JOURNAL OF COMPUTATIONAL PHYSICS **145,** 41–60 (1998)
ARTICLE NO. CP986002

Long Time-Step Particle Pushing in Drift Approximation without Orbit Averaging

Michael V. Smolsky

Department of Condensed Matter Physics, Weizmann Institute of Science, Rehovot, 76100, Israel
E-mail: FNSiguc@Weizmann.Weizmann.Ac.IL

Received September 30, 1997; revised February 24, 1998

An alternative way to utilize the well known drift approximation for particle motion in electromagnetic fields is proposed. Contrary to the traditional approach, in which the motion of the guiding center of a particle is considered, in the suggested algorithm the coordinates and velocities of the particle itself are evaluated. This approach is found to give accurate results provided that the characteristic scales of the change of fields both in time and in space are large compared to the corresponding micro-scales of the particle motion in these fields (e.g., wavelength of radio wave is large compared to the particle gyroradius in the magnetic field of that wave). Under this condition an approximate analytical solution for the Newton–Lorentz Law, accurate within many characteristic micro-timescales of the particle motion, is derived. This approximation is exploited to advance a particle substantially within a single elementary step of the algorithm, which can extend for as long as $\sim m_i/m_e \simeq 2000$ times the traditional one for certain astrophysical plasmas. Utilization of this approach can give a boost of productivity for the existing and new PIC codes, in which substantial computational time is spent solving the Newton–Lorentz Law for superparticles. In the present work the approach is implemented for the ultrarelativistic case, with the magnetic field prevailing over, although being not necessarily much higher than the electric field. The relevant computer code is developed and used to simulate the motion of a particle in an electromagnetic field with a complicated profile. The results exhibit a good agreement with those, obtained by direct integration of the Newton–Lorentz law, using a conventional ODE solver.

I. INTRODUCTION

Plasma is a mixture of charged particles of at least two different species. The masses of different particles are usually very much (1836 times and more) different. Once the mass of a particle enters the equations of motion for it, the orbits of the particles of different species may have very different time and space scales. In the case of the thermolized non-relativistic

0021-9991/98 $25.00

plasma in a magnetic field, a configuration typical for laboratory plasmas, a characteristic space scale for a particle motion is its gyroradius, which is proportional to $\sqrt{m}$, where m is the mass of the particle. In a super-relativistic case the particle gyroradius and gyroperiod are proportional to its energy, and in a heavily non-thermal configuration, like in the problem of interaction of cold relativistic plasma beam with external magnetic barrier they differ by a factor of m_i/m_e [1]. Such a difference of spatial and temporal scales presents a great problem for numerical simulations of plasma phenomena. Not only the accuracy is lost if the particle motion is integrated with a too large time-step, but also numerical instabilities can arise [2, 3].

It is also often the case that the scales of the self-consistent fields are governed by dynamics of the heavier particles. In collisionless astrophysical shocks, for example, the width of the shock is usually the *proton* gyroradius, or larger [4]. External fields, which dominate in some phenomena, may also be large-scale. In such configurations, the relative field variation on the scale of the electron gyromotion can be quite small.

A pushing scheme, which is substantially more economical than traditional short time-step pushers (see, e.g., [5]) under the mentioned conditions, is suggested in the present paper. The time interval of a single push can last for many electron time-scales, and the details of the orbit of an electron being pushed are not lost. The paper deals exclusively with the new pushing method, while the incorporation of the novel technique into a self-consistent plasma simulation package is the course of the ongoing work.

Configurations, in which plasma bulk is ultrarelativistic and fields are weakly non-homogeneous on electron micro-scales, have recently attracted a great attention of members of the astrophysical community, due to the availability of new data obtained from γ-telescopes, like BATSE on board the Compton Gamma Ray Observatory and measurements of super-high-energy cosmic rays. Both such photons and cosmic rays are believed to originate from huge regions, in which particles are accelerated in the process of their interaction with large-scale strong electromagnetic fields. Physical properties of such structures are so peculiar, that their theoretical models can usually not be verified without numerical simulations. Kinetic effects and instabilities play an essential role in those phenomena, which eliminates the possibility of hydrodynamic or hybrid treatment. Present-day successes of truly kinetic study of such configurations are limited to either an incorrect mass ratio (see below) or short-scale 1D runs [1, 6]. The experience accumulated by these authors suggests that the traditional PIC approach could hardly be adequate for so complicated and unusual plasmas. A large pool of competing theoretical models expressed in hundreds of papers published during the last several years exists, which can be neither tested nor refined due to the lack of appropriate numerical algorithms. The present paper is an attempt to construct such an algorithm.

Several methods were proposed since the first PIC simulations to overcome the mentioned difficulties. One of the first ideas was to modify (decrease) the ion-to-electron mass ratio artificially, so that the scales were not separated that much. The effect of incorrect mass ratio on the results of simulations was studied numerically [7, 8] (in the paper [8] the mass ratio $m_i/m_e = 64$ was found to be high enough), but it is not always known *a priori* whether a simulation with a modified mass ratio would lead to qualitatively correct conclusions for a given configuration [1], and full-scale kinetic simulation with the right ratio may be too expensive to be carried out [9].

In *hybrid* treatment [10–16] the idea of simulating all the light particles one-by-one was given up, and electrons were treated as a fluid. Certain kinetic aspects of the phenomena

were clearly lost in such an approach [17, 18]. In another modification of the hybrid method, invented to account for kinetic effects due to super-thermal particles, energetic electrons were treated as particles, and thermal electrons as a fluid [19, 20].

The time scales for particle and field advance were separated in *orbit-averaged* algorithms in such a manner that kinetic effects were not lost [17, 21]. Particles were advanced in a sequence of sufficiently small time steps, so their trajectories were properly resolved, current and charge densities were accumulated over a large number of such "micro" time steps, and field equations were solved once in a much larger "macro" time step. The effects of field variation on particle trajectories were not lost. We emphasize that in that technique a particle micro time-step should be small enough, so that a version of a traditional ODE solver (e.g., Boris push [5]) could be used. In the *direct implicit* method [22–24] particle coordinates were evaluated on the basis of the values of electromagnetic fields taken from the next time step and the predictor iteration for the fields was constructed. In *implicit moment* method [25–28] fluid equations were used to predict future fields. An orbit-averaged algorithm was successfully merged with implicit models [29]. The primary goal of orbit averaged and implicit codes was to overcome the time-step constraint, imposed by numerical stability requirements, in the cases when accuracy considerations did not require that small a time step. This makes integration of the Newton–Lorentz law for particles with a large time-step possible, at the expense of filtering out all the high-frequency phenomena from the results of simulations.

There exists a theoretical approach to problems of particle motion in electromagnetic fields, known as *drift approximation*, which dates back to Alfvén [30]. In that approach the particle motion was presented as a superposition of "fast" oscillations—gyrorotations around a magnetic field line and a slow "drift"—a displacement of the guiding center of the particle orbit. The fast oscillations may be theoretically filtered out of the particle equations of motion, and the resulting drift equations solved. Numerical implementation of the drift approximation is known as a *gyrokinetic* method [31, 32]. This method was merged with most of the other techniques, e.g., [33, 34].

In yet another method, which might be called *semi-analytical*, particle trajectories were calculated analytically in some approximation, and the obtained expressions were used to advance particles over a large time step [35]. This approach was successfully applied to the problem of particle-wave interaction. It was also merged with the hybrid approximation, so that the resonant particles interacted with the wave in a kinetic manner, while the bulk of plasma was described as a fluid, with the kinetic effects included in the source terms of the fluid equations.

The presented algorithm is also semi-analytical, which makes it methodologically similar to [35]. The coordinates and velocity of a particle itself, but *not* of its guiding center, are calculated analytically, under the assumptions of the drift approximation. The resulting formulae are then implemented numerically. The total computational time required for a single push of a particle is independent of the number of its gyrations around the guiding center, which makes use of the proposed pushing technique in kinetic simulations of heavy-ion or super-relativistic plasmas with realistic ion to electron mass ratios possible. The presented derivations and simulations are performed for relativistic particles, and their direct application to the non-relativistic case can lead to substantial inaccuracy. We studied in detail the "gyro" case, i.e., the configuration of the fields with $|B| > |E|$, leaving the opposite possibility, if the approach may be effectively extended for it at all, for future study.

A simplified version of this algorithm, the so-called *cycloid approximation*, was implemented long ago [36], but the fields were kept constant during the particle pushing time step, and as a result a particle could be pushed by no more than $\frac{1}{8}-\frac{1}{5}$ of a cyclotron period in a time step with a reasonable accuracy. See also [37] for discussions of the cycloid approximation from the point of view of time-reversibility. Similar ideas were implemented in cylindrical coordinates in [38] for a computational study of the negative mass instability. Particle gyroradius was not small, compared to the scale of the system in that configuration, but the fields were almost constant along particle trajectory and a slow change of the gyroradius in time was computed. It was noted several times (e.g., [39]), that the cycloid approximation works well only if the time step is small enough, so that the fields can be assumed constant within the time step. Our ideas may be viewed as a next step in the cycloid method, when the change of the fields along the particle path is accounted for more accurately. As we will see, it allows us to extend the integration of particle equations of motion to very long time steps.

Runge–Kutta style correction is proposed to reduce by purely finite-difference techniques the error caused by certain analytical assumptions. The second-order scheme was implemented and found effective in terms of accuracy of algorithm versus computational time competition.

