An explicit method practically preserving multiple invariants for conservative systems

Wenjun Cai[1], Yuezheng Gong[2], Yushun Wang[1]*

[1] Jiangsu Key Laboratory for NSLSCS,
School of Mathematical Sciences, Nanjing Normal University, Nanjing 210023, China

[2] College of Science, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

Abstract

Invariant-preserving numerical methods are widely employed within the broad area of conservative systems, most of which are fully implicit or linearly implicit and therefore computational expensive. The few existing explicit ones are restricted to specific problems, although they can greatly improve the efficiency. The development of explicit invariant-preserving methods for general conservative systems and the generalization of simultaneously preserving multiple invariants are still ongoing. In this paper, an explicit numerical strategy that practically preserves multiple invariants is derived for conservative systems by combining an explicit high-order Runge-Kutta (RK) scheme with a simple modification of the standard projection approach, which is named the explicit invariants-preserving (EIP) method. The proposed approach is shown to have the same order as the underlying RK method, and the error of invariants is analyzed in the order of $\mathcal{O}\big(h^{2(p+1)}\big)$, where h is the time step and p represents the order of the method. When p is appropriately large, the EIP method is practically invariants-conserving because the error of invariants can reach the machine accuracy. The method is illustrated for the cases of single and multiple invariants, with regard to both ODEs and high-dimensional PDEs. Extensive numerical experiments are presented to verify our theoretical results and demonstrate the superior behaviors of the proposed method in a long time numerical simulation. Numerical results suggest that the fourth-order EIP method preserves much better the qualitative properties than the standard fourth-order RK method, and it is more efficient in practice than the fully implicit integrators.

Keywords conservative systems, explicit invariants-preserving method, practically invariants-conserving, multiple invariants

1 Introduction

For an autonomous ordinary differential equation system

$$\dot{y} = f(y), \quad y(0) = y_0, \tag{1.1}$$

where $f : \mathcal{D} \subset \mathbb{R}^d \to \mathbb{R}^d$ is sufficiently smooth, and $y_0 \in \mathbb{R}^d$ is the initial data. A scalar function $I(y)$ is called a first integral or strong invariant if

$$\nabla I(y)^T f(y) = 0, \quad y \in \mathcal{D}.$$

A function $I(y)$ is called a weak invariant of (1.1) if $y_0 \in \mathcal{M}$ implies the exact solution $y(t) \in \mathcal{M}$ and

$$\frac{\mathrm{d}}{\mathrm{d}t} I(y) = \nabla I(y)^T f(y) = 0.$$

Note that strong invariants are independent of initial values, while weak invariants depend on appropriate initial conditions. These invariants play an important role both in qualitative and quantitative studies of

---

*Correspondence author. Email: wangyushun@njnu.edu.cn.

the flow of (1.1). Nowadays the ability to preserve some invariant properties is a criterion to judge the success of a numerical integrator [43].

In the past decades, there has been an increasing interest in invariant-preserving integrators that preserve as much as possible the invariants of underlying differential systems. It is well known that all Runge-Kutta (RK) methods preserve linear invariants, while only those satisfying the symplectic condition conserve all quadratic invariants [20]. However, for $k \geq 3$ no RK method can conserve all polynomial invariants of degree k [33]. In order to develop numerical methods that conserve general invariants, McLachlan et al. rewrote the initial problem with a first integral into a gradient system and then proposed discrete gradient (DG) methods, which are based upon the definition of a discrete counterpart of the gradient operator [47]. There are several choices of discrete gradients, including the coordinate increment DG [38], the midpoint DG [31] and the mean value DG [35] (also called the averaged vector field (AVF) method in [51]) as well as their extensions [58, 13]. Later, these ideas are generalized for conservative PDEs by Furihata, Matsuo and collaborators using the concept of discrete variational derivatives (DVD) [25, 46, 27]. In recent years, high-order integrators have also been widely developed, including the high-order AVF methods [32, 19, 42], Hamiltonian boundary value methods (HBVMs) [8, 5, 7], time finite element methods [4, 55] and so on. Unfortunately, the structure-preserving algorithms mentioned above are often fully implicit for general conservative systems, which require a nonlinear iteration to solve them. In addition, these methods usually can only preserve a single invariant. Only few of their variations are designed to preserve multiple invariants [6, 16], which also belong to fully implicit methods and are computational expensive.

