Implicit-Explicit Time Integration of a High-Order Particle-in-Cell Method with Hyperbolic Divergence Cleaning.

G.B. Jacobs [a] J.S. Hesthaven [b]

[a] *Department of Aerospace Engineering & Engineering Mechanics, San Diego State University, San Diego, CA, 91941*

[b] *Division of Applied Mathematics, Brown University Providence, RI 02906*

Abstract

A high-order implicit-explicit additive Rung-Kutta time integrator is implemented in a particle-in-cell method based on a high-order discontinuous Galerkin Maxwell solver for simulation of plasmas. The method satisfies Gauss law using a hyperbolic divergence cleaner that transports divergence out of the computational domain at several times the speed of light. The stiffness in the field equations induced by high transport speeds is alleviated by an implicit time integration, while an explicit time integration ensures a computationally efficient particle update. Simulations on a plasma wave and a Weibel instability show that the implicit-explicit solver is computationally efficient, allowing for computations with high divergence transport speeds that ensure an accurate representation of the governing plasma equations. The high-order method only requires two time steps per plasma wave period. Numerical instability appears when the time step exceeds the plasma frequency time scale. A divergence transport speed of approximately ten times the speed of light is shown to be optimal, since it combines an accurate representation of Gauss law with a small influence of numerical noise on the solution.

Key words: Time Integration, Particle-in-Cell, Plasmas, Additive Runge-Kutta, Implicit-Explicit, High-Order.

1 Introduction

Particle-in-cell (PIC) methods have proved a valuable tool for the simulation of a variety of space and laboratory plasma problems. In the PIC approach, the electric and magnetic field are solved by means of the Maxwell equations and/or a Poisson equation in the Eulerian framework. Charged plasma particles are forced by the fields and tracked in a Lagrangian framework. Particles

are coupled with the field solver by weighting the sum of the Coulomb forces onto the grid in the form of a charge or current density. A large class of plasma problems with relevance to high power microwave generation and fusion problems can be accurately solved with particle-in-cell (PIC) methods.

PIC methods typically use explicit time integration. Even though plasma physics are characterized by a large range of space and time scales, the physical stiffness of many plasma problems is relatively low. As a result, the time step restriction induced by stability in explicit methods is not too severe. Since explicit PIC methods are simple and computationally efficient, they have been predominant.

The main stream PIC methods for computation of plasmas use the second order-staggered grid for the electromagnetic field solver according to Yee [1]. A symplectic leap frog method is used to update the time. A major advantage of the symplectic scheme is that it is momentum conservative. Downsides of the leap frog are that they are limited to second order, and prone to dispersion errors. The leap frog method introduces a splitting between the particle update and the electromagnetic field solver that limits temporal accuracy to second order. Dispersion errors are highly dependent on a properly chosen CFL criterion and uniform spatial grid spacing. The inherent diffusion and dispersion of the low order method is limiting the long-time simulation of full scale problems.

In Jacobs and Hesthaven [9] a PIC algorithm based on a high-order nodal discontinuous Galerkin solver (DG-PIC) was introduced that addresses some issues related to low order resolution. The high-order PIC solver is based on the DG Maxwell's solver [5] that ensures low dispersion, while providing geometric flexibility and excellent stability properties. A consistent high-order coupling between the particles and the Maxwell's solver preserves the high order resolution of the method. Time integration with a high-order Runge-Kutta scheme ensures high-order temporal accuracy without splitting errors. The high-order nature of the method also reduces errors resulting from the finite grid instability.

In low order PIC the exact charge conserving scheme proposed in [14] is widely used since it eliminates the need to directly impose Gauss law to ensure charge conservation. This charge conservation is, however, very difficult to implement in the high-order PIC, and explicit divergence control is necessary. The hyperbolic divergence cleaning discussed in [9; 13] was shown to be a particular good choice. The DG discretization of the hyperbolic Maxwell equations is easily extended to the hyperbolic divergence cleaning equations. The classic Boris divergence cleaning method [1] requires the solution of a Poisson equation for the correction potential. The DG approximation of the Poisson equation with an elliptic character is more challenging and less established than DG approx-

imations of hyperbolic equations. Moreover, the Boris method was shown to be one order less accurate than the DG-PIC with hyperbolic cleaning.

A drawback of the hyperbolic cleaner is that it stiffens the PIC field solver. The divergence is cleaned along characteristics whose speed is several times the speed of light. With the increase of the largest characteristic speed, the stable time step reduces for the field update, when explicit time integration is used. The same smaller time step must be used for the particle update that is coupled to the solver leading to unnecessary extra iterations of the computationally expensive particle update.

A common method to relax the time step restriction is to use implicit time integration. In the non-linear system of coupled Eulerian-Lagrangian equations the implementation of an implicit time update is, however, challenging and comes at a high computational cost. In a second order formulation, implicit formulations have been reported [8] and are shown to successfully capture low frequency plasma phenomena without resolving the high frequency scales. In the method of moments [15] charge and current densities are updated with separate approximate equations that follow from Taylor expansions. This economizes and slightly simplifies implicit implementation. The simplifications that one can make in these second order methodologies, however, don't extend to high-order formulation. Moreover, the low-order methods tend lead to geometric inflexibility, since they require an orthogonal spatial grid.

The hyperbolic cleaning does not require a full implicit formulation of both the particles and the field. Only the field solver introduces the stiffness. The stiffness of the particle solver is unchanged. The recently developed additive Runge-Kutta method in [12] is perfectly suited to tackle a partially stiff problem. The implicit-explicit (IMEX) Runge-Kutta solver can be implicit in physical and geometrical region of high stiffness, whereas the scheme is explicit in regions with low stiffness. In Kanevsky *et al.* [11] the method was applied for computations with geometric stiffness.

