)
3) the explicit update of \( e_x \) and \( e_y \)
4) the implicit update of \( e_z \)

where steps 1 and 2 as well as steps 3 and 4 can be performed simultaneously. In contrast to conventional HIE methods, the leapfrog ADHIE method preserves stencils that scale with only one dimension, no matter how many dimensions are implicitized. Also, if it would be applied locally, e.g. in a nonuniform grid or in a subgrid, the system that needs to be solved does not blow up if two implicitized regions with implicitization along different axes cross each other. Compared to the traditional leapfrog ADI method, \( C_1 C_1^T \) and \( C_2^T C_2 \) are sparser, making the leapfrog ADHIE method more accurate.

II. FORMULATION

Suppose we want to eliminate the \( x \)-dependence from the Courant limit because we need to resolve an object that is thin along the \( x \)-axis. To do so, the curl \( \nabla \times \mathbf{E} = \nabla E_y - \nabla \times \mathbf{E} = \nabla E_z = \nabla \times \mathbf{H} = 0 \) is split into three parts

\begin{equation}
C_0 = \begin{bmatrix}
0 & -\partial z & \partial y \\
\partial z & 0 & 0 \\
-\partial y & 0 & 0
\end{bmatrix}, \quad
C_1 = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & -\partial x
\end{bmatrix}, \quad
C_2 = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & -\partial x \\
0 & 0 & 0
\end{bmatrix}
\end{equation}

The occurring derivatives are discretized by central differences on the conventional Yee lattice. The proposed leapfrog ADHIE scheme is then given by

III. STABILITY

Our stability analysis strongly relies on the state-space-based approach described in [3]. The update matrices A and...
\( B \) are decomposed into a symmetric contribution \( E \) and a skew-symmetric contribution \( F \) as follows
\[
A = E + F \quad B = E - F,
\]
with
\[
E = \begin{bmatrix}
I + \left( \frac{\Delta \tau}{2} \right)^2 C_1 C_1^T & -\frac{\Delta \tau}{2} C \\
-\frac{\Delta \tau}{2} C^T & I + \left( \frac{\Delta \tau}{2} \right)^2 C_2 C_2^T
\end{bmatrix}
\]
\[
F = \begin{bmatrix}
\frac{\Delta \tau}{2} C^T \\
\frac{\Delta \tau}{2} C
\end{bmatrix}
\]
As thoroughly explained in [3], our method is stable if and only if \( E \) is positive definite and \( F + F^T \) is positive semi-definite. The latter is trivially true if \( \sigma \geq 0 \). For the former, note that \( E \) can be written as \( E = G^T G + H \) with
\[
G = \begin{bmatrix}
\alpha I & -\frac{\Delta \tau}{2} C_2 \\
-\frac{\Delta \tau}{2} C_1^T & \alpha I
\end{bmatrix}
\]
\[
H = \begin{bmatrix}
(1-\alpha^2)I & -\frac{\Delta \tau}{2} C_0 \\
-\frac{\Delta \tau}{2} C_0^T & (1-\alpha^2)I
\end{bmatrix}
\]
Since the sum of positive definite matrices remains positive definite, the positive definiteness of \( H \) is a sufficient condition for numerical stability. Using the insights of [3], it should be clear that for the curl splitting (1) the condition \( H > 0 \) is satisfied if
\[
\Delta \tau < \frac{(1-\alpha^2)}{\sqrt{\frac{\Delta \tau}{\alpha}} + \frac{1}{\alpha}} \quad \alpha \in [0, 1].
\]
No \( \Delta \tau \) occurs in (8). If both \( \Delta x \) and \( \Delta y \) need to be eliminated from the stability limit, the curl is split as follows:
\[
C_0 = \begin{bmatrix} 0 & -\partial z & 0 \\ \partial z & 0 & 0 \end{bmatrix} \quad C_1 = \begin{bmatrix} 0 & 0 & \partial y \\ 0 & 0 & 0 \end{bmatrix} \quad C_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \partial y & 0 \\ 0 & 0 & -\partial x \end{bmatrix}
\]
and (8) is translated to
\[
\Delta \tau < (1-\alpha^2)\Delta z \quad \alpha \in [0, 1].
\]
If the time step needs to be independent of the spatial lattice, the traditional leapfrog ADI method is used, which has \( \alpha = 1 \) and \( C_0 = 0 \) such that \( H = 0 \), and further splits the curl into
\[
C_1 = \begin{bmatrix} 0 & 0 & \partial y \\ \partial z & 0 & 0 \end{bmatrix} \quad C_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \partial y & 0 \\ 0 & 0 & -\partial x \end{bmatrix}
\]
Finally, note that (8) and (10) are sufficient conditions. Numerical examples have shown that they can heavily underestimate the maximum allowed time step. Also, by rewriting \( E \) as
\[
E = \begin{bmatrix}
I & -\frac{\Delta \tau}{2} C \\
-\frac{\Delta \tau}{2} C^T & I
\end{bmatrix} + \left( \frac{\Delta \tau}{2} \right)^2 \begin{bmatrix}
C_1 C_1^T & 0 \\
0 & C_2 C_2^T
\end{bmatrix},
\]
it is clear that stability is ensured if the first matrix is positive definite, which is the case if
\[
\Delta \tau < \frac{1}{\sqrt{\frac{\Delta \tau}{\alpha} + \frac{1}{\alpha^2}}} \quad \alpha \in [0, \infty],
\]
such that the 3D Courant limit is always a valid upper bound for the leapfrog ADHIE method, independent of the curl splitting or the value of \( \alpha \).

**IV. NUMERICAL VALIDATION**

Consider a vacuum cavity divided into \( 30 \times 30 \times 60 \) cells that are uniformly spaced with steps \( \Delta x = 20 \mu \text{m} \) and \( \Delta y = \Delta z = 2 \mu \text{m} \). A magnetic dipole source \( J_\mu \) with a bandwidth of 10 GHz is placed close to one corner and the electric field component \( E_x \) is recorded close to the opposite corner. The simulation bridges a physical time of 6.67 ns, part of which is shown in Fig. 1. The conventional, fully explicit FDTD method with time step equal to the 3D Courant limit (13) serves as a reference. The leapfrog ADHIE method with \( \alpha = 0.5 \) and time step given by (8) is compared to the leapfrog ADI method in terms of CPU time (Table I) and accuracy (Fig. 1). For identical time steps, the leapfrog ADI method, which has to solve six linear systems instead of two, is about two times slower. For a time step about twice as large, the CPU times become comparable. As expected, the traditional leapfrog ADI method is in both cases less accurate than the proposed leapfrog ADHIE method. In conclusion, the proposed leapfrog ADHIE-FDTD method is found to be a powerful tool to tackle multiscale problems such as the cavity example given above. Future work focuses on building a subgridding scheme based on this technique, which has one global time step specified by the coarse part of the grid.

**REFERENCES**