In contrast to fully implicit schemes, linearly implicit methods only involve solving a linear system at each time step, and therefore have attracted continuous attention in recent years. To design linear-implicit conservative schemes, Matsuo and Furihata proposed the multiple points DVD method, which was a generalization of the DVD approach [46]. More generally, Dahlby and Owren utilized the polarization technique to develop a general framework for deriving linearly implicit conservative algorithms for PDEs with polynomial invariants [22]. More recently, the invariant energy quadratization (IEQ) [59, 60, 30] and the scalar auxiliary variable (SAV) approaches [53, 54, 48], originally proposed for dissipative gradient flow models, have been successfully applied to various conservative systems [12, 11, 40, 39, 17, 62]. However, these linearly implicit methods only conserve single polarized invariant or a modified quadratic energy.

Although linearly implicit methods can already reduce the computational cost, it is more preferable to design explicit invariant-preserving algorithms for conservative problems. So far, there are only a few related works existed in the literature. When considering special second-order systems $\ddot{y} = f(y)$, the symplectic Runge-Kutta-Nyström (RKN) methods can naturally preserve quadratic invariants of the form $y^{\top} D \dot{y}$ where D is a skew-symmetric matrix of appropriate dimensions, and importantly various explicit RKN methods with high order were already constructed (see [50, 15] and references therein). For more general invariants of second-order problems, two kinds of explicit schemes based on the DVD method were proposed in [26, 45] to preserve the energy of nonlinear wave equations. However, all these methods are restricted to special second-order systems. For quadratic invariants of general systems, a rational four-stage explicit RK method was derived by del Buono and Mastroserio [23], which can be viewed as an incremental direction projection method and has been further generalized by Calvo et al. [14] to any high-order explicit RK method. Combining the IEQ approach, general invariants can be reformulated into quadratic forms. Based on this fact, Zhang et al. [62] extended the incremental direction projection method to construct explicit conservative schemes for general Hamiltonian ODEs and PDEs. A similar but more efficient strategy was recently proposed by Jiang et al. [41] that incorporates high-order explicit RK methods with standard orthogonal projection techniques. The key of the above projection methods that make the resulting schemes explicit is attributed to the quadratization of the invariant, and as a consequence, the related unknown parameter (also called the Lagrangian multiplier) can be explicitly solved. Otherwise, for non-quadratic invariant, nonlinear iterations are inevitable to solve this parameter in the standard projection methods, and the resulting schemes are no long strictly explicit, although usually only one nonlinear equation is involved when considering to preserve single invariant. However, it should be noticed that the modified quadratic invariant by the IEQ approach is equivalent to the original invariant in the continuous case but will be different after discretizations. Therefore, either the increment direction projection method [62] or the orthogonal projection method [41] actually preserves a modified invariant. Moreover, these two kinds of methods are only suitable for single invariant because there are no explicit solutions for unknown parameters

in multiple invariants cases.

The above mentioned structure-preserving algorithms can be rigorously proved to conserve some invariant properties exactly in theory, but they tend to achieve a practical conservation in numerical computing due to the following reasons. Firstly, nonlinear iterations are required for fully implicit methods whereas linear solvers are usually employed for linearly implicit methods, where an iteration tolerance must be given that may induce accumulated errors in the invariant. Secondly, as all the computations are done on a finite precision computer, even the explicit methods will inevitably suffer round-off errors. Based on this fact, in this paper we propose a novel invariant-preserving method named as EIP which has the following advantages comparing with the aforementioned methods:

(i) The EIP method is totally explicit and practically invariant-preserving for general conservative systems, while the existing explicit approaches are only suitable for specific systems. Meanwhile, numerical performance on the invariant preservation is even better than the strictly invariant-preserving methods.

(ii) The invariant preserved by the EIP method is the original energy, while some efficient approaches can only conserve a modified one.

(iii) Multiple invariants can also be preserved by the EIP method with almost no modifications of the formula, while few existing works can achieve this property.

(iv) The implementation is uniform for any conservative system, requiring only knowledge of the vector field and the invariants to be preserved, while other methods have to derive the concrete schemes before computing.