In this paper we apply the IMEX method to the particle-in-cell method. We solve the stiff field solver with hyperbolic cleaning implicitly, whereas particles are accurately updated explicitly. We report significant computational time savings in benchmark plasma problems, including the plasma wave and the Weibel instability, while properly cleaning divergence.

This paper is organized as follows: In the next section we briefly present the governing equations, including the hyperbolic cleaning equations. This is followed by a summary of the spatial discretization of the high-order PIC method based on DG. We then discuss time integration including explicit and IMEX schemes. The IMEX schemes are tested in the next section on a plasma wave and a Weibel instability. The final section is reserved for conclusions.

3

2 Governing Models

2.1 Electromagnetic Field Equations with Hyperbolic Cleaning

We consider the Maxwell's equations in TE form to describe the electromagnetic field:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{J}, \tag{1}$$

where

$$
\begin{aligned}
\mathbf{q} &= [E_x, E_y, B_z]^T, \\
\mathbf{F} &= \left[(0, -B_z, -E_y)^T, (B_z, 0, E_x)^T \right], \\
\mathbf{J} &= [J_x, J_y, 0]^T.
\end{aligned} \tag{2}
$$

Throughout $\mathbf{E}$, $\mathbf{B}$, and $\mathbf{J}$ represent the electric field, the magnetic field, and the current density, respectively, while the subscripts identify the direction of the vector field variable.

To satisfy divergence of $\mathbf{E}$ according to the Gauss law,

$$\nabla \cdot \mathbf{E} = \rho \tag{3}$$

we follow the hyperbolic cleaning approach described in [9; 13]. In this approach a correction potential is introduced into the Maxwell equations (1) as a Lagrangian multiplier. In the strictly hyperbolic formulation [13], the Maxwell's equations, (1-3), are altered to

$$
\begin{aligned}
\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \tilde{\mathbf{F}} &= \mathbf{J} \\
\frac{\partial \phi}{\partial t} &= \chi (\rho - \nabla \cdot \mathbf{E}) - \varepsilon \phi,
\end{aligned} \tag{4}
$$

with the modified fluxes

$$\tilde{\mathbf{F}} = \left[(\chi\phi, -B_z, -E_y)^T, (B_z, \chi\phi, E_x)^T \right]. \tag{5}$$

Here, ε is a damping constant. (4) is a strictly hyperbolic system of equations, with four characteristic velocities, $\lambda_{1,4} = \chi, -\chi$ and $\lambda_{2,3} = -c, c$, where $c = 1$ is the normalized speed of light. Compared to the uncorrected Maxwell equations, two characteristics, propagating information at speed, χ, are added, and one characteristic with zero velocity is omitted, hence eliminating the DC component of the system.

The χ characteristics effectively reduce the divergence error by propagating it away at velocity χ. Taking $\chi \gg c$ implies that the divergence error be swept out of the domain very rapidly, effectively imposing Gauss' law. As χ approaches infinity, one recovers (1)-(3).

2.2 Particle Equations

Charged particles are individually traced in time in the Lagrangian frame

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{v}_p, \tag{6}$$

$$\frac{dm\mathbf{v}_p}{dt} = q\left(\mathbf{E} + \mathbf{v}_p \times \mathbf{B}\right). \tag{7}$$

$\mathbf{x}_p$ and $m\mathbf{v}_p$ denote the non-dimensional particle coordinate and momentum, respectively, with q and m representing the particle charge and mass. For high-speed plasma the relativistic correction applies to m as $m = m_0/\sqrt{(1 - |\mathbf{v}_p|^2)}$ where m_0 is the mass at rest.

2.3 Coupling of field and particle equations

The field equations are directly coupled to the particle equations through $\mathbf{E}$ and $\mathbf{B}$ in (7). The particle equations are coupled to the field equations through the charge density, ρ and current density, $\mathbf{J}$, which are source terms in (4)

$$\rho(\mathbf{x}) = \sum_{i=1}^{N_p} \mathbf{K}(\mathbf{x}_p, \mathbf{x})q,$$

$$\mathbf{J}(\mathbf{x}) = \sum_{i=1}^{N_p} \mathbf{K}(\mathbf{x}_p, \mathbf{x})q\mathbf{v}_p, \tag{8}$$

where $\mathbf{K}(x, y) = \mathbf{K}(|x - y|)/V$ is a normalized weighting function that distributes the influence of each particle onto the electromagnetic field. N_p is the total number of particles in a finite volume V.

3 Spatial Discretization

The spatial discretization does not affect the temporal schemes we discuss in this paper. In principle, most spatial treatments can be implemented in-

dependently of the temporal discretization. Since the focus of this paper is on the temporal treatment, we only briefly summarize the spatial approximation of the coupled field and particle equations according to a particle-in-cell method based on a nodal discontinuous Galerkin (DG) discretization of the field equations (4). For a more detailed discussion, we refer to [9].

(1) and (4) are discretized with a high-order DG discretization on unstructured grids [7; 5; 9]. In the DG approach, we subdivide the computational domain into non-overlapping tetrahedral elements ensuring geometric flexibility. On each element, we assume that the local solution can be represented as an N'th order polynomial $\mathbf{q}_N(\mathbf{x}, t)$. The polynomial is based on accurate, well-conditioned nodes derived in [6].