Since in this work we present a particle pushing algorithm only, we leave the discussion of a suitable field solver out of consideration. We address neither numerical stability, nor effectiveness of any code, based on the proposed pusher. We also do not present any self-consistent plasma simulation, where the new pushing technique is used. Some analyses [23] hint that overall code performance may be substantially degraded by inaccurate sampling of field profiles by particles. That provides a certain enthusiasm about our pushing scheme, in which fields are sampled very accurately, provided they are smooth enough.

The material is presented as follows. We first (in Section II) formulate the problem in terms of particle equations of motion, and then (Section III) briefly review a fully integrable case, when the fields do not depend on coordinates of a particle. Next (Section IV), we obtain the main formulae of our drift approximation and introduce a finite-difference correction to these analytical formulae. There we also discuss the accuracy of the results obtained. We apply the proposed algorithm to a number of field configurations and discuss the results of the tests in Section V. We briefly outline our main conclusions of the paper in Section VI.

II. FORMULATION OF THE PROBLEM

In this work we will consider the numerical solution of the Newton–Lorentz law for the motion of a relativistic particle in electromagnetic fields. The algorithm is developed for a single particle in $(2+1)$D, meaning that 2 space dimensions (x and y) are taken into account together with the time (t) coordinate. The three quantities compose a vector $\mathbf{x} = \{ct, x, y\}$ in Minkovsky space, which we will refer to as a coordinate of a particle hereafter. Here and below c denotes the speed of light in vacuum. The fields involved include the x and y components of the electric field, and the z component of the magnetic field. Other components of electromagnetic fields cannot enter a $(2+1)$D problem in principle. The Newton–Lorentz law for the particle may be written in the covariant form [40],

$$\frac{d\mathbf{u}}{ds} = \frac{e}{mc^2}\hat{F}(\mathbf{x}(s))\mathbf{u}, \tag{1}$$

where

$$\hat{F} = \begin{bmatrix} 0 & E_x(\mathbf{x}) & E_y(\mathbf{x}) \\ E_x(\mathbf{x}) & 0 & B(\mathbf{x}) \\ E_y(\mathbf{x}) & -B(\mathbf{x}) & 0 \end{bmatrix} \tag{2}$$

is the electromagnetic field tensor, e and m are the charge and mass of the particle, and s is the interval, which plays the role of an independent variable. The signature of the metric tensor used is $(+--)$.

The velocity vector satisfies the condition [40]

$$|\mathbf{u}| = 1, \tag{3}$$

if the norm is understood in the Minkovsky sense, in accordance with our choice of the metric tensor,

$$\mathbf{a}\cdot\mathbf{b} \equiv a^t b^t - a^x b^x - a^y b^y, \qquad |\mathbf{a}| \equiv \mathbf{a}\cdot\mathbf{a}. \tag{4}$$

We are going to use the above convention for the scalar product throughout this paper. It could be easily checked that $\mathbf{u}\cdot\hat{F}\mathbf{u} = 0$ for any $\mathbf{u}$, and thus if the velocity of a particle is initially chosen to satisfy (3), it would follow this normalization rule forever.

Particle coordinates and velocity are connected through the equations

$$\frac{d\mathbf{x}}{ds} = \mathbf{u}(s). \tag{5}$$

The initial conditions for these equations are

$$\mathbf{u}(s=0) = \mathbf{u}_s, \qquad \mathbf{x}(s=0) = \mathbf{0}. \tag{6}$$

The major assumption of all the future work is that all the functions $E_x(\mathbf{x})$, $E_y(\mathbf{x})$, $B(\mathbf{x})$ change only slightly and smoothly on the time and space scales of the orbit of the particle (i.e., gyroperiod and gyroradius in the case $|E| < |B|$ and the typical energy raise time and scales in the opposite case). We will derive some general formulae for an arbitrary relation between $|E|$ and $|B|$, but we will later confine ourselves with the case $|E| < |B|$, but not necessarily $|E| \ll |B|$.

III. MOTION OF A RELATIVISTIC PARTICLE IN STATIONARY HOMOGENEOUS FIELDS

We first review the simplest, fully integrable case, in which the fields do not depend on $\mathbf{x} \equiv \{ct, x, y\}$ coordinates and thus solve the "unperturbed" problem. The general solution of the system of linear ordinary differential equations (1) for the velocity is

$$\mathbf{u}_L(\phi) = C_0\mathbf{u}_0 + C_+\mathbf{u}_+ e^{+\phi} + C_+\mathbf{u}_- e^{-\phi}, \tag{7}$$

where we switched to a new dimensionless independent variable $\phi \equiv (e/mc^2)\sqrt{E^2 - B^2}s$, later referred to as "phase" instead of s and the eigenvectors of the matrix $\hat{F}$ are

$$\mathbf{u}_0 = \begin{bmatrix} B \\ E_y \\ -E_x \end{bmatrix}, \qquad \mathbf{u}_\pm = \begin{bmatrix} E \\ B\frac{E_y}{E} \pm \lambda\frac{E_x}{E} \\ -B\frac{E_x}{E} \pm \lambda\frac{E_y}{E} \end{bmatrix}, \tag{8}$$

$E^2 \equiv E_x^2 + E_y^2$ as well as $\mu \equiv \sqrt{E^2 - B^2}$ are introduced for convenience. We notice that μ is a Lorentz-invariant quantity, i.e., it does not change under the Lorentz transformation of electromagnetic fields [40]. The full set of eigenvalues of the matrix $\hat{F}$ is $\{0, \pm\mu\}$. The above normalization of the eigenvectors has the property that in the important cases $|E| \ll |B|$ as well as $|B| \ll |E|$ both these vectors and the coefficients C_0, C_+, C_- remain finite. In the case $|E| \geq |B|$ μ and together with it all the quantities entering these formulae are real, while in the opposite case $|E| < |B|$ only the real part of the complex functions (7) are to be considered. We switch to another form of writing the solution (7), which is purely real:

$$\mathbf{u}_L(\phi) = C_0\mathbf{u}_0 + C(\mathbf{u}_e\, f_e(\phi) \pm \mathbf{u}_o\, f_o(\phi)). \tag{9}$$

Here and below upper signs correspond to the case $|E| > |B|$, and the lower ones to the opposite case $|E| < |B|$. The special case $\mu = 0$, i.e., $|E| = |B|$ is not covered here, for the reasons made clear below, but may be found in [40]. "Even" $f_e(\phi)$ and "odd" $f_o(\phi)$ functions, which enter these expressions are

$$\begin{aligned} f_e(\phi) &= \cosh(\phi), \qquad f_o(\phi) = \sinh(\phi); \qquad |E| > |B|; \\ f_e(\phi) &= \cos(\phi), \qquad f_o(\phi) = \sin(\phi); \qquad |E| < |B|, \end{aligned} \tag{10}$$

and $\{\mathbf{u}_0, \mathbf{u}_e, \mathbf{u}_o\}$ form a new set of vectors with

$$\mathbf{u}_0 = \frac{1}{\lambda}\begin{bmatrix} B \\ E_y \\ -E_x \end{bmatrix}, \qquad \mathbf{u}_e = \frac{1}{\lambda}\begin{bmatrix} E \\ B\frac{E_y}{E} \\ -B\frac{E_x}{E} \end{bmatrix}, \qquad \mathbf{u}_o = \begin{bmatrix} 0 \\ \frac{E_x}{E} \\ \frac{E_y}{E} \end{bmatrix}, \tag{11}$$

where we introduced a real quantity $\lambda \equiv |\mu|$. We will call these vectors "eigenvectors" of the matrix $\hat{F}$, although they are strictly speaking not. This should not cause a confusion, because we will not use vectors $\mathbf{u}_+$, $\mathbf{u}_-$ in the rest of this paper. We will use Greek indexes α, β for enumerating these vectors.

We notice that the eigenvectors $\{\mathbf{u}_0, \mathbf{u}_e, \mathbf{u}_o\}$ are orthogonal to each other,

$$|\mathbf{u}_0| = \mp 1, \qquad |\mathbf{u}_e| = \pm 1, \qquad |\mathbf{u}_o| = -1, \qquad \mathbf{u}_\alpha \cdot \mathbf{u}_\beta = 0,\ \alpha \neq \beta, \tag{12}$$

and it may be easily checked that the matrix $\hat{F}$ acts on them in the following way:

$$\hat{F}\mathbf{u}_0 = \mathbf{0}, \qquad \hat{F}\mathbf{u}_e = \pm\lambda\mathbf{u}_o, \qquad \hat{F}\mathbf{u}_o = \lambda\mathbf{u}_e. \tag{13}$$

We will also exploit the "vector product," computed according to the rule

$$(\mathbf{a} \times \mathbf{b})^i \equiv e^{ijk}a_j b_k, \qquad \{i, j, k\} = \{t, x, y\}. \tag{14}$$

e^{ijk} is the antisymmetric tensor; $e^{txy} = 1$. The following identities can be easily established,

$$\nabla \times (\mathbf{a} \times \mathbf{b}) = \mathbf{a}(\nabla \cdot \mathbf{b}) + (\mathbf{b} \cdot \nabla)\mathbf{a} - \mathbf{b}(\nabla \cdot \mathbf{a}) - (\mathbf{a} \cdot \nabla)\mathbf{b}, \qquad \mathbf{u}_e \times \mathbf{u}_o = \mathbf{u}_0, \tag{15}$$

and since

$$\mathbf{u}_e \times \mathbf{u}_o \cdot \mathbf{u}_0 = \mp 1 \tag{16}$$

we will say that the basis $\{\mathbf{u}_e, \mathbf{u}_o, \mathbf{u}_0\}$ is left- right-oriented.

