

On the Boris solver in particle-in-cell simulation

Cite as: Phys. Plasmas **25**, 112110 (2018); <https://doi.org/10.1063/1.5051077>

Submitted: 06 August 2018 . Accepted: 31 October 2018 . Published Online: 16 November 2018

 Seiji Zenitani, and Takayuki Umeda



ARTICLES YOU MAY BE INTERESTED IN

[Why is Boris algorithm so good?](#)

Physics of Plasmas **20**, 084503 (2013); <https://doi.org/10.1063/1.4818428>

[Simulation of beams or plasmas crossing at relativistic velocity](#)

Physics of Plasmas **15**, 056701 (2008); <https://doi.org/10.1063/1.2837054>

[Structure-preserving second-order integration of relativistic charged particle trajectories in electromagnetic fields](#)

Physics of Plasmas **24**, 052104 (2017); <https://doi.org/10.1063/1.4979989>



Physics of Plasmas

Features in Plasma Physics Webinars

Register Today!

On the Boris solver in particle-in-cell simulation

Seiji Zenitani^{1,a)} and Takayuki Umeda²

¹Research Institute for Sustainable Humanosphere, Kyoto University, Gokasho, Uji, Kyoto 611-0011, Japan

²Institute for Space-Earth Environmental Research, Nagoya University, Nagoya, Aichi 464-8601, Japan

(Received 6 August 2018; accepted 31 October 2018; published online 16 November 2018)

A simple form of the Boris solver in particle-in-cell (PIC) simulation is proposed. It employs an exact solution of the Lorentz-force part, and it is equivalent to the Boris solver with a gyrophase correction. As a favorable property for stable schemes, this form preserves a volume in the phase space. Numerical tests of the Boris solvers are conducted by test-particle simulations and by PIC simulations. The proposed form provides better accuracy than the popular form, while it only requires few additional computation time. *Published by AIP Publishing.*

<https://doi.org/10.1063/1.5051077>

I. INTRODUCTION

The particle-in-cell (PIC) method^{1–4} is one of the most important techniques in modern plasma simulations. Solving Lagrangian motions of many charged particles and the temporal evolution of Eulerian electromagnetic fields, it can simulate various kinetic phenomena, whose length scale is larger than the grid size.

The particle integrator, an algorithm to advance charged particles, is a fundamental element of PIC simulation. Since the particle integrator is used for all the particles, its accuracy, stability, and computational cost have a big impact on those of the entire simulation run. One of the most common integrators is the Boris solver,⁵ also known as the Buneman–Boris solver. It solves the particle motion in a leap-frog manner, and the acceleration part is split into several parts, as will be shown later. Owing to its simplicity and reliability, the Boris solver has been used for nearly 50 years.

In addition to the Boris solver, various solvers have been developed. For earlier ones, we refer the readers to classic textbooks^{1,2} and references therein. Recently, particle solvers have been developed much more actively than before.^{6–13} Vay⁶ has developed a solver to carefully deal with the force balance. By splitting the integrator into the explicit first half and the implicit second half, his solver better deals with the $\mathbf{E} \times \mathbf{B}$ drift at a relativistic speed, and therefore, it has drawn growing attention in laser physics and in astrophysics. Pétri⁷ has proposed a relativistic implicit solver, which iteratively uses a matrix-form of the Vay solver for non-staggered timesteps. Qiang⁸ has presented a fast Runge–Kutta relativistic integrator, ready for the force balance problem. Umeda⁹ has proposed a multi-step extension of the Boris integrator. His solver effectively deals with the gyration at a half timestep.

From the theoretical viewpoint, Qin *et al.*¹⁰ have recently pointed out that the nonrelativistic Boris solver is stable because it preserves a volume in the phase space every timestep. This property can be examined by a simple Jacobian matrix. Presently, the volume preservation is regarded as a key property for stable solvers. Zhang *et al.*¹¹

split the scheme into several substeps to discuss the volume preservation of their solver, which appears to be another expression of the Boris solver. Higuera and Cary¹² have proposed a relativistic volume-preserving solver, which employs Vay's characteristic velocity in the Lorentz-force part of the Boris solver. Ripperda *et al.*¹³ extensively compared a selected set of particle solvers.

In this contribution, we propose a simple form of the Boris solver, based on an exact solution for the Lorentz-force part. We further examine the volume preservation of the proposed form of the Boris solver. Then, we will present numerical tests of the Boris solvers, followed by discussion and summary.

II. BORIS SOLVER

First, we outline the Boris algorithm.⁵ It handles the particle motion in the following discrete forms:

$$\frac{\mathbf{x}^{n+\frac{1}{2}} - \mathbf{x}^{n-\frac{1}{2}}}{\Delta t} = \frac{\mathbf{u}^n}{\gamma^n}, \quad (1)$$

$$m \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = q(\mathbf{E}^{n+\frac{1}{2}} + \bar{\mathbf{v}}^{n+\frac{1}{2}} \times \mathbf{B}^{n+\frac{1}{2}}), \quad (2)$$

where the superscripts ($n, n+1, \dots$) indicate timesteps, $\mathbf{u} = \gamma \mathbf{v}$ is the spatial part of a 4-vector, $\gamma = [1 - (v/c)^2]^{-1/2} = [1 + (u/c)^2]^{1/2}$ is the Lorentz factor, and $\bar{\mathbf{v}}$ is an effective velocity. Other symbols have their standard meanings. The action part is split into the Coulomb force for the first half timestep, the Lorentz force for the entire timestep, and the Coulomb force for the second half

$$\mathbf{u}^- = \mathbf{u}^n + \varepsilon \Delta t, \quad (3)$$

$$\frac{\mathbf{u}^+ - \mathbf{u}^-}{\Delta t} = \frac{q}{m} (\bar{\mathbf{v}}^{n+\frac{1}{2}} \times \mathbf{B}^{n+\frac{1}{2}}), \quad (4)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^+ + \varepsilon \Delta t, \quad (5)$$

where $\varepsilon = (q/2m)\mathbf{E}^{n+\frac{1}{2}}$, \mathbf{u}^- and \mathbf{u}^+ are two intermediate states. Hereafter, we denote the field quantities $\mathbf{E}^{n+\frac{1}{2}}$ and $\mathbf{B}^{n+\frac{1}{2}}$ at $n = n + \frac{1}{2}$ as \mathbf{E} and \mathbf{B} for brevity. Since Eq. (4) is an energy-conserving rotation in the momentum space, the