To seek equations for the N local unknowns, we require the local approximate solution, $\mathbf{q}_N$,

$$\int_D \left(\frac{\partial \mathbf{q}_N}{\partial t} + \nabla \cdot \mathbf{F}_N \right) L_i(\vec{x}) d\vec{x} = \oint_{\partial D} L_i(\mathbf{x}) \hat{\vec{n}} \cdot [\mathbf{F}_N - \mathbf{F}^*] \, d\mathbf{x} + \mathbf{J}_N. \qquad (9)$$

Here, L_i signifies the typical Galerkin test function, $\mathbf{F}^*$ signifies a numerical flux and $\hat{\vec{n}}$ is an outward pointing unit vector defined at the boundary of the element. The role of the numerical flux is to connect the elements and ensure stability of the computational scheme. We use Lax-Friedrichs to determine the numerical flux. Within the multi-element context, the DG formulation is inherently discontinuous and yields a highly parallel local scheme. The scheme is exponentially convergent with increasing N.

With the operators, $\hat{\mathrm{M}}_{ij} = \int_D L_i L_j d\vec{x}$, $\quad \hat{\mathrm{S}}_{ij} = \int_D \nabla L_j L_i d\vec{x}$, $\quad \hat{\mathrm{F}}_{ij} = \oint_{\partial D} L_i L_j d\vec{x}$, we recover from (9) the fully explicit local scheme,

$$\hat{\mathrm{M}} \frac{d\hat{\mathbf{q}}}{dt} + \hat{\mathrm{S}} \cdot \hat{\mathbf{F}} = \hat{\mathrm{F}} \hat{\mathbf{n}} \cdot \left[\hat{\mathbf{F}} - \hat{\mathbf{F}}^* \right], \qquad (10)$$

where $\hat{\mathbf{q}}$ represents the $3N$-vector of nodal values, $\mathbf{q}_N$, at D. Similarly, $\hat{\mathbf{F}}^a$, $\hat{\mathbf{F}}^*$ denote nodal values for the flux and the numerical flux, respectively.

ρ_N and $\mathbf{J}_N$ are determined according to (8) by the cumulative projection of each particle onto the grid using the smooth weighting function

$$\mathbf{K}(x, y) = \frac{\alpha + 1}{\pi R^2} \left[1 - \left(\frac{|x - y|}{R} \right)^2 \right]^\alpha \qquad |x - y| = 0 \cdots R, \qquad (11)$$

where R is the radius of the spatial base of the function. α is a constant. With increasing α the weight function tends to a delta function.

The electric and magnetic field are determined at the particle position by interpolation using the element based interpolant $\mathbf{q}_N$.

4 Temporal Discretization

The semi-discrete initial value problem remains after spatial discretization

$$\frac{d\mathbf{U}}{dt} = \mathbf{S}(t, \mathbf{U}(t)), \qquad \mathbf{U}(t_0) = \mathbf{U}_0 \tag{12}$$

with

$$\mathbf{U} = \left[\mathbf{q}^T, \mathbf{x}_p^T, \mathbf{v}_p^T\right]^T \tag{13}$$

This systems of ODEs may be integrated in time with any scheme.

4.1 Explicit Runge-Kutta Time Integration

In our recent high-order PIC method [9], we update (12) with $2N$-storage, high-order, explicit Runge-Kutta (ERK) methods,

$$\left.\begin{aligned}
\mathbf{w}_i &= \alpha_i \mathbf{w}_{i-1} + \Delta t \mathbf{S}(t_{i-1}, \mathbf{U}^{(i-1)}), \\
\mathbf{U}^{(i)} &= \mathbf{U}^{(i-1)} + \beta_i \mathbf{w}_i,
\end{aligned}\right\} \quad i = 1, 2, ..., s, \tag{14}$$

where $\alpha_1 = 0$ for the algorithm to be self-starting, $\mathbf{U}^{(0)} = \mathbf{U}^{n-1}$, $\mathbf{U}^{(s)} = \mathbf{U}^n$, and $t_i = t^{n-1} + c_i \Delta t$. This is a $2N$ storage scheme, since only $\mathbf{U}$ and $\mathbf{w}$ require storage.

The five stage, $s=5$, scheme by Carpenter and Kennedy [3] was shown to conserve the particle momentum well with only a slight dissipation. The high-order nature of the ERK scheme and high-order discontinuous Galerkin space discretization leads to minimal dispersion. The ERK schemes are efficient and simply implemented.

However, the CFL stability criterion restricts the time step size according to

$$\Delta t <= \Delta t_{stab} = CFL \times \min_{\Omega}\left[\frac{\Delta x}{\lambda}\right] \tag{15}$$

7

where Δx represents a grid spacing and λ an eigenvalue matrix of $\mathbf{S}$ in (12). CFL depends on the stability regions of the RK method [11]. For problems that have high geometry or physics induced stiffness the time step may be very small leading to excessive computation in unsteady problems.

4.2   Implicit-Explicit Time Integration

When hyperbolic divergence cleaning is used the largest eigenvalue is determined as $\lambda = \chi$ which is typically at least a factor five times the speed of light. The time step required to update the field is then much smaller than the required time step for an accurate particle update.

We implement the additive Runge-Kutta scheme by Kennedy and Carpenter [12] to implicitly update the stiff field equations, while economically updating the particles in an explicit manner. The scheme can in principle be of any order, N. Here, we consider the six stage fourth order scheme.

The s stage, N additive Runge-Kutta method approximates (12) as

$$\mathbf{U}^{(i)} = \mathbf{U}^{(n)} + (\Delta t) \sum_{\nu=1}^{N} \sum_{\mathrm{j}=1}^{s} a_{ij}^{\nu} \mathbf{S}^{\nu} \left(t^{(n)} + c_j \Delta t, \mathbf{U}^{(j)} \right) \tag{16}$$

$$\mathbf{U}^{(n+1)} = \mathbf{U}^{(n)} + (\Delta t) \sum_{\nu=1}^{N} \sum_{\mathrm{j}=1}^{s} b_{i}^{\nu} \mathbf{S}^{\nu} \left(t^{(n)} + c_i \Delta t, \mathbf{U}^{(i)} \right) \tag{17}$$

where superscripts (n) identify a variable at $t^{(n)}$. Superscripts (i) indicate a variable at time $t^{(n)} + c_i \Delta t$ at stage i. Runge-Kutta methods are based on polynomial fits through intermediate nodes at $t^{(n)} + c_i \Delta t$. In the IMEX method an explicit and implicit integration is formulated at the same nodes c_i.