The fundamental idea of the EIP method is based on the projection methods. We stop the Newton iteration for the unknown parameters in the standard orthogonal projection approach after only one step and thus obtain an explicit form of the unknown parameters. Similar idea can be found in [9], without theoretical analysis, to correct the drift of energy where the underlying methods are chosen as conservative HBVMs. In the present work, we show that actually any explicit high-order RK methods, even non-conservative, can be taken into our framework to achieve a practical invariant preservation. Moreover, we give a rigorous proof that such simplification does not affect the accuracy of the projection method, which still retains the same order as the underlying RK method. Meanwhile, the invariants are also preserved to round-off errors as long as high-order RK methods are employed. The advantages of the EIP method are illustrated by many practical applications, including the perturbed Kepler problem, the solar system, the charged particle dynamics and the rotating Gross-Pitaevskii equation. Numerical results suggest that the EIP method perform the superior behaviors in a long time simulation and essentially improve the numerical performance of the standard fourth-order RK method.

The rest of this paper is organized as follows. In section 2, we present the EIP method and analyze its order of accuracy as well as the corresponding error in the invariant. In section 3, extensive numerical tests are reported to confirm the theoretical results and demonstrate the efficiency of the EIP method. Finally, concluding remarks are drawn in section 4.

2   Explicit and practically invariants-preserving method

In this section, we first present the algorithm of the explicit invariants-preserving (EIP) method. Afterward, the analysis on its accuracy and invariants-preserving property are carried out. Some implementation issues are provided at the end of this section.

Suppose we have a $(d - l)$-dimensional submanifold of $\mathbb{R}^d$,

$$\mathcal{M} = \{y; g(y) = 0\} \tag{2.1}$$

$(g : \mathbb{R}^d \longrightarrow \mathbb{R}^l)$, and the system (1.1) with the property that

$$y_0 \in \mathcal{M} \qquad \text{implies} \qquad y(t) \in \mathcal{M} \text{ for all } t. \tag{2.2}$$

Note that all components $g_i(y)$ of $g(y)$ are regarded as weak invariants of (1.1). Denote y_n be the approximation of $y(t)$ at $t_n = t_0 + nh$ with a step size h, $n = 0, 1, 2, \cdots$. Let Φ_h define an explicit RK method of order p with time step h. For given $y_n \in \mathcal{M}$, the EIP method is defined by:

1. Compute $\widehat{y}_{n+1}$ such that $\widehat{y}_{n+1} = \Phi_h(y_n)$;

2. Compute the value of vector $\widehat{\lambda} \in \mathbb{R}^l$ by

$$\widehat{\lambda} = -\left(\nabla g(\widehat{y}_{n+1})^\top \nabla g(\widehat{y}_{n+1})\right)^{-1} g(\widehat{y}_{n+1}), \quad \text{with} \quad \nabla g(\widehat{y}_{n+1}) \in \mathbb{R}^{d \times l}; \tag{2.3}$$

3. Update $y_{n+1} = \widehat{y}_{n+1} + \nabla g(\widehat{y}_{n+1})\widehat{\lambda}$.

Proposition 2.1. *The EIP method is equivalent to the standard orthogonal projection approach [33] for system* (1.1) *with only one Newton iteration to solve the corresponding nonlinear system.*

Proof. *The algorithm of the standard orthogonal projection approach can be written as follows:*

1. *Compute $\widehat{y}_{n+1}$ by $\widehat{y}_{n+1} = \Phi_h(y_n)$;*

2. *Project the value $\widehat{y}_{n+1}$ onto the manifold $\mathcal{M}$ through*

$$y_{n+1} = \widehat{y}_{n+1} + \nabla g(\widehat{y}_{n+1})\lambda \quad and \quad g(y_{n+1}) = 0, \tag{2.4}$$

where $\nabla g(\widehat{y}_{n+1})$ is the projection direction. At each time step, we have to solve a nonlinear system of λ as

$$g\left(\widehat{y}_{n+1} + \nabla g(\widehat{y}_{n+1})\lambda\right) = 0. \tag{2.5}$$

Denote $F(\lambda) = g\left(\widehat{y}_{n+1} + \nabla g(\widehat{y}_{n+1})\lambda\right)$. Then a Newton iteration for (2.5) *yields*

$$\lambda_{k+1} = \lambda_k - \left[\nabla F(\lambda_k)\right]^{-1} F(\lambda_k), \quad k = 0, 1, 2, \cdots. \tag{2.6}$$

Since λ is actually of small magnitude as stated in [33] and will be further proved below, we let the starting value $\lambda_0 = 0$, then one Newton iteration step of (2.6) *yields the formula* (2.3) *exactly.* $\quad\square$

2.1 Accuracy and invariants preservation

Although the EIP method is a further simplification of the standard orthogonal projection approach, we will show that its accuracy is still the same as the underlying RK method. Moreover, the invariant error can be significantly reduced to a magnitude of $\mathcal{O}(h^{2(p+1)})$, that is, utilizing a high-order RK method the EIP method can preserve the invariant to the machine accuracy in practical computation.