^{a)}Email: zenitani@rish.kyoto-u.ac.jp

Lorentz factor is set to be constant during the operation, $\gamma^- = \gamma^+$. The phase angle in the rotation part is

$$\theta = \frac{q\Delta t}{m\gamma^-} B. \quad (6)$$

The rotation is solved in the following ways:

$$\mathbf{t} = \tan \frac{\theta}{2} \mathbf{b}, \quad (7a)$$

$$\mathbf{t} = \frac{\theta}{2} \mathbf{b} = \frac{q\Delta t}{2m\gamma^-} \mathbf{B}, \quad (7b)$$

$$\mathbf{u}' = \mathbf{u}^- + \mathbf{u}^- \times \mathbf{t}, \quad (8)$$

$$\mathbf{u}^+ = \mathbf{u}^- + \frac{2}{1+t^2} (\mathbf{u}' \times \mathbf{t}), \quad (9)$$

where $\mathbf{b} = \mathbf{B}/|\mathbf{B}|$ is a unit vector. There are two choices in Eq. (7). One can advance \mathbf{u}^n to \mathbf{u}^{n+1} by using either of the two equation sets. Boris⁵ presented a procedure of Eqs. (3), (6), (7a), (8), (9), and (5) in his original article. The subsequent textbooks^{1,2} described a simplified procedure [Eqs. (3), (7b), (8), (9), and (5)], by replacing Eq. (7a) with Eq. (7b). We call the two procedures the Boris-A solver and the Boris-B solver, respectively.

Although we do not describe in detail, the Boris-A solver accurately handles the rotation. As a result of the replacement [Eq. (7a) \rightarrow Eq. (7b)], the Boris-B solver approximates the rotation, as illustrated in gray in Fig. 1. There is always a delay in the gyrophase angle²

$$\delta\theta = \theta - 2\arctan \frac{\theta}{2} = \theta \left(\frac{1}{12} \theta^2 - \frac{1}{80} \theta^4 + \dots \right). \quad (10)$$

Thus, the Boris-B solver is an approximate form of the original Boris solver (the Boris-A solver). Despite this, owing to its simplicity and computational cost, the Boris-B solver is widely used. Scientists often indicate the Boris-B solver simply by “the Boris solver.” Equations (7a) and (7b) differ by a factor of $\tan(\frac{\theta}{2})/(\frac{\theta}{2})$. Therefore, the Boris-A solver is

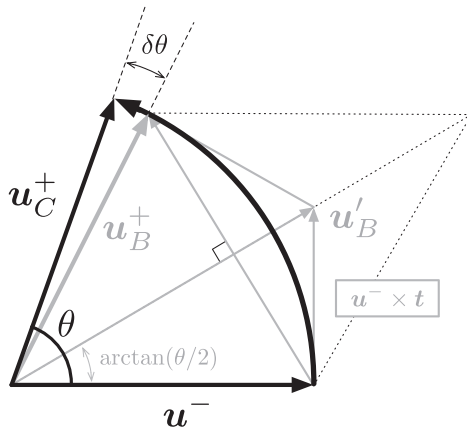


FIG. 1. Schematic diagram for the Lorentz-force part of the Boris solvers. Procedures by the Boris-C solver (in black) and by the Boris-B solver (in gray) are illustrated. There is an error in the phase angle ($\delta\theta$) in the Boris-B solver.

sometimes referred to as the Boris solver with a gyrophase correction factor.^{1,2}

III. EXACT GYRATION SOLVER

To avoid the phase error in the Boris-B solver, one can use the Boris-A solver. Here, we propose another expression. We propose the following rotation procedure based on an analytic solution:

$$\mathbf{u}_{\parallel} = (\mathbf{u}^- \cdot \mathbf{b}) \mathbf{b}, \quad (11)$$

$$\mathbf{u}^+ = \mathbf{u}_{\parallel} + (\mathbf{u}^- - \mathbf{u}_{\parallel}) \cos \theta + (\mathbf{u}^- \times \mathbf{b}) \sin \theta. \quad (12)$$

This is illustrated in black in Fig. 1. Then, one can advance the particle by Eqs. (3), (6), (11), (12), and (5) in a time-reversible manner. We call this the Boris-C solver. In practice, we use a threshold ϵ_b in $|\mathbf{B}|^2 = \max(\mathbf{B}^2, \epsilon_b)$, to avoid dividing by zero. This does not cause a significant error in the $B \rightarrow 0$ limit.

The Boris-C solver provides second-order accuracy. The splitting into the Coulomb-force part [Eqs. (3) and (5)] and the Lorentz-force part [Eqs. (11) and (12)] is equivalent to an operator splitting, also known as the Strang splitting.^{14,15} The Strang splitting gives second-order accuracy if each operator maintains more than second-order accuracy. In this case, since both of the two parts give exact solutions, the resulting scheme should have second-order accuracy.