We split $\mathbf{U}$ in (13) into an explicit and implicit part as

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}_{[im]} \\ \mathbf{U}_{[ex]} \end{pmatrix}. \tag{18}$$

We take $\nu{=}1$ and 2 in (17) for the explicit and implicit integration, respectively.

The particles are integrated explicitly, $\mathbf{U}_{[ex]} = \left[\mathbf{x}_p^T, \mathbf{v}_p^T \right]^T$, with the explicit Runge-Kutta coefficients presented in Butcher tableaux format,

$$\begin{array}{c|c} c_i & a_{ij} \\ \hline & bi \end{array}$$

in Appendix A. The fluid $\mathbf{U}_{[im]} = \mathbf{q}$ is integrated implicitly with a singly-diagonally implicit Runge-Kutta solver. The coefficient are presented in a Butcher tableaux format in Appendix A.

The charge density, $\rho_N^{(i)}$, and current density, $\mathbf{J}_N^{(i)}$, at stage i are required for the implicit field update to stage i and are determined with the particle coordinate at stage i also. The particle trace therefore needs to be updated from stage $i-1$ to i explicitly before the field may be update implicitly.

With the source terms determined, it remains to update the approximated Maxwell's field equations with hyperbolic cleaning (9) implicitly,

$$\mathbf{U}_{[im]}^{(i)} = \mathbf{U}_{[im]}^{(n)} + \Delta t \sum_{j=1}^{s-1} a_{ij}^{(2)} \mathbf{S}\left(t^{(n)} + c_j \Delta t, \mathbf{U}^{(j)}\right)$$
$$a_{ii}^{(2)} \Delta t \mathbf{S}\left(t^{(n)} + c_i \Delta t\right). \tag{19}$$

In the singly-diagonally Runge-Kutta scheme $a_{ii}^{(2)}$ has the same value for all stages and $a_{ij}^{(2)} = 0 \quad \forall \quad j > i$. Solving for terms at (i) using (10), we can thus rewrite (19)

$$\mathbf{q}^{(i)} - \Delta t a_{ii}^{(2)} \left[\hat{\mathrm{M}^{-1}} \hat{\mathbf{S}} \cdot \mathbf{F}^{\hat{(i)}} + \hat{\mathrm{M}^{-1}} \hat{\mathbf{F}} \hat{\mathbf{n}} \cdot \left[\mathbf{F}^{\hat{(i)}} \right] - \hat{\mathbf{F}}^* \right]^{(i)} =$$
$$\Delta t \sum_{j=1}^{i-1} a_{ij}^{(2)} \mathbf{S}\left(t^{(n)} + c_j \Delta t, \mathbf{U}^{(j)}\right) \tag{20}$$

where the right hand side can be determined at each stage i. The fluxes in the hyperbolic cleaning equations, $\mathbf{F}$, contain only linear operators on $\mathbf{q}$ and $\mathbf{F}^{(i)}$ and $\mathbf{q}^{(i)}$ are linearly related. So, (20) represents a linear system of equations

$$A\mathbf{q}^{(i)} = \mathbf{b}. \tag{21}$$

In DG approximations A is a very sparse matrix. Fast direct solvers are available to determine its inverse [4]. Since a_{ii} is the same for all stages $i = 1, .., 6$, the inverse can be stored and reused throughout the time integration. For large systems standard iterative solvers can be implemented.

5 Tests

In this section we will illustrate the performance of the IMEX method in simulations of a one-dimensional plasma wave and a two dimensional Weibel instability.

5.1 Plasma Wave

We consider the simulation of a plasma wave using a quasi-electrostatic field solver based on the hyperbolic cleaning. Neglecting the magnetic field, we consider the hyperbolic field equations that govern the electric field:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{S}, \tag{22}$$

with

$$
\begin{aligned}
\mathbf{q} &= [E_x, E_y, \phi]^T, \\
\mathbf{F} &= \left[(\chi\phi, 0, -\chi E_x)^T, (0, \chi\phi, -\chi E_y)^T \right], \\
\mathbf{S} &= [0, 0, \chi\rho]^T.
\end{aligned}
\tag{23}
$$

In the limit that $\chi \to \infty$ these equations reduce to the Gauss law used for conventional electrostatic field solvers. The particles are updated with

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{v}_p, \tag{24}$$

$$\frac{d\mathbf{v}_p}{dt} = q/m\mathbf{E} \tag{25}$$

We follow the setup of the problem described in [9]. We set $\chi = 10$, which was shown in [9] to give an accurate, matching result with published data [1].

The stable time step for the explicit field solver is determined with (15) with the largest eigenvalue $|\lambda| = \chi$, Δx the minimum grid spacing. CFL is now dependent on the stability regions of the Runge-Kutta methods as discussed in [11]. In the explicit Runge-Kutta (ERK) method in (14) $CFL_{max} \sim 2.8$. The explicit part of the implicit-explicit method (IMEX-ERK) has a comparable stability region to the ERK. The implicit part of the IMEX is L-stable, and has no time step restriction.

We first establish that the various schemes achieve their formal fourth order convergence rate. Since no analytical plasma results are available, we study convergence by comparing results computed with $CFL > 0.25$ to a result computed with $CFL = 0.25$ at time $t = 0.5$ in the development of the plasma wave.