If we compute the Minkovsky norm of the expression (9), recalling the normalization rule for the covariant velocity (3) and the basic properties of the trigonometric and hyperbolic functions, we come to the following relation between C and C_0,

$$C^2 - C_0^2 \equiv \pm 1, \tag{17}$$

which leaves only one of C, C_0 to be defined from the initial conditions. Scalar-multiplying (9) by $\mathbf{u}_0$, and then by $\mathbf{u}_e$ and $\mathbf{u}_o$, we derive

$$C_0 = \mp \mathbf{u}|_{s=0} \cdot \mathbf{u}_0, \qquad f_e(\phi_0) = \pm \frac{\mathbf{u} \cdot \mathbf{u}_e}{C}, \qquad f_o(\phi_0) = \mp \frac{\mathbf{u} \cdot \mathbf{u}_o}{C}, \tag{18}$$

where $\phi_0 \equiv \phi|_{s=0}$ is the value of the phase in the starting point of the particle trajectory. We integrate the linear solution (9) over the interval, to obtain the coordinate of the particle

$$\mathbf{x}_L(\phi) = \frac{mc^2}{e\lambda} \left\{ C_0 \mathbf{u}_0 (\phi - \phi_0) + C(\mathbf{u}_e f_o(\phi) + \mathbf{u}_o f_e(\phi)) - C(\mathbf{u}_e f_o(\phi_0) + \mathbf{u}_o f_e(\phi_0)) \right\}. \tag{19}$$

This expression also satisfies the initial condition for the coordinate. Thus, formulae (9), (17), (18), (19) together with definitions (10), (11) solve the unperturbed ($\mathbf{E}(\mathbf{x}) = \text{const}$, $\mathbf{B}(\mathbf{x}) = \text{const}$) linear problem (1), (2), (5), (6).

IV. THE ANALYTICAL SOLUTION OF THE EQUATIONS OF PARTICLE MOTION IN DRIFT APPROXIMATION

Now we turn to the non-linear problem, i.e., we no longer keep the fields fixed. It is important here to overcome a temptation of presenting the perturbed velocity in the form $\mathbf{u} = \mathbf{u}_L + \mathbf{u}_{\mathrm{NL}}$, as it is often done in semi-analytical methods (e.g., [35]), because in such a representation $\mathbf{u}_{\mathrm{NL}}$ appears to be proportional to ϕ^2 and fails to remain a small correction within even a modest phase change, which would undermine the use of the perturbation theory. We seek the solution for the velocity in a less evident form,

$$\mathbf{u}(\phi) = \tilde{C}_0(\phi)\tilde{\mathbf{u}}_0 + \tilde{C}(\phi)(\tilde{\mathbf{u}}_e f_e(\phi) \pm \tilde{\mathbf{u}}_o f_o(\phi)), \tag{20}$$

in which the phase remains undefined so far, the constants C, C_0 are replaced by functions of the phase, and the eigenvectors $\tilde{\mathbf{u}}_\alpha$ are no longer assumed to be constant, but depend on $\mathbf{x}$. For each given value of the vector $\mathbf{x}$ of the time-space domain the quantities $E_x(\mathbf{x})$, $E_y(\mathbf{x})$, $B(\mathbf{x})$, $\tilde{\lambda}(\mathbf{x}) \equiv |E_x(\mathbf{x})^2 + E_y(\mathbf{x})^2 - B(\mathbf{x})^2|^{1/2}$ are taken in that particular point $\mathbf{x}$ and a set of eigenvectors is computed according to the formulae (11). The vectors $\tilde{\mathbf{u}}_\alpha$ might thus be called "local" eigenvectors of the matrix $\hat{F}(\mathbf{x})$. It is very important for all our future derivations that the rules (12), (13) hold for these local eigenvectors (with the "local" eigennumber $\tilde{\lambda}(\mathbf{x})$ substituted in place of λ); it does not matter that all these quantities are dependent on $\mathbf{x}$. The functions $f_e(\phi)$, $f_o(\phi)$ remain trigonometric or hyperbolic as before, exactly as defined in (10). The relation (17) remains valid for $\tilde{C}$, $\tilde{C}_0$ due to the validity of formulae (3), (12), no matter whether $\hat{F}$ is a constant matrix or not. We plug the expression for the velocity (20) into the Newton–Lorentz law (1) to obtain

$$\begin{aligned} &\tilde{C}_0'\tilde{\mathbf{u}}_0 + C'(\tilde{\mathbf{u}}_e f_e(\phi) \pm \tilde{\mathbf{u}}_o f_o(\phi)) + \tilde{C}_o\tilde{\mathbf{u}}_0' + C(\tilde{\mathbf{u}}_e' f_e(\phi) \pm \tilde{\mathbf{u}}_o' f_o(\phi)) \\ &\pm \tilde{C}(\tilde{\mathbf{u}}_e f_o(\phi) + \tilde{\mathbf{u}}_o f_e(\phi))\phi' = \pm\tilde{C}\frac{e\tilde{\lambda}}{mc^2}(\tilde{\mathbf{u}}_e f_o(\phi) + \tilde{\mathbf{u}}_o f_e(\phi)), \end{aligned} \tag{21}$$

where the prime symbol stands for the derivative with respect to the interval, d/ds. When performing the substitution, the rules (13) may be used. We now require that "large" terms, proportional to both $\tilde{\lambda}$ and ϕ', get canceled by each other, which leads to a particular choice of the new independent variable:

$$\frac{d\phi}{ds} = \frac{e}{mc^2}\sqrt{\left|E_x(\mathbf{x})^2 + E_y(\mathbf{x})^2 - B(\mathbf{x})^2\right|}; \qquad \phi|_{s=0} = \phi_0. \tag{22}$$

To extract a scalar equation for $\tilde{C}_0$ from the vector Eq. (21), we scalar-multiply it by $\tilde{\mathbf{u}}_0$:

$$\tilde{C}_0' = \tilde{C}\tilde{\mathbf{u}}_0' \cdot (\mp\tilde{\mathbf{u}}_e f_e - \tilde{\mathbf{u}}_o f_o). \tag{23}$$

The following identities were used: $\tilde{\mathbf{u}}_0' \cdot \tilde{\mathbf{u}}_0 = (\tilde{\mathbf{u}}_0 \cdot \tilde{\mathbf{u}}_0)'/2 = 0$, $\tilde{\mathbf{u}}_\nu' \cdot \tilde{\mathbf{u}}_0 = (\tilde{\mathbf{u}}_\nu' \cdot \tilde{\mathbf{u}}_0)' - \tilde{\mathbf{u}}_\nu \cdot \tilde{\mathbf{u}}_0' = -\tilde{\mathbf{u}}_\nu \cdot \tilde{\mathbf{u}}_0'$, $\nu = \{e, o\}$, which follow from the rules (12). The expression (23) is the *exact*, but implicit solution of (1), because $\mathbf{x}$ enters it as a coordinate, at which eigenvectors are evaluated. That formula also holds if C and C_0 are interchanged, as it follows from (17). It is more useful for our perturbative approach, than (1), because it involves only terms which are small under our assumption.

Next, we replace the derivative d/ds with $\tilde{\mathbf{u}} \cdot \nabla$,

$$\nabla \equiv \mathbf{e}_t \frac{\partial}{c\partial t} + \mathbf{e}_x \frac{\partial}{\partial x} + \mathbf{e}_y \frac{\partial}{\partial y} \tag{24}$$

to obtain

$$\tilde{C}_0' = \tilde{C}(\mp\tilde{\mathbf{u}}_e f_e - \tilde{\mathbf{u}}_o f_o) \cdot (\tilde{C}_0\tilde{\mathbf{u}}_0 + \tilde{C}(\tilde{\mathbf{u}}_e f_e \pm \tilde{\mathbf{u}}_o f_o) \cdot \nabla)\tilde{\mathbf{u}}_0. \tag{25}$$

Together with (23), (25) is also an exact formula.

We now exploit our major assumption that the field components vary only slightly and smoothly in space and time. The derivatives $\partial\tilde{u}_0^i/\partial x^j$ ought to be evaluated using the known derivatives of the field components and may be considered small in the drift approximation. As it is evident from (25), $\tilde{C}$, $\tilde{C}_0$ vary only a little along the trajectory of our particle, and we may substitute some constant values for these quantities, from within the interval of their variation into the RHS of (25). We will drop the tilde symbol from above an $\mathbf{x}$-dependent quantity whenever we imply such a substitution of that quantity from the range of its variation. We will later specify which exact value should be substituted to achieve the maximum accuracy.

In the rest of the paper we will confine ourselves to the "gyro" case, $|E| < |B|$. Such a choice is dictated by two reasons: first it is a more frequent configuration in both technical and astrophysical problems, and second, the proposed method allows integration of particle trajectory over a greater time step in this case, and is thus more worth using. This is because in the "acceleration" case, $|E| > |B|$, functions f_e, f_o grow exponentially and become very large after even a moderate phase shift, which limits the range of applicability of the method considerably.