One can also prove the second-order accuracy in a straightforward manner. In the constant fields, the second-order expansion of the new state \mathbf{u}^{n+1} is

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \dot{\mathbf{u}}^n \Delta t + \frac{1}{2} \ddot{\mathbf{u}}^n \Delta t^2 + \mathcal{O}(\Delta t^3), \quad (13)$$

where

$$\dot{\mathbf{u}} = \frac{q}{m} (\mathbf{E} + \mathbf{v} \times \mathbf{B}), \quad (14)$$

$$\begin{aligned} \ddot{\mathbf{u}} &= \frac{q}{m} (\dot{\mathbf{v}} \times \mathbf{B}) = \frac{q}{m} \left(\frac{\dot{\mathbf{u}}}{\gamma} - \frac{\mathbf{u} \cdot \dot{\mathbf{u}}}{\gamma^3 c^2} \mathbf{u} \right) \times \mathbf{B} \\ &= \left(\frac{q}{m} \right)^2 \left(\frac{\mathbf{E} + \mathbf{v} \times \mathbf{B}}{\gamma} - \frac{\mathbf{u} \cdot \mathbf{E}}{\gamma^3 c^2} \mathbf{u} \right) \times \mathbf{B}. \end{aligned} \quad (15)$$

Following the Boris procedure, we find

$$\mathbf{u}^- = \mathbf{u}^n + \mathbf{E} \Delta t, \quad (16)$$

$$\begin{aligned} \mathbf{u}^+ &= \mathbf{u}^- + \frac{q}{m} (\mathbf{v}^- \times \mathbf{B}) \Delta t \\ &\quad + \frac{1}{2} \left\{ \left(\frac{q}{m} \right)^2 \left(\frac{\mathbf{v}^- \times \mathbf{B}}{\gamma^-} \right) \times \mathbf{B} \right\} \Delta t^2 + \mathcal{O}(\Delta t^3) + \dots, \end{aligned} \quad (17)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^+ + \mathbf{E} \Delta t. \quad (18)$$

In Eq. (17), the Lorentz-force part is expanded for \mathbf{u}^- , similar to Eqs. (13)–(15) but with $\mathbf{E} = 0$. Although the Boris-C solver accurately solves the third and higher-order terms in Eq. (17), we focus on the terms up to second-order. Importantly, Eqs. (16) and (18) do not contain second or

higher-order terms, $\mathcal{O}(\Delta t^2)$, because the solver already gives an exact solution for the Coulomb-force part. From Eq. (16), we obtain first-order expansions of the following variables:

$$\begin{aligned} v^- &= \frac{1}{\gamma} \left(\mathbf{u}^n + \frac{q}{2m} \mathbf{E} \Delta t \right) \left(1 + \frac{q \mathbf{u}^n \cdot \mathbf{E}}{mc^2 \gamma^2} \Delta t + \mathcal{O}(\Delta t^2) \right)^{-1/2} \\ &= \mathbf{v}^n + \frac{q}{2m\gamma} \left(\mathbf{E} - \frac{\mathbf{u}^n \cdot \mathbf{E}}{c^2 \gamma^2} \mathbf{u}^n \right) \Delta t + \mathcal{O}(\Delta t^2), \end{aligned} \quad (19)$$

$$\frac{v^-}{\gamma^-} = \frac{v^n}{\gamma} + \frac{q}{2m\gamma} \left(\mathbf{E} - \frac{2\mathbf{u}^n \cdot \mathbf{E}}{c^2 \gamma^2} \mathbf{u}^n \right) \Delta t + \mathcal{O}(\Delta t^2). \quad (20)$$

Substituting Eqs. (19) and (20) into Eqs. (16)–(18), one can obtain an expanded form of the Boris-C solver, $\mathbf{u}^{n+1} = \mathbf{u}^n + \dots + \mathcal{O}(\Delta t^3)$. The first and second-order coefficients are identical to those of the Talyor expansion [Eq. (13)–(15)] for $\mathbf{u} = \mathbf{u}^n$. This proves that the Boris-C solver has second-order accuracy.

As a theoretical property, we examine the volume preservation¹⁰ of the Boris-C solver. This discussion holds true for the Boris-A solver as well because both methods accurately solve the rotation. We consider the temporal evolution of a phase-space volume by splitting the Boris-C solver to the following substeps.¹¹ We consider half-adjusted positions \mathbf{x}^n and \mathbf{x}^{n+1} . R stands for a 3-D rotation matrix

$$1: \quad \mathbf{x}^{n+\frac{1}{2}} = \mathbf{x}^n + \frac{\Delta t}{2} \frac{\mathbf{u}^n}{\gamma^n}, \quad \mathbf{u}^n = \mathbf{u}^n, \quad (21)$$

$$2: \quad \mathbf{x}^{n+\frac{1}{2}} = \mathbf{x}^{n+\frac{1}{2}}, \quad \mathbf{u}^- = \mathbf{u}^n + \varepsilon \Delta t, \quad (22)$$

$$3: \quad \mathbf{x}^{n+\frac{1}{2}} = \mathbf{x}^{n+\frac{1}{2}}, \quad \mathbf{u}^+ = R \mathbf{u}^-, \quad (23)$$

$$4: \quad \mathbf{x}^{n+\frac{1}{2}} = \mathbf{x}^{n+\frac{1}{2}}, \quad \mathbf{u}^{n+1} = \mathbf{u}^+ + \varepsilon \Delta t, \quad (24)$$

$$5: \quad \mathbf{x}^{n+1} = \mathbf{x}^{n+\frac{1}{2}} + \frac{\Delta t}{2} \frac{\mathbf{u}^{n+1}}{\gamma^{n+1}}, \quad \mathbf{u}^{n+1} = \mathbf{u}^{n+1}. \quad (25)$$

We evaluate the Jacobian J_k for the k -th substep

$$J_k = \left| \frac{\partial(\mathbf{x}^{new}, \mathbf{u}^{new})}{\partial(\mathbf{x}^{old}, \mathbf{u}^{old})} \right| = \det \begin{pmatrix} \frac{\partial \mathbf{x}^{new}}{\partial \mathbf{x}^{old}} & \frac{\partial \mathbf{x}^{new}}{\partial \mathbf{u}^{old}} \\ \frac{\partial \mathbf{u}^{new}}{\partial \mathbf{x}^{old}} & \frac{\partial \mathbf{u}^{new}}{\partial \mathbf{u}^{old}} \end{pmatrix}. \quad (26)$$