As stated in Proposition 2.1, the EIP method is related to the standard projection approach. Therefore, we can analyze the EIP method from the aspect of the simplified projection approach, specifically regarding to the solution of the parameter λ which is the key difference between the two kinds of methods.

Lemma 2.1. *Assume $\nabla g(y_n)$ has full column rank. There exists $h_\star > 0$ such that the nonlinear system* (2.5) *for the standard projection approach has a unique solution $\lambda_\star = \lambda_\star(h)$ for $h \in [0, h_\star]$ and $\lambda_\star$ is of size $\mathcal{O}(h^{p+1})$. Therefore, the standard projection approach is of order p.*

Proof. *For simplicity, we denote*

$$\widehat{F}(\lambda, h) = g\left(\widehat{y}_{n+1} + \nabla g(\widehat{y}_{n+1})\lambda\right) = 0.$$

Notice that

$$\widehat{F}(0, 0) = g(y_n) = 0, \quad and \quad \nabla_\lambda \widehat{F}(0, 0) = \nabla g(y_n)^\top \nabla g(y_n)$$

is invertible as $\nabla g(y_n)$ has full rank, then the implicit function theorem ensures the existence of a neighborhood $[0, h_\star]$ and a unique smooth solution $\lambda_\star = \lambda_\star(h)$ such that $\lambda_\star(0) = 0$ and $\widehat{F}\left(\lambda_\star(h), h\right) = 0$ for all $h \in [0, h_\star]$. Moreover, we can expand

$$\widehat{F}(\lambda_\star, h) = \widehat{F}(0, h) + \nabla_\lambda \widehat{F}(0, h)\lambda_\star + \mathcal{O}(\lambda_\star^2),$$

with

$$\widehat{F}(0, h) = g(\widehat{y}_{n+1}) = \mathcal{O}(h^{p+1}), \quad \nabla_\lambda \widehat{F}(0, h) = \nabla_\lambda \widehat{F}(0, 0) + \mathcal{O}(h).$$

Then, we can derive $\lambda_\star = \lambda_\star(h) = \mathcal{O}(h^{p+1})$, which implies the standard projection approach is of order p. $\quad\square$

Lemma 2.2. *Assume $\lambda_\star$ is the exact solution of the nonlinear system (2.5). Then the k-th iteration solution λ_k obtained by (2.6) satisfies $\lambda_k - \lambda_\star = \mathcal{O}(h^{2^k(p+1)})$. Moreover, we have $F(\lambda_k) = \mathcal{O}(h^{2^k(p+1)})$.*

Proof. *A direct calculation yields*

$$
\begin{aligned}
\lambda_{k+1} - \lambda_\star &= \lambda_k - \left[\nabla F(\lambda_k)\right]^{-1} F(\lambda_k) - \lambda_\star \\
&= \lambda_k - \lambda_\star - \left[\nabla F(\lambda_k)\right]^{-1} \left(F(\lambda_\star) - \nabla F(\lambda_k)(\lambda_\star - \lambda_k) - \mathcal{O}(\|\lambda_\star - \lambda_k\|^2) \right) \\
&= \lambda_k - \lambda_\star + \left[\nabla F(\lambda_k)\right]^{-1} \left(\nabla F(\lambda_k)(\lambda_\star - \lambda_k) + \mathcal{O}(\|\lambda_\star - \lambda_k\|^2) \right) \\
&= \mathcal{O}(\|\lambda_\star - \lambda_k\|^2),
\end{aligned}
$$

where we have used $F(\lambda_\star) = 0$ and $\|\cdot\|$ is the Euclidean norm. Let the starting value $\lambda_0 = 0$. We can obtain $\lambda_1 - \lambda_\star = \mathcal{O}(\|\lambda_\star\|^2) = \mathcal{O}(h^{2(p+1)})$ by Lemma 2.1. Thus we can recursively derive $\lambda_k - \lambda_\star = \mathcal{O}(h^{2^k(p+1)})$, which leads to $F(\lambda_k) = \mathcal{O}(h^{2^k(p+1)})$ by using Taylor expansion. This completes the proof. $\square$

As a direct consequence of Lemma 2.2, we have the following main results about the accuracy and invariant error of the EIP method.