It proves useful to introduce a matrix

$$\hat{V} = \begin{bmatrix} \frac{\partial(B(\mathbf{x})/\tilde{\lambda}^2)}{c\partial t} & \frac{\partial(B(\mathbf{x})/\tilde{\lambda}^2)}{\partial x} & \frac{\partial(B(\mathbf{x})/\tilde{\lambda}^2)}{\partial y} \\ \frac{\partial(E_y(\mathbf{x})/\tilde{\lambda}^2)}{c\partial t} & \frac{\partial(E_y(\mathbf{x})/\tilde{\lambda}^2)}{\partial x} & \frac{\partial(E_y(\mathbf{x})/\tilde{\lambda}^2)}{\partial y} \\ -\frac{\partial(E_x(\mathbf{x})/\tilde{\lambda}^2)}{c\partial t} & -\frac{\partial(E_x(\mathbf{x})/\tilde{\lambda}^2)}{\partial x} & -\frac{\partial(E_x(\mathbf{x})/\tilde{\lambda}^2)}{\partial y} \end{bmatrix} \tag{26}$$

of partial derivatives of components of $\tilde{\mathbf{u}}_0/\tilde{\lambda}$ and define "matrix components" of the eigenvectors as

$$\mathbf{V}_\alpha \equiv \hat{V}\tilde{\mathbf{u}}_\alpha, \qquad V_{\alpha\beta} \equiv \tilde{\mathbf{u}}_\alpha \cdot \hat{V}\tilde{\mathbf{u}}_\beta, \qquad \Lambda_\alpha \equiv \nabla\tilde{\lambda} \cdot \tilde{\mathbf{u}}_\alpha. \tag{27}$$

Next, we will use the symbol $\oint$ to designate the integration over an integer number of periods of our trigonometric functions. Moreover, for compactness we will drop explicit substitution of the boundaries of integration. Thus, we will write $\oint \phi \sin\phi\, d\phi = -\phi\cos\phi$ instead of $\int_\phi^{\phi+2\pi n} \chi \sin\chi\, d\chi = \phi\cos\phi - (\phi + 2\pi n)\cos(\phi + 2\pi n)$. We will use the letter Δ to denote the difference between the values of some quantity over the initial and final points of the particle trajectory.

With these acronyms in mind, we integrate Eq. (23) and evaluate the particle displacement. For simplicity, we do not intend to find all the unknown quantities at *any* value of the interval; we will instead confine ourselves to those values of Δs, which correspond to $\Delta\phi = 2\pi n$, $n = 1, 2, 3, \ldots$. First of all, we find

$$\Delta\tilde{C}_0 \approx \pi n \frac{mc^2}{e} C^2 (V_{ee} + V_{oo}), \qquad \Delta\tilde{C} \approx \pi n \frac{mc^2}{e} C C_0 (V_{ee} + V_{oo}). \tag{28}$$

To derive these formulae, we integrated (25) by parts after replacing $\tilde{\mathbf{u}}_0' \Rightarrow \tilde{\lambda}(\tilde{\mathbf{u}}_0/\tilde{\lambda})'$. Such a substitution is valid, since $\tilde{\mathbf{u}}_0$ is orthogonal to both $\tilde{\mathbf{u}}_e$ and $\tilde{\mathbf{u}}_o$, and thus $\tilde{\lambda}(\tilde{\mathbf{u}}_0/\tilde{\lambda})' \cdot \tilde{\mathbf{u}}_{e,o} = \tilde{\mathbf{u}}_0' \cdot \tilde{\mathbf{u}}_{e,o}$. Those were numerical experiments we carried out, which hinted us to push $\tilde{\lambda}$ under the derivative operator. Indeed, if $\tilde{\lambda}$ is kept as a constant factor in the expression for $\Delta\tilde{C}_0$, with a correspondent redeclaration of V_{ee}, V_{oo}, the accuracy of the formula, which would replace (28) in that case would become substantially lower. In fact, together with (28) the formula for particle displacement to be derived also gains in accuracy from such a substitution. Technically, only one of $\Delta\tilde{C}$ and $\Delta\tilde{C}_0$ should be determined by the above expressions, while the formula (17) should be used for the other in order to preserve the velocity normalization rule (3).

We once again exploit integration by parts, to integrate (5):

$$\begin{aligned}
\Delta\mathbf{x} &= (mc^2/e) \oint (\tilde{C}_0\tilde{\mathbf{u}}_0 + \tilde{C}(\tilde{\mathbf{u}}_e\cos\phi - \tilde{\mathbf{u}}_o\sin\phi))/\tilde{\lambda}\, d\phi \\
&\approx (mc^2/e\lambda)\{C_0\mathbf{u}_0\phi + C(\mathbf{u}_e\sin\phi + \mathbf{u}_o\cos\phi)\} \\
&\quad - (m^2c^4/e^2\lambda) \oint (\tilde{C}_0\tilde{\mathbf{u}}_0/\tilde{\lambda})'\phi\, d\phi \\
&\quad - (m^2c^4/e^2\lambda) \oint \left((\tilde{C}\tilde{\mathbf{u}}_e/\tilde{\lambda})'\sin\phi + (\tilde{C}\tilde{u}_o/\tilde{\lambda})'\cos\phi\right) d\phi
\end{aligned} \tag{29}$$

The first of the integrals in the RHS evaluates to

$$\begin{aligned}
\oint (C_0\mathbf{u}_0/\lambda)'\phi\, d\phi \approx \mathbf{u}_0 \Big\{ &(\pi^2n^2 + \pi n\phi_0)C^2(V_{ee} + V_{oo}) + 2\pi n C C_0(\sin\phi_0 V_{e0} + \cos\phi_0 V_{o0}) \\
&+ \pi n \sin 2\phi_0 C^2(V_{ee} - V_{oo}) + \frac{1}{2}\pi n\cos 2\phi_0 C^2(V_{oe} + V_{eo}) \Big\} \\
&+ 2(\pi^2n^2 + \pi n\phi_0)C_0^2\mathbf{V}_0 + 2\pi n\sin\phi_0 C C_0\mathbf{V}_e + 2\pi n\cos\phi_0 C C_0\mathbf{V}_o.
\end{aligned} \tag{30}$$

Now we turn to the second integral. The following preliminary considerations are in order:

$$\oint ((\tilde{\mathbf{u}}_e/\tilde{\lambda})' \sin\phi + (\tilde{\mathbf{u}}_o/\tilde{\lambda})' \cos\phi)\, d\phi$$
$$\approx \pi n C \lambda ((\mathbf{u}_e/\lambda \cdot \nabla)(\tilde{\mathbf{u}}_o/\tilde{\lambda}) - (\mathbf{u}_o/\lambda \cdot \nabla)(\tilde{\mathbf{u}}_e/\tilde{\lambda}))$$
$$= \pi n C \big(\nabla\tilde{\lambda} \times \mathbf{u}_0/\lambda^2 - \nabla \times (\tilde{\mathbf{u}}_0/\tilde{\lambda}) + \mathbf{u}_e \nabla \cdot (\tilde{\mathbf{u}}_o/\tilde{\lambda}) - \mathbf{u}_o \nabla \cdot (\tilde{\mathbf{u}}_e/\tilde{\lambda})\big). \qquad (31)$$

The vector product was introduced in (14) and its features (15) were used.

We are now ready to accomplish the evaluation of the second integral in (29):

$$\oint ((C\mathbf{u}_e/\tilde{\lambda})' \sin\phi + (C\mathbf{u}_o/\tilde{\lambda})' \cos\phi)\, d\phi$$
$$= \pi n C \big((\Lambda_o \mathbf{u}_e - \Lambda_e \mathbf{u}_o)/\lambda^2 - \nabla \times (\tilde{\mathbf{u}}_0/\tilde{\lambda}) + \mathbf{u}_e \nabla \cdot (\tilde{\mathbf{u}}_o/\tilde{\lambda}) - \mathbf{u}_o \nabla \cdot (\tilde{\mathbf{u}}_e/\tilde{\lambda})\big)$$
$$+ \pi n C_0^2 (V_{e0}\mathbf{u}_o - V_{o0}\mathbf{u}_e). \qquad (32)$$

Let us consider the case of a very small electric field, e.g., the vicinity of a knot in a Langmuir wave, excited in a magnetized plasma. The even and odd eigenvectors, if chosen according to (11), are very unstable in this domain. This breaks our main assumption that they are quasi-stationary along the particle trajectory. A more suitable choice of these eigenvectors, which however does not affect any of the expressions obtained above, is proposed below.