From Eqs. (21)–(25), we obtain

$$J_1 = J_5 = \left| \begin{array}{cc} I & \frac{\partial \mathbf{x}^{new}}{\partial \mathbf{u}^{old}} \\ 0 & I \end{array} \right| = 1, \quad J_2 = J_4 = \left| \begin{array}{cc} I & 0 \\ \frac{\partial \mathbf{u}^{new}}{\partial \mathbf{x}^{old}} & I \end{array} \right| = 1, \quad (27)$$

$$J_3 = \left| \begin{array}{cc} I & 0 \\ \frac{\partial \mathbf{u}^{new}}{\partial \mathbf{x}^{old}} & \frac{\partial \mathbf{u}^{new}}{\partial \mathbf{u}^{old}} \end{array} \right| = \left| \frac{\partial \mathbf{u}^{new}}{\partial \mathbf{u}^{old}} \right|. \quad (28)$$

The third step [Eqs. (23) and (28)] corresponds to a γ -dependent rotation, i.e., $\theta \propto \gamma^{-1}$. In this case, we confirm $J_3 = |\partial(\mathbf{u}^{new})/\partial(\mathbf{u}^{old})| = 1$ as shown in Appendix A. Thus,

we obtain $J_k = 1$ for $k = 1 \dots 5$. This indicates that the solver preserves a volume in the phase space every substep. Therefore, the Boris-C solver is volume-preserving during the entire step from $(\mathbf{x}^n, \mathbf{v}^n)$ to $(\mathbf{x}^{n+1}, \mathbf{v}^{n+1})$.

For comparison, we examine the volume preservation for the nonrelativistic fourth-order Runge–Kutta method in Appendix B. The phase-space volume is not preserved in this case, $J_{RK4} \neq 1$.

IV. NUMERICAL TESTS

In order to see the accuracy, stability, and performance of the Boris solvers, we have carried out four numerical tests. First, to check the accuracy, we have carried out test-particle simulations in a uniform electromagnetic field. Physics parameters are set to $m=1$, $q=1$, $c=1$, and $\mathbf{u}(t=0) = (1, 0, 0)$. The time interval, the timestep, and the numerical threshold are set to $0 < t < 12/\pi$, $\Delta t = \pi/6$, and $\epsilon_b = 10^{-20}$. We consider five configurations, as shown in Table I. The first case corresponds to direct acceleration by the electric field. A weak magnetic field is imposed in the second case, but the electric field is still dominant. The third is a special case of $\mathbf{E} \perp \mathbf{B}$ and $|\mathbf{E}| = |\mathbf{B}|$. The fourth case corresponds to the $\mathbf{E} \times \mathbf{B}$ drift. The drift is slightly modulated by the relativistic effect. The last case corresponds to gyration about the magnetic field. In each cases, we evaluate an error in 4-vector to a reference value, $\delta \mathbf{u} = \mathbf{u} - \mathbf{u}_{ref}$. For cases 1 and 5, we employ analytic solutions as reference values. For cases 2–4, we refer numerical results with a small timestep $\Delta t = \pi/240$ because we do not know analytic solutions as a simple function of t . The numerical reference values are checked by analytic solutions in other forms.^{16,17}

Figure 2(a) shows temporal evolution of errors $|\delta \mathbf{u}|/|\mathbf{u}|$ in our test-particle simulations. The dotted lines represent the results by the Boris-B solver, whereas the solid lines represent the results by the Boris-C solver. The results by the Boris-A solver are not shown because they are essentially the same as those by the Boris-C solver. In case 1 (in red), the Boris-B and Boris-C solvers use the same parts [Eqs. (3) and (5)], and their results are identical. The two solvers give accurate results because Eqs. (3) and (5) give an exact solution for the linear acceleration by \mathbf{E} . In cases 2 and 3 (in magenta and in orange), the two solvers give similar results. The curves drop because $|\delta \mathbf{u}|$ does not grow and $|\mathbf{u}|$ increases in time. In case 4 of the $\mathbf{E} \times \mathbf{B}$ drift (in green), the Boris-C solver drastically improves the results. It reduces the error by two orders-of-magnitude, as indicated by the green arrow. The error by the Boris-B solver grows in time because the

TABLE I. Field settings for test-particle simulations.

#	\mathbf{B}	\mathbf{E}	
1	(0,0,0)	(1,0,0)	Direct acceleration by \mathbf{E}
2	(0,0,0.1)	(1,0,0)	\mathbf{E} -dominated
3	(0,0,1)	(1,0,0)	$ \mathbf{E} = \mathbf{B} $
4	(0,0,1)	(0.1,0,0)	$\mathbf{E} \times \mathbf{B}$ drift
5	(0,0,1)	(0,0,0)	Gyration about \mathbf{B}