Figure 1 shows the L_2 error in the E_x field so determined. Both ERK and IMEX are fourth order accurate conform their formal convergence rate. Naturally, the ERK results cannot be obtained beyond $CFL > 3$. IMEX is stable at $CFL = 256$. The error stops following the formal rate for $CFL > 64$, when the time step is on the order of the time period of the plasma wave and the time step is too large to capture the smallest plasma time scale.
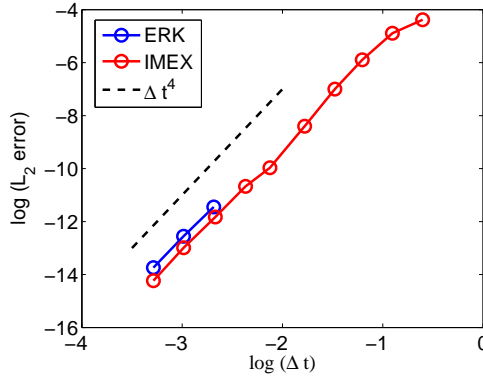

Fig. 1. L_2 error plotted versus time step for a plasma wave computation with an explicit and IMEX hyperbolic cleaning solver.

In a plasma wave the field potential energy, $E_x^2 + E_y^2$ is exchanged with kinetic energy of the particles (Fig. 2a). The divergence, which should be zero according to the Gauss law, also shows a periodic trend in Fig. 2b. The solution is resolved with an acceptable level of divergence at χ=10 [9] for $CFL < 64$. At $CFL = 128$, the solution is eyeball inaccurate. At $CFL = 64$ only two time steps are taken to resolve one time period of the plasma wave. With six stages per Runge-Kutta step this gives around 12 time nodes to resolve one sinus plasma wave in time. This is consistent with a lower resolution requirement for higher order methods.

We computed the plasma wave in Matlab and used a built-in conjugate gradient iterative solver to solve the linear system in (21). The IMEX solver was about thirty percent slower than the ERK solver. The IMEX scheme allowed for a time step that was twenty times larger, while the results are accurate. IMEX thus significantly improves the computation efficiency.
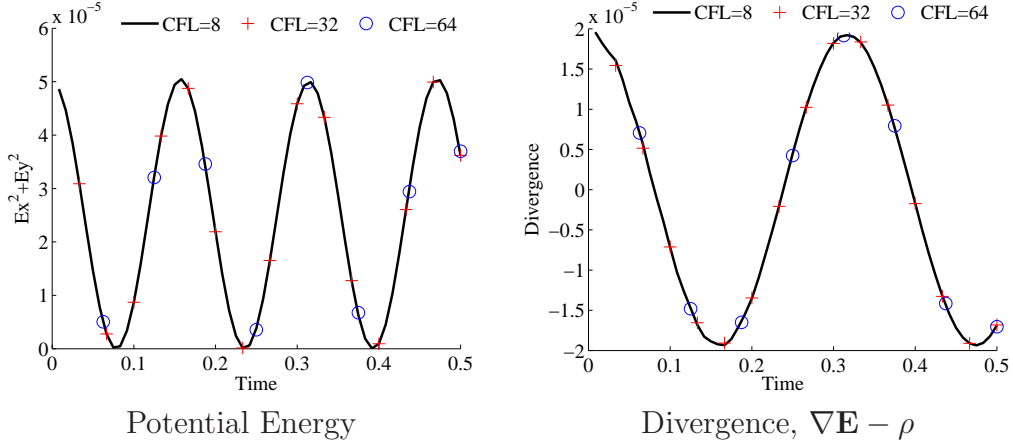
Potential Energy

Divergence, $\nabla \mathbf{E} - \rho$

Fig. 2. Potential energy and divergence trends for a plasma wave computation with IMEX hyperbolic cleaning solver at various CFL.

5.2   Weibel Instability

We repeat the simulation of the Weibel instability as outlined in [9]. The governing system of equations is described in section 2. The Weibel instability simulations are performed on a unit square with periodic boundary conditions. We consider a quasi-neutral plasma with a thermal velocity ratio of 5 of the velocity in x, $u_{the} = 0.25$, and y, $v_{the} = 0.05$, direction. The plasma frequency is fifteen times the length of the square, i.e., $\omega_{pe} = 15$ resulting in $\frac{q}{m} = -(15\pi)^2$ with the electron charge density set to $\rho = -1$. We track 300×300 particles in this domain for two time units. We performed computations with a code programmed in the C-language on a Beowulf cluster with four Intel Xeon 5110 processors per node at 1.60GHz.

With the IMEX solver we can perform affordable simulations of this flow with much higher values of χ than reported in [9; 10]. At higher χ the hyperbolic cleaning equations in (4) are more accurately representing the system of Maxwell's equations plus Gauss law that govern the plasma physics. However, when increasing χ the total energy exhibits a larger increase in time (Fig. 3). This indicates a less accurate capturing of the physical conservation of energy.