Let us choose some vector field, $\mathbf{a}(\mathbf{x})$, and introduce

$$\tilde{\mathbf{v}}_e \equiv \tilde{\mathbf{u}}_0 \times \mathbf{a}, \qquad \tilde{\mathbf{v}}_o \equiv \tilde{\mathbf{v}}_e \times \tilde{\mathbf{u}}_0 = \mathbf{a} - (\tilde{\mathbf{u}}_0 \cdot \mathbf{a})\tilde{\mathbf{u}}_0. \qquad (33)$$

If we require that the even vector $\tilde{\mathbf{v}}_e$ satisfies the same normalization rule, as $\tilde{\mathbf{u}}_e$ (12), we conclude that $(\tilde{\mathbf{u}}_0 \cdot \mathbf{a})^2 - |\mathbf{a}| = 1$, and, consequently, $\mathbf{a}$ may not be directed along $\tilde{\mathbf{u}}_0$. Other normalization rules (12) are clearly satisfied with the new eigenvectors $\{\tilde{\mathbf{u}}_0, \tilde{\mathbf{v}}_e, \tilde{\mathbf{v}}_o\}$ in place of $\{\mathbf{u}_0, \mathbf{u}_e, \mathbf{u}_o\}$. The above choice of sign of $\tilde{\mathbf{v}}_e$ makes the triplet $\{\tilde{\mathbf{u}}_0, \tilde{\mathbf{v}}_e, \tilde{\mathbf{v}}_o\}$ right-oriented, see (16). In accordance with the invariance of our solution under the rotation of even and odd eigenvectors, we may use the newly constructed set as a basis, instead of $\{\tilde{\mathbf{u}}_0, \tilde{\mathbf{u}}_e, \tilde{\mathbf{u}}_o\}$. That set is more appropriate for our purposes, because smoothness of electromagnetic fields causes smoothness of all three of the new vectors, provided that $\mathbf{a}(\mathbf{x})$ is smooth and is not directed along $\mathbf{u}_0$ in any point of our space-time domain. Since we are most interested in the gyro case $|E| < |B|$, we may choose $\mathbf{a}(\mathbf{x})$ to be constant and directed along the x-axis. Such a choice of $\mathbf{a}$ is impossible if we do not assume $|E| < |B|$, because in the acceleration case $|E| > |B|$ $\tilde{\mathbf{u}}_0$ might be at some point directed along the x-axis as well, thus being collinear to $\mathbf{a}$, which is forbidden. We do not substitute this particular set of the eigenvectors, although the numerical results presented below were obtained under it. We leave the choice of eigenvectors arbitrary within the mentioned bounds (they should form a right orthogonal triplet, the vector $\tilde{\mathbf{u}}_0$ defined in (11) and their components smooth functions of $\mathbf{x}$). In the following formulae by $\tilde{\mathbf{u}}_\alpha$ we mean an arbitrary set of vectors which satisfy the mentioned conditions.

Let us now take a closer look at the formulae (28), (29), (30), (32). Each of these expressions is a sum of terms, proportional to ϕ, $\sin\phi$, $\cos\phi$, or products of those. Since all the above derivations were primarily designed for the use in cases, when a particle rotates

a large number of times around a magnetic field line, these terms might be of very different orders of magnitude. Indeed, even after a single gyrorotation, the difference between such terms is $\sim\phi = 2\pi$, and in realistic configurations a particle may rotate by hundreds of periods in a single time step of the algorithm. It seems reasonable to introduce a large parameter ϕ besides our standard smallness parameter into consideration. We will evaluate the terms, proportional to the highest power of ϕ, with a higher accuracy with respect to the smallness parameter than the other terms. A standard procedure of integration by parts may be used for this purpose. We will illustrate the procedure with a simple example,

$$\int_i^f ax\,dx = \int_i^f ad\frac{x^2}{2} = \Delta\frac{ax^2}{2} - \frac{1}{2}\int_i^f \frac{da}{dx}x^2\,dx \approx \overline{a}\overline{x}\Delta x + \frac{1}{12}\Delta a(\Delta x)^2, \qquad a \approx \text{const}, \tag{34}$$

where the bar designates taking the arithmetic average of the boundary points, $\bar{y} \equiv (y^f + y^i)/2$, and Δ stands for the difference over the interval, $\Delta y \equiv y^f - y^i$. The symbols ^{i} and ^{f} denote the left and the right points of the interval, respectively. This procedure leads to the standard "arithmetic average" rule of the derivative approximation in the simple case of $\int a\,dx, a \approx$ const, which we apply to the evaluation of ΔC instead of (28). This provides us with the means to fix the arbitrariness in choosing a particular value of a quantity from within the interval of its change over the particle orbit in order to achieve the highest accuracy of the formula.

We now write the expression for the displacement and final coefficients of a particle after an integer number of gyrations in the drift approximation ($\Delta\phi = 2\pi n$, $n = 1, 2, 3, \ldots$):

$$\begin{aligned}
\Delta C_0 &\approx C^{i\,2} q\big(1 + C_0^i q\big), \qquad \text{where } q \equiv \pi n\frac{mc^2}{e}(\bar{V}_{ee} + \bar{V}_{oo});\\
\Delta\mathbf{x} &\approx (mc^2/e)\big\{2\pi n C_0^f \mathbf{u}_0^f/\lambda^f + \Delta(C\mathbf{u}_0\phi_0/\lambda + C(\mathbf{u}_e\sin\phi_0 + \mathbf{u}_o\cos\phi_0)/\lambda)\big\}\\
&\quad + \pi n(m^2c^4/e^2\bar{\lambda})\bar{\mathbf{u}}_0\Big\{(\pi n + \phi_0)\overline{C^2}(\bar{V}_{ee} + \bar{V}_{oo}) + 2\bar{C}\bar{C}_0(\sin\phi_0\bar{V}_{e0} + \cos\phi_0\bar{V}_{o0})\\
&\quad + \sin 2\phi_0\overline{C^2}(\bar{V}_{ee} - \bar{V}_{oo}) + \frac{1}{2}\cos 2\phi_0\overline{C^2}(\bar{V}_{oe} + \bar{V}_{eo})\Big\}\\
&\quad + 2\pi n(m^2c^4/e^2\bar{\lambda})\big\{(\pi n + \phi_0)\bar{C}_0^2\bar{\mathbf{V}}_0 + \sin\phi_0\bar{C}\bar{C}_0\bar{\mathbf{V}}_e + \cos\phi_0\bar{C}\bar{C}_0\bar{\mathbf{V}}_o\big\}\\
&\quad + \pi n(m^2c^4/e^2\bar{\lambda})\bar{C}\big\{(\bar{\Lambda}_o\bar{\mathbf{u}}_e - \bar{\Lambda}_e\bar{\mathbf{u}}_o)/\bar{\lambda}^2 - \overline{\nabla\times(\tilde{\mathbf{u}}_0/\tilde{\lambda})} + \bar{\mathbf{u}}_e\overline{\nabla\cdot(\tilde{\mathbf{u}}_o/\tilde{\lambda})}\\
&\quad - \bar{\mathbf{u}}_o\overline{\nabla\cdot(\tilde{\mathbf{u}}_e/\tilde{\lambda})}\big\} + \pi n(m^2c^4/e^2\bar{\lambda})\bar{C}_0^2(\bar{V}_{e0}\bar{\mathbf{u}}_o - \bar{V}_{o0}\bar{\mathbf{u}}_e)\\
&\quad + \frac{1}{6}\pi^2n^2(m^2c^4/e^2)\Delta\big(C^2\mathbf{u}_0(V_{ee} + V_{oo})/\lambda\big) + \frac{1}{3}\pi^2n^2(m^2c^4/e^2)\Delta\big(C_0^2\mathbf{V}_0/\lambda\big).
\end{aligned} \tag{35}$$

There is yet one more problem about these formulae. Indeed, we know how to evaluate the displacement of a particle, provided we know the eigenvectors and values of $\hat{F}$ in the final point of the particle trajectory. This point, however, remains unknown until we apply the above expressions. Although not the coefficients, as in (25), but the particle displacement is still given by an implicit expression. We note again that the terms in the formula for particle displacement are of different orders of magnitude. We will try to evaluate the largest term, which is proportional to n and does not contain derivatives of the field (it is due to the $\mathbf{E}\times\mathbf{B}$

drift) with the highest accuracy, and will use a rough estimate for the eigenvectors and values, when evaluating other terms. We may choose a linear displacement, which results from the $\mathbf{E}\times\mathbf{B}$ drift, as a point at which to take approximate values for eigenvectors and their final derivatives, and we will define the large term with the implicit approach,

$$\Delta x^i = 2\pi n \frac{mc^2}{e} C_0^f \left.\frac{\partial\left(\tilde{u}_0^i/\tilde{\lambda}\right)}{\partial x^j}\right|_{\mathbf{x}=\mathbf{x}^f} \Delta x^j + \Delta x^i_{\text{expl}}, \tag{36}$$

where $\Delta\mathbf{x}_{\text{expl}}$ designates smaller terms of the formula (35) together with the large term, evaluated at the starting point. The matrix of partial derivatives occurring in the above formula is exactly $\hat{V}^f$, and it needs to be computed for other purposes as well.

After we formulated the final analytical result of our algorithm, we turned to the issues of its accuracy. The perturbation parameter used in the algorithms is $\Delta\tilde{C}/C$ (or a similar expression with C_0), and it is assumed to be small. Besides this, the quantities related to $\hat{V}$ are assumed to be constant throughout the integration step here and there in the *analytical* part of the work, but this restriction is softened a little by applying a higher order approach to the finite-difference formula. We emphasize that $\Delta\mathbf{x}$ is *not* used as the perturbation parameter of our expansion. There is a major difference between the two: $\Delta\mathbf{x}\propto n$, while $\Delta C\propto\hat{V}n$. Moreover, there are no terms $\propto\hat{V}n^2$ in the expression for ΔC, contrary to the formula for $\Delta\mathbf{x}$.

As it follows from (26), (27), (28), the difference between the non-linear coefficient and its linear counterpart depends on the phase and partial field derivatives through the parameter

$$\varepsilon \propto |\Delta\lambda|/\bar{\lambda}, \tag{37}$$

where bar and Δ operators are defined as above. The choice of $\bar{\lambda}$ as the natural quantity for the accuracy estimates instead of some component of the electromagnetic field is dictated by its Lorentz invariance. Our analytical derivation provided the correct linear dependence of all the quantities obtained on ε, but we formally disregarded the non-linear terms, quadratic in ε. When we used higher-order implicit techniques for writing expressions (35), we raised the accuracy of our finite-difference expressions over the analytical ones by one more factor of ε. We therefore expect the error of our algorithm to be at least cubic in ε.