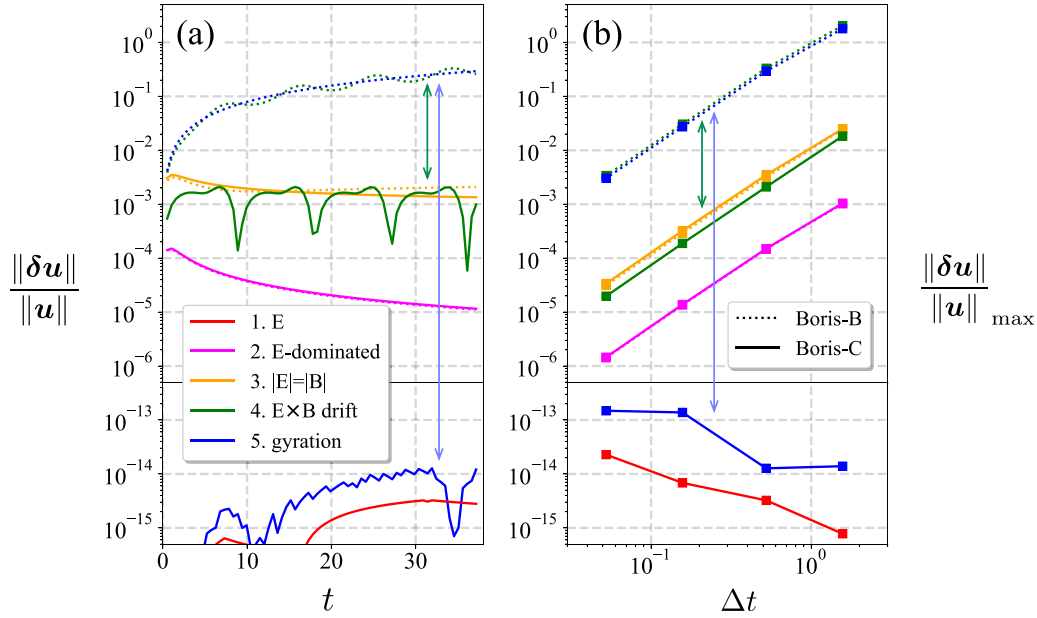


FIG. 2. (a) Time evolution of numerical errors $|\delta \mathbf{u}|/|\mathbf{u}|$ in test-particle simulations. The dotted lines indicate the results by the Boris-B solver, whereas the solid lines indicate the results by the Boris-C solver. The timestep is fixed to $\Delta t = \pi/6$. (b) Maximum errors $(|\delta \mathbf{u}|/|\mathbf{u}|)_{\max}$ during $0 < t < 12\pi$ as a function of Δt .

phase error $|\delta \mathbf{u}| \approx |\mathbf{u}| \delta \theta = \delta \theta$ accumulates. On the other hand, the error by the Boris-C solver remains small. The repeated drop corresponds to the gyroperiod. Since the Boris-C solver exactly solves the phase, the solid curve repeatedly shows the same pattern, and it does not grow further. In case 5 of gyration (in blue), the error by the Boris-B solver linearly grows in time ($\propto t$) because of the accumulation of the phase error. In contrast, the Boris-C solver gives accurate results, as we have expected.

Figure 2(b) presents maximum errors $(|\delta \mathbf{u}|/|\mathbf{u}|)_{\max}$ during $0 < t < 12\pi$ as a function of the timestep, $\Delta t = \pi/60, \pi/20, \pi/6$, and $\pi/2$. The results demonstrate the second-order accuracy of the solvers. As evident in cases 1–3, the Boris-C solver is as good as the Boris-B solver when $|E| \gtrsim |B|$. As the electric field dominates, the errors decrease, and then, they give accurate results in case 1 (red line). When $|B| \gtrsim |E|$, the two solvers give different results. In cases 4 and 5, the Boris-B solver remains moderately good (the green and blue dotted lines). From Eq. (10) and using $\gamma = \sqrt{2}$, one can estimate the curve in case 5, $|\delta \mathbf{u}|/|\mathbf{u}| \approx (12\pi/\Delta t) \delta \theta = (\pi/\sqrt{8}) \Delta t^2$, in agreement with the blue dotted line. In contrast, the Boris-C solver produces drastically smaller errors than the Boris-B solver, even though it maintains the second-order accuracy. As the magnetic field dominates, the errors decrease, and then, the solver gives accurate results in case 5 (blue solid line). In the bottom part in Fig. 2(b), the errors in the two exact cases probably come from a machine error. For example, for case 5, if an error of $\mathcal{O}(10^{-15})$ in double precision floating numbers is accumulated every timestep, the total error would be $|\delta \mathbf{u}|/|\mathbf{u}| \leq 10^{-15} \times (12\pi/\Delta t) \approx 10^{-13.5} \Delta t^{-1}$, in consistent with the blue line. The error in case 1 (red line) is even smaller because of larger $|\mathbf{u}|$.

As a second numerical test, we evaluate the long-term stability of the particle solvers. We have run a code in the following field, as was done in Ref. 10:

$$\mathbf{B} = (x^2 + y^2)^{1/2} \mathbf{e}_z, \quad \phi = 0.01(x^2 + y^2)^{-1/2}. \quad (29)$$

The initial conditions are $\mathbf{u}(t=0) = (0.1, 0.0)$, $\mathbf{x}(t=0) = (0.9, 0.0)$, $m = c = 1$, and $\Delta t = \pi/10$. Figure 3 shows the trajectories at (a) an initial stage and at (b) a late stage by the Boris-C solver (in blue) and by the fourth order Runge–Kutta solver (in gray). From the beginning, the particle undergoes a fast small-scale gyration and a slow large-scale rotation due to the ∇B drift and the $\mathbf{E} \times \mathbf{B}$ drift. The large-scale drifts keep the particle in the same domain, and then, we inspect the evolution of the trajectory. After a long time (300th turn; $t \approx 2 \times 10^5$), as evident in Fig. 3(b), the Runge–Kutta solver dissipates the small-scale motion and the relevant kinetic energy. This is because the Runge–Kutta solver is not volume-preserving, at least in the uniform fields in the nonrelativistic regime (Appendix B), and quite likely so in generic relativistic cases. In contrast, the Boris-C solver is free from the numerical damping, similar to the Boris-B solver (not shown). This numerical experiment demonstrates that the Boris-C solver does preserve the phase-space volume for the small-scale gyration over a long time.