The total energy increase is caused by increases in the kinetic energy and electric energy trend. With increasing χ the electric field responds faster to changes and in the charge density, ρ. The noise levels in the charge density field that are typically high in PIC simulations (Fig. 4a), thus yield an increase in the noise levels in the electric energy with the faster response of the hyperbolic cleaner to changes in ρ. The increased electric energy is via the Lorenz force also directly responsible for faster moving particles and hence larger kinetic energy of the plasma. The increased electric and kinetic energy lead to an unphysical increase in total energy. The magnetic field energy appears unaffected by changes in χ, consistent with the hyperbolic formulation that doesn't affect

12

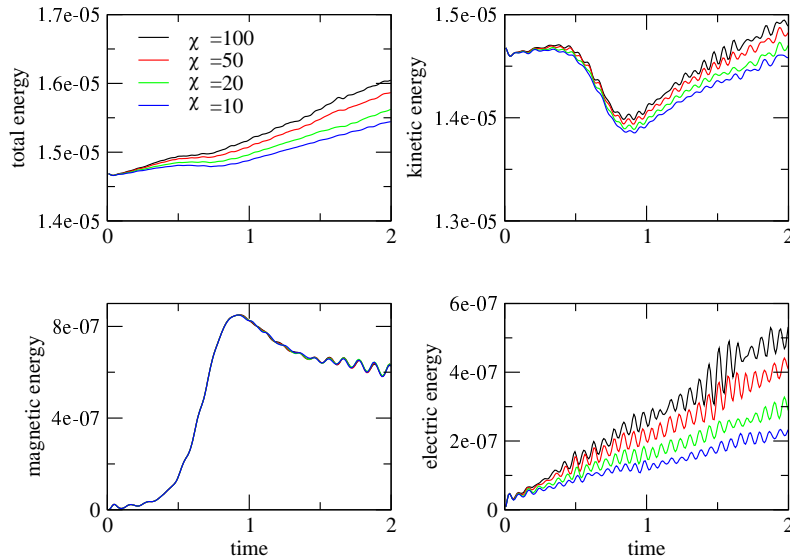the equation for $\mathbf{q}(3) = H_z$ in the original Maxwell's equations.

Fig. 3. Comparison of various plasma energy components for computations of the Weibel instability with hyperbolic divergence cleaning with χ=10,20,50 and 100 and an IMEX time integration.

The divergence cleaning performance is comparable for all χ considered here. The noise in ρ typical for PIC methods (Fig. 4a) is mostly responsible for the divergence magnitude. The magnitude of χ only slightly reduces the divergence (Fig. 4b). With the increases in noise in the energy trends and the lack of divergence cleaning improvement, it would appear unwise to increase χ beyond a value of ten.

For a stable *explicit* time integration, the time damping constant must be $\varepsilon \leq \Delta t_{stab}$. The implicit time integration has no stability restrictions and hence the damping constant may be taken any value. With a reduced damping the fluctuations in the correction potential, ϕ, are smaller. The smaller fluctuations are translated to the electric field as seen by the reduced magnitude of the electric energy and the reduced amplitude of the fluctuations in the electric energy trend (Fig. 5d). The decreases in the electric energy trend lead to a decrease in kinetic energy and total energy whereas the magnetic energy trend is unaffected. When no damping is applied, i.e. ε=0, the solution only slightly deviates from the solution with a damping of $\varepsilon = \Delta t/4$.
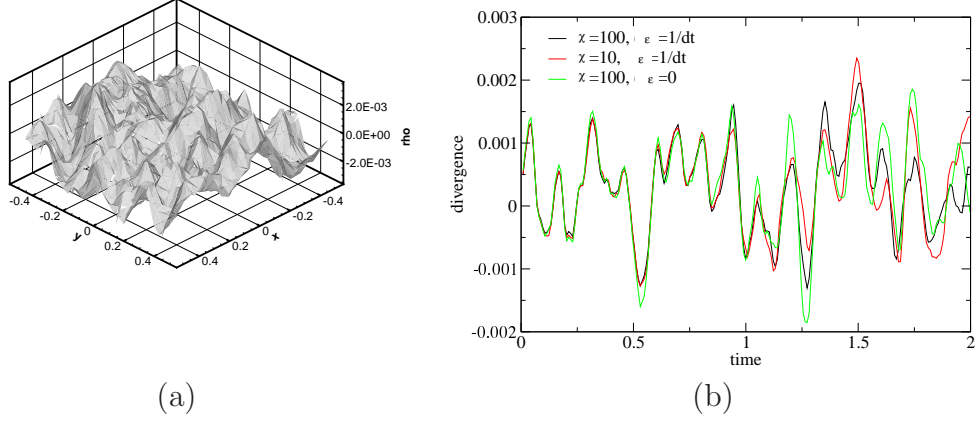
<center>(a)</center>

<center>(b)</center>

Fig. 4. (a) Surface plot of the instantaneous charge density, ρ. (b) Comparison of divergence for computations of the Weibel instability with hyperbolic divergence cleaning with χ=10,20,50 and 100 and an IMEX time integration.
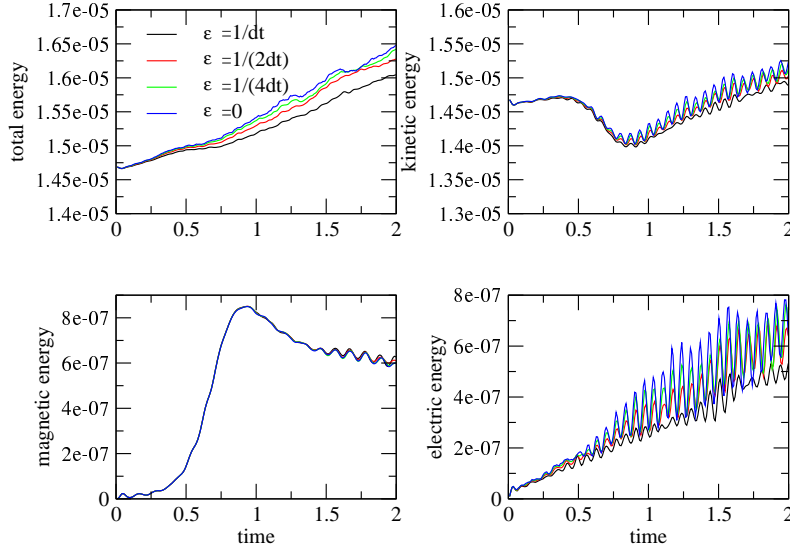

Fig. 5. Comparison of various plasma energy components for computations of the Weibel instability with hyperbolic divergence cleaning with $\varepsilon = 1/\Delta t$, $0.5/\Delta t$, $0.25/\Delta t$, and 0 and an IMEX time integration.