This estimate for the accuracy is good only for the configurations, when parameters of the problem do not vary much within a *single* period of the particle motion. If they do, the approximate equalities similar to $\oint V_{0e}\cos\phi\, d\phi \approx 0$ are no longer valid, and we relied on such expressions heavily in our derivations. We come to a conclusion that the algorithm works better for the case of very many gyrations of a particle in very slow-varying fields than for the case of less rotations in the fields with larger gradients or rise speeds. Quantitatively, we expect that besides the cubic inaccuracy, the algorithm has also a linear one, the latter, however, being important only if the number of gyrations is small. In other words, we may say that the linear inaccuracy prevails over the cubic one only in very accurate simulations,

$$\frac{\delta C}{C} \propto \frac{\delta\mathbf{x}}{R} \propto \begin{cases} \varepsilon^3, & \text{if } \varepsilon > \varepsilon_{\text{cr}} \\ \varepsilon, & \text{if } \varepsilon < \varepsilon_{\text{cr}}, \end{cases} \tag{38}$$

where R is some fixed length, i.e., particle gyroradius. We will demonstrate this effect numerically in Section V.

V. RESULTS OF A NUMERICAL EXPERIMENT

In this section we present the results of comparison of the proposed algorithm with a well-tested standard method of numerical integration of the system of differential equations for particle motion. The traditional technique returns a set of coordinates and velocities, which form a curve in **x** space. Our algorithm returns just the coordinates of a point, where the particle is located after the specified interval of time. The essence of the test is to verify how far this point lies from the curve, generated by the traditional solver. As the solver we used the ode45 routine from the matlab [41] package, which we ran with the accuracy 10^{-9}. We verified that significant modification of this parameter did not change those digits, important for our comparisons. Matlab's internal number representation is double word and the correct processor arithmetics in this range can be guaranteed.

For the particular problem we used a typical $(2+1)$D configuration. A large magnetic field B pointed in the $+\mathbf{e}_z$ direction, and a smaller electric field $\mathbf{E}$ was applied along the x and y directions. All the fields depended weakly on all the three components of $\mathbf{x}$. These fields caused gyrorotation of a particle along the z axis with the drift of the guiding center of the particle orbit due to "$\mathbf{E}\times\mathbf{B}$" as well as ∇B and gradients of $\mathbf{E}$. The fields did *not* satisfy Maxwell's equations, in particular Faradey's law. We consider the independence of our algorithm, which treats particles of the equations for fields to be its advantage.

Three sets of the test were run. In one of them these non-linear drifts were large, so that the proposed algorithm worked well only within a small time interval (just a few gyrorotations), and in the other two the drifts were small, so that the algorithm provided quite an accurate result when a few tens of gyrorotations were computed in a single time step. We emphasize that we mean here solely drifts, proportional to partial derivatives of the fields. The magnitude of the linear drift $\mathbf{E}\times\mathbf{B}$ should not affect the accuracy of our

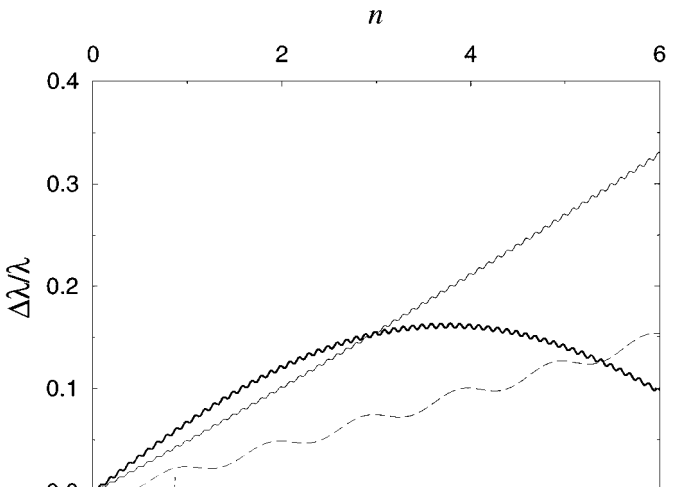

FIG. 1. Relative difference of the parameter $\tilde{\lambda}$ between the final and initial points of the particle trajectory, as a function of the number of gyrations n for different field configurations being simulated. Thin line, weak field gradients, monotonous case; thick line, weak field gradients, non-monotonous case; dashed line, strong field gradients. Lower abscissa, for the case of weak; upper abscissa, for strong field gradients.

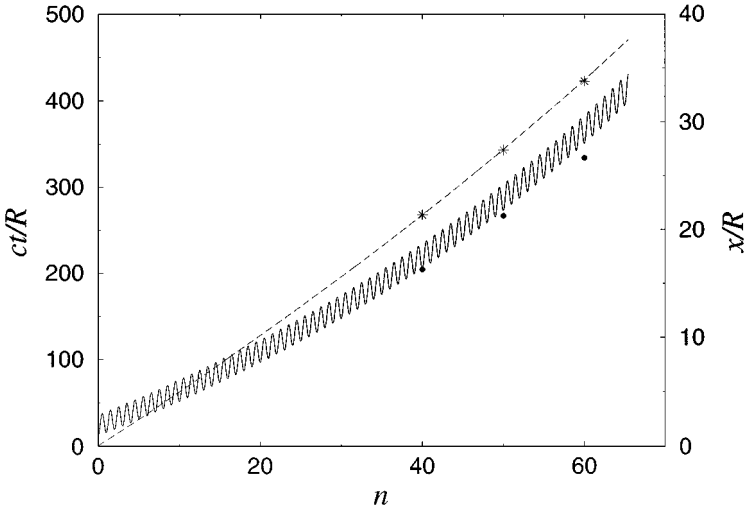

FIG. 2. Normalized particle x (solid line) and t (dashed line) coordinate versus the number of particle gyrations in the monotonous case with weak gradients, as generated by ode45 routine. $*$, normalized t; and $\bullet$, normalized x coordinate, as evaluated with the new algorithm for 3 independent jumps from $n = 0$ to $n = 40, 50, 60$.

algorithm. Those two kinds of tests with small field gradients differed from each other by the time-space dependence of the fields: $\tilde{\lambda}$ was a monotonous function of ϕ in one category and had a local extremum in the other. When speaking about the extremum we do not mean gyro-scale variations of $\tilde{\lambda}$, but rather variations on the scale of $\Delta\mathbf{x}$.

The following should be made clear before we discuss the results of the simulations. We found it more illustrative to run the algorithm in the regime, when its inaccuracy is just visible from the plots. In practical simulations one should choose a smaller parameter ε (37) to obtain robust results.

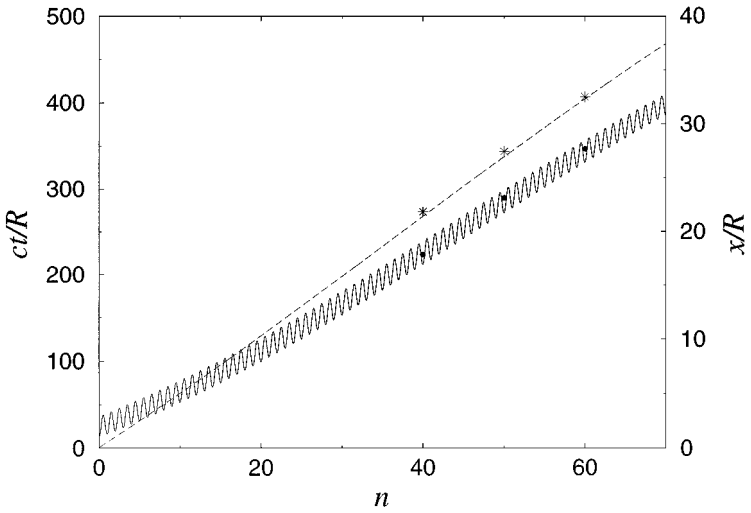

FIG. 3. Normalized particle x (solid line) and t (dashed line) coordinate versus the number of particle gyrations in the non-monotonous case with weak gradients, as generated by ode45 routine. $*$, normalized t; and $\bullet$, normalized x coordinate, as evaluated with the new algorithm for 3 independent jumps from $n = 0$ to $n = 40, 50, 60$.

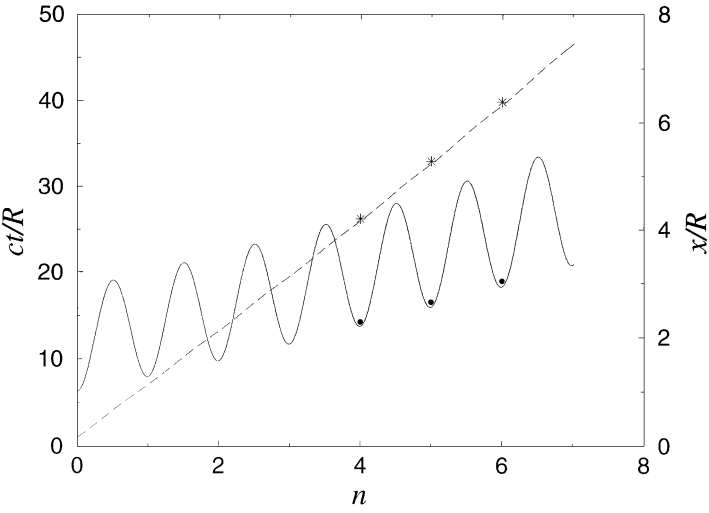

FIG. 4. Normalized particle x (solid line) and t (dashed line) coordinate versus the number of particle gyrations in the case with strong gradients, as generated by ode45 routine. $*$, normalized t; and $\bullet$, normalized x coordinate, as evaluated with the new algorithm for 3 independent jumps from $n = 0$ to $n = 4, 5, 6$.