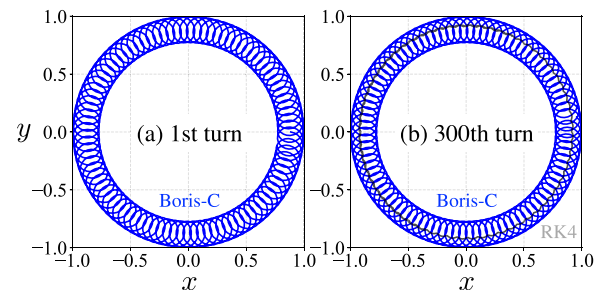


FIG. 3. Particle trajectories by the Boris-C solver during (a) the initial stage and (b) the late stage. A trajectory by the RK4 solver is overplotted in gray in (b) the late stage.

As a third numerical test, to compare computational costs, we have run case 5 in Table I until $t = 2 \times 10^7 \pi$ (1.2×10^8 steps) by only advancing the particle velocity. Average elapse times are proportional to 1.46:1:1.24 for the Boris-A, B, and C solvers on our PC (Intel processor) and 1.46:1:1.26 on FX100 computer (SPARC processor) at Japan Aerospace Exploration Agency (JAXA). While performance depends on implementation, compilers, and CPU architectures, the Boris-C solver runs faster than the Boris-A solver, and it runs only 25% slower than the Boris-B solver.

As a final test, we have carried out PIC simulations of a physics problem on magnetic reconnection. The settings are documented in Ref. 18, and therefore, we do not repeat them here. Comparing average elapse times, we have found that the Boris-C solver is slower than the Boris-B solver by 1.2% on the XC50 supercomputer (Intel processor) at the National Astronomical Observatory of Japan and by 3.3% on the FX100 computer (SPARC processor) at JAXA.

V. DISCUSSION AND SUMMARY

As already discussed, the Boris-C solver is a combination of the two exact solvers for the Coulomb-force part and for the Lorentz-force part, while the Boris-B solver is based on the exact Coulomb-force solver and a second-order solver for the Lorentz-force part. From the viewpoint of operator-splitting, in both cases, the combined solver maintains second-order accuracy because each part has second-order accuracy. This is evident in Fig. 2(b). Since the Boris-C solver better deals with the Lorentz-force part, the amplitude of the second-order error is much smaller than in the Boris-B solver, in particular in the $|B| \geq |E|$ cases. The Boris-B solver gives a second-order error in phase [Eq. (10)]. Other second-order solvers, such as the Higuera–Cary solver and the Umeda solver, could suppress the total error by better solving the phase.^{9,12,13} In this line, the Boris-C and A solvers provide the best possible results because they give no phase error in phase.

We have formally proved that the Boris-C solver preserves a volume in the phase space during the temporal evolution, and then, we have confirmed that it preserves the small-scale gyration after a long-time computation. The volume preservation of the popular Boris-B solver has been extensively studied;^{10–12} however, it has never been evaluated in the Boris-A or C solvers. Since the Boris-C solver is simple, the proof is given straightforwardly. This gives further confidence to the reliability of the Boris-C and A solvers.

Many scientists prefer the Boris-B solver to the Boris-A solver because of the low computation cost. In fact, according to the numerical test, the Boris-A solver is 46% more expensive than the Boris-B solver. Since the Boris-C solver employs simple expressions, it is more favorable than the Boris-A solver. The Boris-C solver gives 25% in test-particle simulations. Obviously, the conventional solvers (the Boris-A and Boris-B solvers) tried to avoid trigonometric functions which were expensive at that time. Presently, these functions are not so expensive as they used to be, and therefore, the Boris-C solver runs adequately fast. In PIC simulations, the

Boris-C solver runs slower only by 1%–3% in PIC simulations. Typically, the particle integrator and the electric current calculator account for most of the computation time, and they consume an equal amount of time. Then, largely due to memory-access performance, the particle integrator does not run at full speed. If we assume that it runs at 50% efficiency, the particle integrator is responsible for 25% of the computation time. Then, the 25% slow-down in the particle integrator should result in 6% slow-down in the entire run. We have observed 1%–3% in our PIC simulations, probably because our code runs less efficiently, but the results are reasonable. The computational cost can be easily compensated by employing a larger Δt .

From the viewpoint of the stability, we usually keep the timestep small, $\omega_{\max} \Delta t < 2$, where ω_{\max} is the maximum frequency to resolve.² Nevertheless, in PIC simulation, the magnitude of the magnetic field may instantly approach or exceed the $\frac{qB}{\gamma mc} \Delta t = 2$ criteria, and so, it is useful to check the stability for a large Δt limit. The Boris-B solver delays the gyrophase [Eq. (10)], and then, the angle never exceeds π , i.e., $\theta < \pi$. The particle eventually moves back-and-forth.¹⁹ The Boris-A and Boris-C solvers have no such limitation. They simply allow particles to gyrate more than π , although it leads to an opposite gyration. Meanwhile, the Boris-A solver needs some care for $\theta \approx \pi$, where Eq. (7a) is undefined or approaches $\pm\infty$. Therefore, the Boris-B and Boris-C solvers are safer choices among the three.

Finally, we note that there has been continual progress in developing relativistic symplectic solvers.^{20–23} In fact, the symplectic algorithms^{24–26} are favored in many applications, owing to their long-term accuracy. However, the symplectic schemes are often implicit and computationally expensive, while the number of explicit symplectic solvers for relativistic charged particles in arbitrary electromagnetic fields is limited.^{22,23} In addition, it was reported that volume-preserving solvers are sometimes more accurate than symplectic solvers.²⁷ Considering these issues, it will take some time before symplectic solvers will be popular in relativistic PIC simulations. Meantime, it is important to improve the second-order Boris solvers, which have an advantage in computational efficiency and which are proven to be reliable.

In summary, we have proposed a simple form of the Boris solver, based on the exact solution for the Lorentz-force part. It is equivalent to the Boris solver with a gyro-phase correction. It has a favorable property of preserving the phase-space volume, and therefore, it appears to be stable. The proposed form gives much more accurate results than the popular form (the Boris-B solver), while it only requires few additional computation time. We hope that the proposed form will be a good alternative to the conventional Boris solvers.