For the above cases, we have taken the time step smaller than the stable time step, Δt_{stab}, that would be required for time integration of the Maxwell equations without hyperbolic cleaning in (1). This "stable" time step is determined with (15) taking $\lambda = c$, where c is the speed of light. Since the implicit hyperbolic cleaner does not have a stability restriction on the time step, we can take

<center>14</center>

$\Delta t > \Delta t_{stab}$. The Weibel instability is accurately computed for $\Delta t = 2\Delta t_{stab}$ (Fig. 6). With the increased Δt the total energy trend shows a slight increase, which is attributed to the finite grid instability. To prevent grid significant grid heating in PIC the time step has to satisfy $\Delta t < 2/\omega_{pe} = 2/15$. When $\Delta t > 4\Delta t_{stab}$, this stability criterion is violated, and the PIC simulation is unstable.
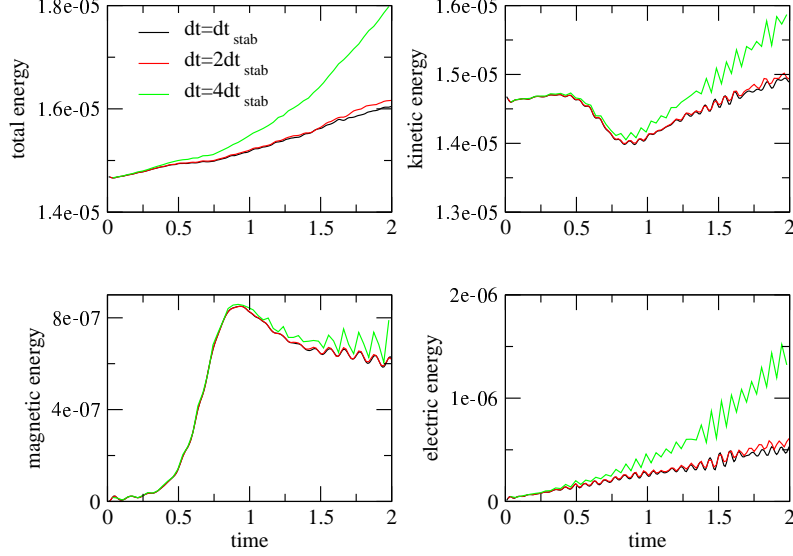

Fig. 6. Comparison of various plasma energy components for computations of the Weibel instability with hyperbolic divergence cleaning with $\Delta t = \Delta t_{stab}$, $2\Delta t_{stab}$, and $4\Delta t_{stab}$, and an IMEX time integration.

If a direct solver is used to solve (21), the implicit time integration of the continuum field equations is a factor 14 slower than the explicit time integration. The particle update, however, is far more computational expensive than the field update and hence the computational cost of particle-in-cell simulations are dominated by the particle solver. The Weibel instability, for example, is computed with 880 time steps per wall clock hour, whereas the computational speed of the pure implicit electromagnetic field with the same grid is 14,500 time steps per hour. Since the computational efficiency of the explicit and IMEX *particle* time integration is comparable, the penalty of the implicit field solver is negligible as compared to the explicit particle-in-cell solver. The alleviation of the time step by the IMEX method thus directly translates to improved computational efficiency of the PIC with IMEX compared to explicit PIC.

6 Conclusions

We have implemented a high-order implicit-explicit Runge-Kutta time integration scheme for simulations of plasmas with particle-in-cell methods that use hyperbolic cleaning methods to satisfy Gauss law. An implicit solver updates the stiff field equations, while an explicit solver updates the particle equations.

The implicit-explicit method improves upon computational efficiency when the hyperbolic cleaner is used. The governing hyperbolic cleaning model more accurately represents the governing, physical electromagnetic model, when cleaning along characteristics with increasing speeds larger than the speed of light. The implicit-explicit method firstly alleviates the increasingly restrictive time step of the hyperbolic cleaning method with increasing speed. Secondly, the non-stiff particle model is at the same time inexpensively updated in an explicit manner.

The penalty in computational time of the implicit-explicit particle-in-cell solver as compared to the explicit solver is minimal for the linear electromagnetic equations. The discretization of the linear electromagnetic equations yields a cheaply directly invertible system of equations. Moreover, the single diagonally Runge-Kutta method reuses the same inverse of the linear system of equations that can thus be stored in memory and reused. Sparse direct solvers solve the system at low cost. Moreover, the particle time integration is more computationally expensive then the field solve and is mostly determining the computational efficiency of the algorithm. Since the computational cost of the explicit particle update in explicit and IMEX particle-in-cell is comparable, the alleviation in time step by the implicit time integration thus translates nearly completely to computation efficiency improvement.

With an implicit-explicit time integration, the time step may be increased beyond the stable time step required for pure electromagnetic solver (without hyperbolic cleaning) leading to a true implicit solver. The time step in PIC is then restricted by the finite grid instability as $\Delta t < 2/\omega_{pe}$, where ω_{pe} is the plasma frequency.

We have shown that the implicit-explicit particle-in-cell method based on a discontinuous Galerkin discretization in space has formal fourth order accuracy in time. The fourth order scheme only requires two time steps to resolve one period of a plasma wave improving over the time resolution typically required for low order methods.