Since the natural independent variable of our analytical formulae is n, the number of particle gyrations around a magnetic field line, we chose it as an abscissa for most of our plots. Time and space coordinates are normalized by the gyroradius in some typical magnetic field for a simulation, which is not necessarily $\bar{B}$, computed for some typical Lorentz factor of a particle. Since the gyro phase of the particle is an independent variable, the Lorentz factor is the only parameter needed to specify the particle velocity exactly, since the velocity vector is directed at the angle ϕ with respect to $\tilde{\mathbf{u}}_e$. Of the three particle coordinates $\{t, x, y\}$ we plot only two (t and x) for compactness. We did not consider all the field components worth plotting versus every coordinate for every simulation. We instead plotted just the

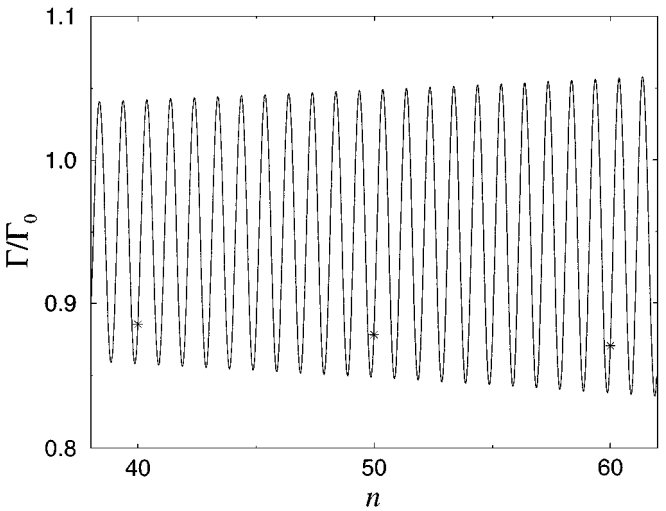

FIG. 5. Normalized particle Lorentz factor in the case of monotonous weak gradients (line), as generated by ode45 routine. $*$, the same, evaluated with the new algorithm for 3 independent jumps from $n = 0$ to $n = 40, 50, 60$.

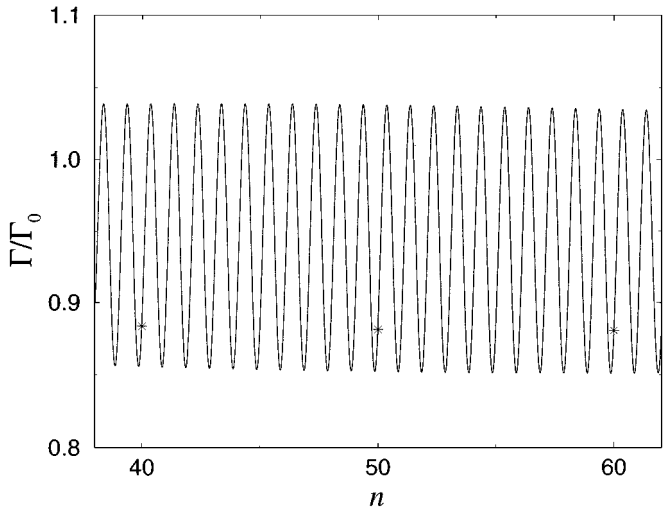

FIG. 6. Normalized particle Lorentz factor in the case of non-monotonous weak gradients (line), as generated by ode45 routine. $*$, the same, evaluated with the new algorithm for 3 independent jumps from $n=0$ to $n=40, 50, 60$.

parameter $\varepsilon(n)$, as it is defined in (37), see Fig. 1. These data are clearly extracted from the direct numerical integration by ode45. We define absolute errors of the simulation as

$$\begin{aligned} \delta\mathbf{x} &\equiv \left\{(\Delta t-\Delta t_{\text{ode}})^2+(\Delta x-\Delta x_{\text{ode}})^2+(\Delta y-\Delta y_{\text{ode}})^2\right\}^{1/2}, \\ \delta\Gamma &\equiv |\Delta\Gamma-\Delta\Gamma_{\text{ode}}|, \end{aligned} \tag{39}$$

where Γ is the particle Lorentz factor and $_{\text{ode}}$ marks the quantities generated by the ode45 routine. It is interesting to notice that the normalization of the eigenvectors (11) is such that C, C_0, and Γ are of the same order of magnitude. When estimating the inaccuracies

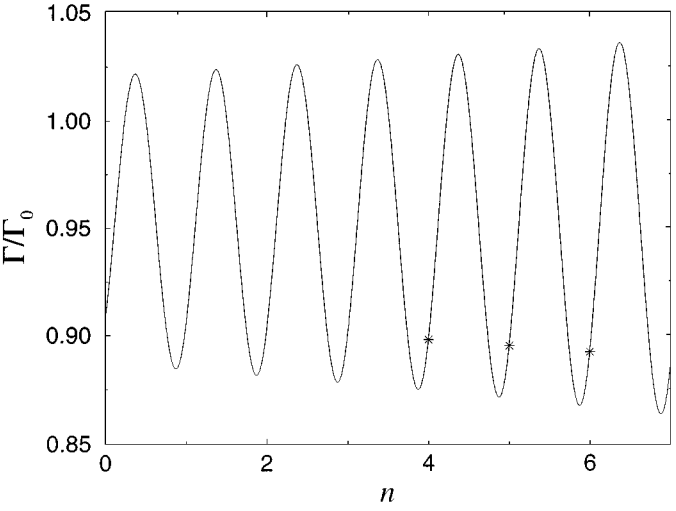

FIG. 7. Normalized particle Lorentz factor in the case of strong gradients (line), as generated by ode45 routine. $*$, the same, evaluated with the new algorithm for 3 independent jumps from $n=0$ to $n=4, 5, 6$.

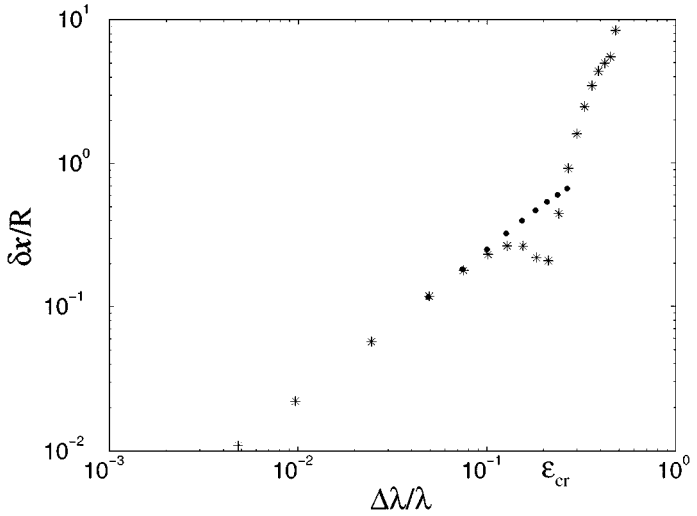

FIG. 8. Inaccuracy of the new algorithm in $\Delta\mathbf{x}$ (see formula (38)) as a function of field variation over the scale of the particle orbit. $*$, for the monotonous case with weak gradients; $\bullet$, high gradient case. $\varepsilon_{\rm cr}$ marks the critical parameter for the weak gradient case.

visually from the plots, one should bear in mind that Δx and $c\Delta t$ coordinates differ by an order of magnitude.

To avoid misunderstanding we also note that *all* the discrete marks on the plots (stars and bullets) correspond to independent *single iteration* pushes from the phase $n = 0$, which was thus the starting point of any push presented on the plots. Formulae (35), (36) were applied only once to obtain each such set of marks on the plots from the particle position and velocity at $n = 0$.

From Figs. 4, 7 one can see that the algorithm is reasonably correct within $\varepsilon \simeq 10\%$ if the number of gyrations within this ε is about a few. Comparing those plots with Figs. 2, 3, 5, 6, we notice that the accuracy is at least a few times higher for the large number of gyrations (with the same value of ε). We believe this is mainly due to the competition of two terms in (38). This is also supported by Fig. 8, which clearly indicates that we are in the cubic regime of the formula (38) in the weakly non-stationary cases, and in the linear one in the configuration with larger field gradients.

It seemed to us to be an interesting feature of the algorithm that the fields or at least $\tilde{\lambda}(\phi)$ does not have to be monotonous over the particle trajectory although it was assumed that the partial derivatives of all the field components are constant across the time step. We attribute such a gain of accuracy to the correction we introduced in the final-difference scheme.

From comparison of the plots illustrating x and t dependence on n with the ones devoted to the Lorentz factor it is evident that the latter is computed with a higher accuracy than $\Delta\mathbf{x}$. That is probably because the formulae for $\Delta\tilde{C}$ (or $\Delta\tilde{C}_0$) are substantially simpler than those for $\Delta\mathbf{x}$. As it was mentioned above, we gained a lot in $\Delta\tilde{C}$ accuracy with the proper declaration of $\hat{V}$.