ACKNOWLEDGMENTS

The authors acknowledge the anonymous reviewer for his/her professional remarks. One of the authors (S.Z.) acknowledges T.-N. Kato, Y. Omura, and T. Nogi for useful comments. This work was supported by Grant-in-Aid for

Scientific Research (C) 17K05673 and (S) 17H06140 from the Japan Society for the Promotion of Science (JSPS).

APPENDIX A: JACOBIAN FOR RELATIVISTIC GYRATION

Without losing generality, we consider a gyration about $\mathbf{B} = (0, 0, B)$. Then, the rotation procedure is expressed as

$$\mathbf{u}^{new} = R\mathbf{u} = \begin{pmatrix} u_x \cos \frac{\theta_0}{\gamma} + u_y \sin \frac{\theta_0}{\gamma} \\ -u_x \sin \frac{\theta_0}{\gamma} + u_y \cos \frac{\theta_0}{\gamma} \\ u_z \end{pmatrix}, \quad (\text{A1})$$

where $\theta_0 = (qB\Delta t)/m$. Considering

$$\frac{\partial}{\partial u_x} \left(\frac{\theta_0}{\gamma} \right) = \theta_0 \frac{\partial}{\partial u_x} (1 + (u/c)^2)^{-\frac{1}{2}} = -\frac{\theta_0}{\gamma^3 c^2} u_x, \quad (\text{A2})$$

we obtain a Jacobian matrix for $\mathbf{u} \rightarrow \mathbf{u}^{new}$

$$\frac{\partial(\mathbf{u}^{new})}{\partial(\mathbf{u})} = \begin{pmatrix} \cos \frac{\theta_0}{\gamma} - f_0 f_1 u_x & \sin \frac{\theta_0}{\gamma} - f_0 f_1 u_y & -f_0 f_1 u_z \\ -\sin \frac{\theta_0}{\gamma} - f_0 f_2 u_x & \cos \frac{\theta_0}{\gamma} - f_0 f_2 u_y & -f_0 f_2 u_z \\ 0 & 0 & 1 \end{pmatrix}, \quad (\text{A3})$$

where

$$\begin{cases} f_0 = \frac{\theta_0}{\gamma^3 c^2}, \\ f_1 = -u_x \sin \frac{\theta_0}{\gamma} + u_y \cos \frac{\theta_0}{\gamma}, \\ f_2 = -u_x \cos \frac{\theta_0}{\gamma} - u_y \sin \frac{\theta_0}{\gamma}. \end{cases} \quad (\text{A4})$$

From Eqs. (A3) and (A4), some algebra leads to

$$\left| \frac{\partial(\mathbf{u}^{new})}{\partial(\mathbf{u})} \right| = 1. \quad (\text{A5})$$

APPENDIX B: VOLUME PRESERVATION FOR THE RUNGE–KUTTA METHOD

Here, we focus on a simple case of nonrelativistic particle motion in a uniform magnetic field. The fourth-order Runge–Kutta method advances a particle in the following way:

$$\mathbf{v}^{n+1} = \mathbf{v}_n + \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)\Delta t, \quad (\text{B1})$$

$$\mathbf{k}_1 = \mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon}, \quad (\text{B2})$$

$$\mathbf{k}_2 = \left(\mathbf{v}_n + \frac{1}{2}\mathbf{k}_1\Delta t \right) \times \boldsymbol{\omega} + \boldsymbol{\epsilon}, \quad (\text{B3})$$

$$\mathbf{k}_3 = \left(\mathbf{v}_n + \frac{1}{2}\mathbf{k}_2\Delta t \right) \times \boldsymbol{\omega} + \boldsymbol{\epsilon}, \quad (\text{B4})$$

$$\mathbf{k}_4 = (\mathbf{v}_n + \mathbf{k}_3\Delta t) \times \boldsymbol{\omega} + \boldsymbol{\epsilon}, \quad (\text{B5})$$

where $\boldsymbol{\omega} = q\mathbf{B}/m$ and $\boldsymbol{\epsilon} = q\mathbf{E}/m$. We obtain new states

$$\begin{aligned} \mathbf{v}^{n+1} = & \mathbf{v}_n + (\mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon})\Delta t + \frac{1}{2}((\mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon}) \times \boldsymbol{\omega})\Delta t^2 \\ & + \frac{1}{3!}(((\mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon}) \times \boldsymbol{\omega}) \times \boldsymbol{\omega})\Delta t^3 \\ & + \frac{1}{4!}((((\mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon}) \times \boldsymbol{\omega}) \times \boldsymbol{\omega}) \times \boldsymbol{\omega})\Delta t^4, \end{aligned} \quad (\text{B6})$$

$$\begin{aligned} \mathbf{x}^{n+1} = & \mathbf{x}_n + \mathbf{v}_n\Delta t + \frac{1}{2}(\mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon})\Delta t^2 \\ & + \frac{1}{3!}((\mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon}) \times \boldsymbol{\omega})\Delta t^3 \\ & + \frac{1}{4!}(((\mathbf{v}_n \times \boldsymbol{\omega} + \boldsymbol{\epsilon}) \times \boldsymbol{\omega}) \times \boldsymbol{\omega})\Delta t^4. \end{aligned} \quad (\text{B7})$$

In a uniform field, the Jacobian for the Runge–Kutta method is

$$J_{\text{RK4}} = \left| \frac{\partial(\mathbf{x}^{n+1}, \mathbf{v}^{n+1})}{\partial(\mathbf{x}^n, \mathbf{v}^n)} \right| = \begin{vmatrix} I & \frac{\partial \mathbf{x}^{n+1}}{\partial \mathbf{v}^n} \\ 0 & \frac{\partial \mathbf{v}^{n+1}}{\partial \mathbf{v}^n} \end{vmatrix} = \left| \frac{\partial \mathbf{v}^{n+1}}{\partial \mathbf{v}^n} \right|. \quad (\text{B8})$$