We suggest that the speed of the characteristic, χ, along which divergence is cleaned should be optimally taken at approximately $\chi = 10$. For $\chi < 10$ it was shown that the governing equations are inaccurately represented by the

hyperbolic cleaner [9]. On the other hand, with increasing χ, the the charge density, ρ is more tightly coupled to the electric field and numerical noise in ρ is increasingly passed onto the electric energy and kinetic energy. As a result the total energy was found to unphysically increase for larger χ. Ergo, taking $\chi > 10$ is not advisable based on the tests performed in this paper.

A damping on the divergence, that is convected away by the hyperbolic cleaning method, reduces numerical noise in the electric field and improves the accuracy of total energy conservation.

We conclude finally that the implicit-explicit method should be generally applicable to particle-mesh methods that have a discrepancy in stiffness between the continuum and particle phase.

References

[1]   C. K. Birdsall and A. B. Langdon. *Plasma physics via computer simulation.* McGraw-Hill, Inc., 1985.

[2]   J. D. Buckmaster and G. S. S. Ludford. *Theory of Laminar Flames.* Cambridge University Press, Cambridge, U.K., 1982. Also, Lectures on Mathematical Combustion, SIAM, Philadelphia, PA., 1983.

[3]   M. H. Carpenter and C. A. Kennedy. A fourth-order 2N-storage Runge-Kutta scheme. NASA TM 109112, June 1994.

[4]   T. Davis. UMFPACK. available at http://www.cise.ufl.edu/research/sparse/umfpack, University of Florida, FL.

[5]   J. Hesthaven and T. Warburton. Nodal high-order methods on unstructured grids. i. time-domain solution of Maxwell's equations. *J. Comp. Phys.*, 181:186–221, 2002.

[6]   J. S. Hesthaven. From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex. *SIAM J. Num. Anal.*, 35:655–676, 1998.

[7]   J. S. Hesthaven and D. Gottlieb. Stable spectral methods for conservation laws on triangles with unstructured grids. *Comput. Methods Appl. Mech. Engr.*, 175:361–381, 1999.

[8]   D. W. Hewett and B. Langdon. Electromagnetic direct implicit plasma simulation. *J. Comp. Phys.*, 72:121–155, 1987.

[9]   G. B. Jacobs and J. S. Hesthaven. High-order nodal discontinuous Galerkin particle-in-cell method on unstructured grids. *J. Comp. Phys.*, 214:96–121, 2006.

[10] G.B. Jacobs, G. Lapenta, and J.S. Hesthaven. Simulations of plasmas with a high-order discontinuous Galerkin particle-in-cell solver. AIAA Paper 2006-1171, 2006.

[11] A. Kanevsky, M. H. Carpeneter, D. Gottlieb, and J. S. Hesthaven. Ap-

plication of implicit-explicit high order Runge-Kutta methods to discontinous Galerkin schemes. *J. Comp. Phys.*, 225:1753–1781, 2007.

[12] C.A. Kennedy and M.H. Carpenter. Additive Runge-Kutta schemes for convection-diffusion-reaction equations. *Appl. Num. Math.*, 44:129–181, 2003.

[13] C. D. Munz, P. Omnes, R. Schneider, E. Sonnendrukker, and U. Voss. Divergence correction techniques for Maxwell solvers based on a hyperbolic model. *J. Comp. Phys.*, 161:484–511, 2000.

[14] J. Villasenor and O. Buneman. Rigorous charge conservation for local electromagnetic field solvers. *Comput. Phys. Comm.*, 69:306–316, 1992.

[15] H. X. Vu and J. U. Brackbill. Celeste1d: An implicit, fully kinetic model for low-frequency, electromagnetic plasma simulation. *Comput. Phys. Comm.*, 69:253, 1992.

Table I tableau:

0	0	0	0	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	0	0	0
$\frac{83}{250}$	$\frac{13861}{62500}$	$\frac{6889}{62500}$	0	0	0	0
$\frac{31}{50}$	$\frac{-116923316275}{2393684061468}$	$\frac{-2731218467317}{15368042101831}$	$\frac{9408046702089}{11113171139209}$	0	0	0
$\frac{17}{20}$	$\frac{-451086348788}{2902428689909}$	$\frac{-2682348792572}{7519795681897}$	$\frac{12662868775082}{11960479115383}$	$\frac{3355817975965}{11060851509271}$	0	0
1	$\frac{647845179188}{3216320057751}$	$\frac{73281519250}{8382639484533}$	$\frac{552539513391}{3454668386233}$	$\frac{3354512671639}{8306763924573}$	$\frac{4040}{17871}$	0
	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$\frac{-2260}{8211}$	$\frac{1}{4}$

Table I
The explicit Runge-Kutta coefficients of the additive Runge-Kutta scheme presented in a Butcher tableau.

Table II tableau:

0	0	0	0	0	0	0
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	0	0	0	0
$\frac{83}{250}$	$\frac{8611}{62500}$	$\frac{-1743}{31250}$	$\frac{1}{4}$	0	0	0
$\frac{31}{50}$	$\frac{5012029}{34652500}$	$\frac{-654441}{2922500}$	$\frac{174375}{388108}$	$\frac{1}{4}$	0	0
$\frac{17}{20}$	$\frac{15267082809}{155376265600}$	$\frac{-71443401}{120774400}$	$\frac{730878875}{902184768}$	$\frac{2285395}{8070912}$	$\frac{1}{4}$	0
1	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$\frac{-2260}{8211}$	$\frac{1}{4}$
	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$\frac{-2260}{8211}$	$\frac{1}{4}$

Table II
The single diagonal implicit Runge-Kutta coefficients of the additive Runge-Kutta scheme presented in a Butcher tableau.