An important characteristic of a PIC code is the amount of time the code spends per elementary push of a single superparticle. This time usually counts not only CPU cycles

spent in a ODE-solving routine, but also field interpolation expenses, superparticle gather/scatter operations, etc. It is therefore hard to estimate the effectiveness of solely a pushing algorithm before it is incorporated into some working PIC code. From *very* preliminary considerations we were able to conclude that the new pusher requires about a few ten times more CPU cycles per particle per push than a traditional scheme, similar to Boris push. On the other hand, the time interval the new push allows (under the conditions of its applicability) may be up to hundreds of times longer than a traditional pusher would permit (see Figs. 2, 3, 5, 6). That means that in a wide class of ultrarelativistic problems the new scheme would be about an order of magnitude or more superior to a traditional one.

VI. CONCLUSIONS

The proposed new algorithm for long time-step advance of a relativistic particle in weakly non-homogeneous and non-stationary electromagnetic fields with $|B| > |E|$ in $(2+1)$D proves to be a good alternative to traditional particle pushing, i.e., Boris push [5] in such configurations. The CPU time required for a single push is independent of the duration of the push (the energy of the particle). The linear $\mathbf{E} \times \mathbf{B}$ drift is accounted for with a higher accuracy, and all the non-linear drifts (∇B, polarization, etc.) with similar accuracies since they are not separated formally from one another. The accuracy of the particle data after the push is better the less energetic the particle is. The results of numerical experiments conducted basically confirm the estimate (38) for the numerical error introduced, Fig. 8, which provides a ground for choosing the optimal pushing interval. The elementary pushing step may extend for an integer number of gyroperiods only. Additional short time-step pre-push or post-push should be used to obtain displacement by an arbitrary Δt. Maxwell equations are not exploited in the design of the algorithm, i.e., it may be effectively used if they are violated either intentionally or due to computational errors.

In its present form the method would be applicable to typical configurations of high-energy astrophysics, like those described in the papers [1, 4, 6, 9].

ACKNOWLEDGMENTS

It is a pleasure to acknowledge very helpful conversations with Dr. William Peter and Professor Vladimir Usov. The research was supported by MINERVA foundation, Munich, Germany.

REFERENCES

1. M. V. Smolsky and V. V. Usov, Relativistic beam-magnetic barrier collision and nonthermal radiation of cosmological γ-ray bursters, *Ap. J.* **461**, 858 (1996).
2. B. B. Godfrey, Numerical Cherenkov instabilities in electromagnetic particle codes, *J. Comput. Phys.* **15**, 504 (1974).
3. A. B. Langdon, Analysis of the time integration in plasma simulation, *J. Comput. Phys.* **30**, 202 (1979).
4. R. Blandford and D. Eichler, Particle acceleration at astrophysical shocks: A theory of cosmic ray origin, *Phys. Rep*. **154**(1), 1 (1987).
5. C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation* (IOP Publishing, 1991).
6. V. V. Usov and M. V. Smolsky, Wide ultrarelativistic plasma beam-magnetic barrier collision and astrophysical applications, *Phys. Rev. E* **57** (1998).
7. P. Burger, Theory of large-amplitude oscillations in the 1-dimensional low-pressure cesium themionic converter, *J. Appl. Phys.* **36**, 1938 (1965).

8. P. Burger, D. A. Dunn, and A. S. Halsted, Computer experiments on the randomization of electrons in a collisionless plasma, *Phys. Fluids* **8**, 2263 (1965).
9. M. Hoshino, J. Arons, Y. A. Gallant, and A. B. Langdon, Relativistic magnetosonic shock waves in synchrotron sources: Shock structure and nonthermal acceleration of positrons, *Ap. J.* **390**, 454 (1992).
10. S. G. Alikhanov, R. Z. Sagdeev, and P. S. Chebotaev, Destruction of a large-amplitude ion-acoustic wave, *Sov. Phys.—JETP* **30**, 847 (1970); *Zh. Ekspr. Teor Fiz.* **57**, 1565 (1969).
11. R. Mason, Computer simulation of ion-acoustic shocks: The diaphragm problem, *Phys. Fluids* **14**, 1943 (1971).
12. D. W. Forslund, J. M. K. K. Lee, and E. L. Lindman, Absorbtion of laser light on self-consistent plasma density profiles, *Phys. Rev. Lett.* **36**, 35 (1976).
13. A. Friedman, R. L. Ferch, R. N. Sudan, and A. T. Drobot, Numerical simulation of strong proton rings, *Plasma Phys.* **19**, 1101 (1977).
14. J. A. Byers, B. I. Cohen, W. C. Condit, and J. D. Hanson, Hybrid simulation of quazineutral phenomena, *J. Comput. Phys.* **27**, 363 (1978).
15. D. W. Hewett and C. Nielsen, A multidimensional quazineutral plasma simulation model, *J. Comput. Phys.* **29**, 219 (1978).
16. Y. A. Omelchenko and R. N. Sudan, A 3-d Darwin-EM hybrid PIC code for ion ring instability, *J. Comput. Phys.* **133**, 146 (1997).
17. B. I. Cohen, T. A. Brengle, D. B. Conley, and R. P. Freis, An orbit averaged particle code, *J. Comput. Phys.* **38**, 45 (1980).
18. G. W. Hammett, W. Dorland, and F. W. Perkins, Fluid models of phase mixing, Landau damping, and nonlinear gyrokinetic dynamics, *Phys. Fluids B* **4**, 2052 (1992).
19. R. J. Mason, Monte Carlo (hybrid) suprathermal electron transport, *Phys. Rev. Lett.* **43**, 1795 (1979).
20. R. Mason, Monte Carlo hybrid modeling of electron transport in laser produced plasmas, *Phys. Fluids* **23**, 2204 (1980).
21. B. I. Cohen and R. P. Freis, Stability and application of an orbit-averaged magneto-inductive particle code, *J. Comput. Phys.* **45**, 367 (1982).
22. A. Friedman, A. B. Langdon, and B. I. Cohen, A dierct method for implicit Particle-in-Cell simulations, *Comments Plasma Phys. Cont. Fusion* **6**, 225 (1981).
23. A. B. Langdon, B. I. Cohen, and L. Friedman, Direct implicit large time-step particle simulation of plasmas, *J. Comput. Phys.* **51**, 107 (1983).
24. D. W. Hewett and A. B. Langdon, Electromagnetic direct implicit plasma simulation, *J. Comput. Phys.* **72**, 121 (1987).
25. R. J. Mason, Implicit moment particle simulation of plasmas, *J. Comput. Phys.* **41**, 233 (1981).
26. J. Denavit, Time-filtering particle simulations with $\omega_{pe}\Delta t \gg 1$, *J. Comput. Phys.* **42**, 337 (1981).
27. J. M. Wallace, J. U. Brackbill, and D. W. Forslund, An implicit moment electromagnetic plasma simulation in cylindrical coordinates, *J. Comput. Phys.* **63**, 434 (1986).
28. R. J. Mason, P. L. Auer, R. N. Sudan, B. V. Oliver, C. E. Seyler, and J. B. Greenly, Nonlinear magnetic field transport in opening switch plasmas, *Phys. Fluids* **5**, 1115 (1993).
29. B. I. Cohen, R. P. Freis, and V. Thomas, Orbit-averaged implicit particle codes, *J. Comput. Phys.* **45**, 345 (1982).
30. H. Alfvén, *Cosmical Electrodynamics* (Clarendon, Oxford, 1953).
31. W. W. Lee, Gyrokinetic approach in particle simulation, *Phys. Fluids* **26**, 556 (1983).
32. W. W. Lee, Gyrokinetic particle simulation model, *J. Comput. Phys.* **72**, 243 (1987).
33. W. Park, S. Parker, H. Biglari, M. Chance, L. Chen, C. Z. Cheng, T. S. Hahm, W. W. Lee, R. Kulsrud, D. Monticello, L. Sugiyama, and R. White, Three-dimensional hybrid gyrokinetic-magnetohydrodynamics simulation, *Phys. Fluids B* **4**, 2033 (1992).
34. B. I. Cohen, A. M. Dimits, J. J. Stimson, and D. C. Barnes, Implicit-moment, partially linearized particle simulation of kinetic plasma phenomena, *Phys. Rev. E* **53**, 2708 (1996).
35. C. E. Rathmann, J. L. Vomvoridis, and J. Denavit, Long-time-scale simulation of resonant particle effects in Langmuir and whistler waves, *J. Comput. Phys.* **26**, 408 (1978).

36. S. P. Yu, G. P. Kooyers, and O. Buneman, Time-dependent computer analysis of electron-wave interaction in crossed fields, *J. Appl. Phys.* **36**, 2550 (1965).
37. O. Buneman, Time-reversible difference procedures, *J. Comput. Phys.* **1**, 517 (1967).
38. M. S. Grewal and J. A. Byers, A computational study of the negative mass instability in the non-linear regime, *Plasma Phys.* **11**, 727 (1969).
39. O. Buneman, C. W. Barnes, J. C. Green, and D. E. Nielsen, Review: Principles and capabilities of 3-d E-M particle simulations, *J. Comput. Phys.* **38**, 1 (1980).
40. L. D. Landau and E. M. Livshitz, *The Classical Theory of Fields (Course of Theoretical Physics)* (Pergamon, Elmsford, NY, 1975), Vol. 2.
41. *MATLAB, a Language for Technical Computing* (The MathWorks, Inc., `http://www.mathworks.com`).