Without losing generality, we consider the case of $\mathbf{B} = (0, 0, B)$. Then, the Jacobian J_{RK4} is given by

$$\begin{vmatrix} 1 - \frac{1}{2}(\omega\Delta t)^2 + \frac{1}{24}(\omega\Delta t)^4 & -(\omega\Delta t) + \frac{1}{6}(\omega\Delta t)^3 & 0 \\ (\omega\Delta t) - \frac{1}{6}(\omega\Delta t)^3 & 1 - \frac{1}{2}(\omega\Delta t)^2 + \frac{1}{24}(\omega\Delta t)^4 & 0 \\ 0 & 0 & 1 \end{vmatrix} = 1 - \frac{1}{72}(\omega\Delta t)^6 + \frac{1}{576}(\omega\Delta t)^8. \quad (\text{B9})$$

In the parameter range of our interest, $(\omega\Delta t) < 1$, one can see that the phase-space volume shrinks: $J_{\text{RK4}} < 1$.

¹R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (McGraw-Hill, New York, 1981).

²C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation* (McGraw-Hill, New York, 1985).

³J. M. Dawson, “Particle simulation of plasmas,” *Rev. Mod. Phys.* **55**, 403 (1983).

⁴J. P. Verboncoeur, “Particle simulation of plasmas: Review and advances,” *Plasma Phys. Controlled Fusion* **47**, A231 (2005).

⁵J. P. Boris, “Relativistic plasma simulation—Optimization of a hybrid code,” in *Proceedings of 4th Conference on Numerical Simulation of Plasmas* (Naval Research Laboratory, Washington D. C., 1970), pp. 3–67.

⁶J. L. Vay, “Simulation of beams or plasmas crossing at relativistic velocity,” *Phys. Plasmas* **15**, 056701 (2008).

⁷J. Pétri, “A fully implicit numerical integration of the relativistic particle equation of motion,” *J. Plasma Phys.* **83**, 705830206 (2017).

⁸J. Qiang, “A fast numerical integrator for relativistic charged particle tracking,” *Nucl. Instrum. Methods Phys. Res., A* **867**, 15 (2017).

⁹T. Umeda, “A three-step Boris integrator for Lorentz force equation of charged particles,” *Comput. Phys. Commun.* **228**, 1 (2018).

¹⁰H. Qin, S. Zhang, J. Xiao, J. Liu, Y. Sun, and W. M. Tang, “Why is Boris algorithm so good?,” *Phys. Plasmas* **20**, 084503 (2013).

¹¹R. Zhang, J. Liu, J. Liu, H. Qin, Y. Wang, Y. He, and Y. Sun, “Volume-preserving algorithm for secular relativistic dynamics of charged particles,” *Phys. Plasmas* **22**, 044501 (2015).

- ¹²A. V. Higuera and J. R. Cary, "A fully implicit numerical integration of the relativistic particle equation of motion," *Phys. Plasmas* **24**, 052104 (2017).
- ¹³B. Ripperda, F. Bacchini, J. Teunissen, C. Xia, O. Porth, L. Sironi, G. Lapenta, and R. Keppens, "A comprehensive comparison of relativistic particle integrators," *Astrophys. J. Suppl. Series* **235**, 21 (2018).
- ¹⁴G. Strang, "On the construction and comparison of difference schemes," *SIAM J. Numer. Anal.* **5**, 506 (1968).
- ¹⁵R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems* (Cambridge University Press, 2002).
- ¹⁶L. D. Landau and E. M. Lifshitz, *The Classical Theory of Fields*, 3rd ed. (Pergamon Press, Oxford, 1975), pp. 58–59.
- ¹⁷Y. Friedman and M. D. Semon, "Relativistic acceleration of charged particles in uniform and mutually perpendicular electric and magnetic fields as viewed in the laboratory frame," *Phys. Rev. E* **72**, 026603 (2005).
- ¹⁸S. Zenitani, "Dissipation in relativistic pair-plasma reconnection: Revisited," *Plasma Phys. Controlled Fusion* **60**, 014028 (2018).
- ¹⁹S. E. Parker and C. K. Birdsall, "Numerical error in electron orbits with large $\omega_{ce}\Delta t$," *J. Comput. Phys.* **97**, 91 (1991).
- ²⁰Y. K. Wu, E. Forest, and D. S. Robin, "Explicit symplectic integrator for s-dependent static magnetic field," *Phys. Rev. E* **68**, 046502 (2003).
- ²¹B. A. Shadwick, A. B. Stamm, and E. G. Evstatiev, "Variational formulation of macro-particle plasma simulation algorithms," *Phys. Plasmas* **21**, 055708 (2014).
- ²²R. Zhang, Y. Wang, Y. He, J. Xiao, J. Liu, H. Qin, and Y. Tang, "Explicit symplectic algorithms based on generating functions for relativistic charged particle dynamics in time-dependent electromagnetic field," *Phys. Plasmas* **25**, 022117 (2018).
- ²³A. Wolski and A. T. Herrod, "Explicit symplectic integrator for particle tracking in s-dependent static electric and magnetic fields with curved reference trajectory," *Phys. Rev. Accel. Beams* **21**, 084001 (2018).
- ²⁴P. J. Channell and C. Scovel, "Symplectic integration of Hamiltonian systems," *Nonlinearity* **3**, 231 (1990).
- ²⁵H. Yoshida, "Recent progress in the theory and application of symplectic integrators," *Celestial Mech. Dyn. Astron.* **56**, 27 (1993).
- ²⁶J. E. Marsden and M. West, "Discrete mechanics and variational integrators," *Acta Numerica* **10**, 357 (2001).
- ²⁷S. A. Chin, "Symplectic and energy-conserving algorithms for solving magnetic field trajectories," *Phys. Rev. E* **77**, 066401 (2008).