



# Higher-order hybrid implicit/explicit FDTD time-stepping



W. Tierens

Max Planck Institut für Plasmaphysik, Germany

## ARTICLE INFO

### Article history:

Received 17 July 2016

Received in revised form 29 September 2016

Accepted 30 September 2016

Available online 4 October 2016

### Keywords:

FDTD

High order

Partially implicit

## ABSTRACT

Both partially implicit FDTD methods, and symplectic FDTD methods of high temporal accuracy (3rd or 4th order), are well documented in the literature. In this paper we combine them: we construct a conservative FDTD method which is fourth order accurate in time and is partially implicit. We show that the stability condition for this method depends exclusively on the explicit part, which makes it suitable for use in e.g. modelling wave propagation in plasmas.

© 2016 Elsevier Inc. All rights reserved.

## 1. Introduction

The Finite Difference Time Domain (FDTD) algorithm is a popular computational method for solving Maxwell's equations in time domain [12]. FDTD is explicit, and, like most explicit methods, it has a stability condition which puts an upper bound on the time step. Usually this condition is

$$c\Delta_t \leq \Delta/\sqrt{d} \quad (1)$$

where  $c$  is the speed of light,  $\Delta_t$  the time step,  $\Delta$  the space step (assuming cells of unit aspect ratio), and  $d$  the number of spatial dimensions.

Many attempts have been made to hybridise FDTD with stabler implicit methods, such that particular difficult sub-problems of limited size can be handled implicitly, while the overall algorithm retains the efficiency of explicit FDTD. Sometimes this is done in the context of spatial refinement, where small features need to be resolved and the discretisation length must be (locally) small [4,8,2]. In other cases this is used to improve the behaviour of auxiliary differential equations which describe materials in the simulation.

Such hybrid methods are used to simulate electromagnetic waves in both magnetised and unmagnetised plasmas [10, 13,11]. Thanks to this partially implicit approach, these algorithms remain stable at the vacuum stability condition even in dense plasmas.

Partially implicit FDTD algorithms have two main advantages

- Unlike fully explicit approaches, their stability condition is not sensitive to parameters of the implicit sub-problem.
- Unlike fully implicit approaches, the implicit equations have a structure which can be exploited to solve them efficiently (they are typically block-diagonal or banded, or of constant size).

E-mail address: wouter.tierens@ipp.mpg.de.

Standard FDTD uses leapfrog time-stepping [12]. It is second-order accurate in time. Extending FDTD to third or fourth order accuracy in time is relatively straightforward: all it requires is making multiple leapfrog-like time steps with modified coefficients [6,9,7]. Doing so retains FDTD's symplecticity (a discrete energy remains exactly conserved).

In this paper, we will show how to construct a fourth-order accurate time stepping operator which is partially implicit: it retains the two main advantages listed above. We then apply this technique to simulate electromagnetic waves in unmagnetised plasmas.

In section 2, we construct a fourth-order accurate partially implicit time stepping operator, and show that it has the desired characteristics. Then in section 3, we apply our method to more realistic problems.

## 2. Constructing a fourth-order accurate hybrid implicit/explicit time stepping operator

### 2.1. In vacuum

As an introduction, let us first consider the vacuum case. Wave propagation is described by Maxwell's equations

$$\frac{\partial \vec{B}}{\partial t} = -\vec{\nabla} \times \vec{E} \quad (2)$$

$$\frac{\partial \vec{E}}{\partial t} = \frac{1}{\epsilon_0 \mu_0} \vec{\nabla} \times \vec{B} \quad (3)$$

which can be transformed in anti-symmetric form using  $\vec{\mathcal{E}} = \sqrt{\epsilon_0} \vec{E}$ ,  $\sqrt{\mu_0} \vec{B} = \vec{\mathcal{B}}$

$$\frac{\partial \vec{\mathcal{B}}}{\partial t} = -c \vec{\nabla} \times \vec{\mathcal{E}} \quad (4)$$

$$\frac{\partial \vec{\mathcal{E}}}{\partial t} = c \vec{\nabla} \times \vec{\mathcal{B}} \quad (5)$$

We wish to focus on temporal discretisation. Let us assume (without loss of generality) that only  $\mathcal{E}_y$  and  $\mathcal{B}_z$  are nonzero, and that they are proportional to  $\exp(ikx)$

$$\frac{\partial \mathcal{B}_z}{\partial t} = ick \mathcal{E}_y \quad (6)$$

$$\frac{\partial \mathcal{E}_y}{\partial t} = ick \mathcal{B}_z \quad (7)$$

In matrix form

$$\frac{\partial}{\partial t} \begin{bmatrix} \mathcal{E}_y \\ \mathcal{B}_z \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & ick \\ ick & 0 \end{bmatrix}}_M \begin{bmatrix} \mathcal{E}_y \\ \mathcal{B}_z \end{bmatrix} \quad (8)$$

The exact solution is

$$\begin{bmatrix} \mathcal{E}_y(t) \\ \mathcal{B}_z(t) \end{bmatrix} = \exp(Mt) \begin{bmatrix} \mathcal{E}_y(0) \\ \mathcal{B}_z(0) \end{bmatrix} \quad (9)$$

Note that the matrix  $M$  is anti-Hermitian. Its eigenvalues are therefore purely imaginary, and the eigenvalues of  $\exp(Mt)$  lie on the unit circle for all  $t$ . This means that this eigenmode has a constant amplitude: it neither decays to zero, nor diverges to infinity, as time goes on. It is stable and lossless.

The most common way of discretising (8) is the leapfrog approach

$$\begin{bmatrix} \mathcal{E}_y(t + \Delta_t) \\ \mathcal{B}_z(t + \frac{3}{2}\Delta_t) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 \\ ick\Delta_t & 1 \end{bmatrix}}_{M_E} \underbrace{\begin{bmatrix} 1 & ick\Delta_t \\ 0 & 1 \end{bmatrix}}_{M_B} \begin{bmatrix} \mathcal{E}_y(t) \\ \mathcal{B}_z(t + \frac{1}{2}\Delta_t) \end{bmatrix} \quad (10)$$

$M_E M_B$  is indeed a second-order accurate approximation of (a staggered version of)  $\exp(M\Delta_t)$ . Its eigenvalues are also on the unit circle, conditional on  $\Delta_t$  being sufficiently small:

$$\Delta_t \leq \frac{2}{c|k|} \quad (11)$$

If  $\Delta_t$  obeys (11),  $M_E M_B$  is stable and lossless, like the continuous case.

It was shown in [7,6] that repeated steps of the form (10) with modified coefficients can be used to achieve better than second order accuracy. For example, third order accuracy is achieved using the following steps:

Download English Version:

<https://daneshyari.com/en/article/4967973>

Download Persian Version:

<https://daneshyari.com/article/4967973>

[Daneshyari.com](https://daneshyari.com)