Theorem 2.1. *The EIP method retains the same order of accuracy as the underlying RK method, and the invariant error is of size $\mathcal{O}(h^{2(p+1)})$.*

Proof. *From Lemma 2.2, the parameter $\widehat{\lambda}$ in the EIP method satisfies $\widehat{\lambda} = \lambda_1 = \lambda_\star + \mathcal{O}(h^{2(p+1)}) = \mathcal{O}(h^{p+1})$. Since Φ_h corresponds to a RK method of order p, by Taylor's Theorem we have $\widehat{y}_{n+1} = y(t_{n+1}) + \mathcal{O}(h^{p+1})$ and*

$$
\nabla g(\widehat{y}_{n+1}) = \nabla g(y(t_{n+1})) + \mathcal{O}(h^{p+1}).
$$

Hence the local error of the EIP method yields

$$
y_{n+1} = \widehat{y}_{n+1} + \nabla g(\widehat{y}_{n+1})\widehat{\lambda} = y(t_{n+1}) + \mathcal{O}(h^{p+1}),
$$

that is, the EIP method has an accuracy with the same order as the underlying RK method. While the invariant error is a straightforward result of Lemma 2.2 by setting $k = 1$. $\square$

Remark 2.1. *Given a time step h, if we choose a high-order underlying RK method in the first step of the EIP method, the invariant can reach to the round-off error of double precision machines. In this sense, we call it a practically invariant-preserving method.*

Remark 2.2. *Due to the equivalence between the EIP method and the projection method, $\nabla g(\widehat{y}_{n+1})$ in the last step of the EIP method actually corresponds to a projection direction. Therefore, we can replace this term with any other suitable projection directions. Of course, the steepest descent direction for convergence is $\nabla g(y_{n+1})$. However, to ensure that the resulting method is fully explicit, the direction can only involve some known solutions, such as y_n and $\widehat{y}_{n+1}$. For example, if we take $\nabla g(y_n)$ or $\nabla g\left(\frac{1}{2}(y_n + \widehat{y}_{n+1})\right)$ as the projection direction, we can also obtain the same results about the accuracy and the invariant error. Meanwhile, it is not necessary to restrict the direction to forms of $\nabla g(y)$. Instead, any other effective projection directions can be utilized to construct the EIP method. However, among those directions, which one has the fastest convergence turns into an optimization problem and is worthy of further study.*

To illustrate the above results about the accuracy and invariant error of the EIP method, we consider the harmonic oscillator as a concrete example

$$
\dot{y} = \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} y,
$$

where $y = (p, q)^\top$ and by definition $\omega > 0$. This is a Hamiltonian system with quadratic Hamiltonian $H(y) = \frac{\omega}{2} y^\top y$. For a given initial data y_0, the invariant in this example becomes $g(y) = H(y) - H(y_0)$.

After one step of a p-th order RK method $y_n \to \widehat{y}_{n+1}$, we first apply the standard projection approach and obtain the corresponding nonlinear system (2.5) which is actually a quadratic equation of λ

$$a_2\lambda^2 + 2a_1\lambda + a_0 = H(y_0),$$

where $a_k = \frac{\omega^{k+1}}{2}\widehat{y}_{n+1}^\top \widehat{y}_{n+1}$ and thereby can be solved analytically, which is exactly the basic ideas in [14, 62, 41] for quadratic invariant. Of the two solutions that exist for λ we choose the one with the smallest absolute value since we expect λ to be close to zero for small h. Hence, we get

$$\lambda_\star = -\frac{1}{\omega}\left(1 - \frac{\|y_0\|}{\|\widehat{y}_{n+1}\|}\right) = \mathcal{O}(h^{p+1}). \tag{2.7}$$

While for the EIP method, $\widehat{\lambda}$ can be directly calculated by

$$\widehat{\lambda} = -\frac{1}{2\omega}\left(1 - \frac{\|y_0\|^2}{\|\widehat{y}_{n+1}\|^2}\right) = \frac{1}{2}\left(1 + \frac{\|y_0\|}{\|\widehat{y}_{n+1}\|}\right)\lambda_\star = \lambda_\star + \mathcal{O}(h^{2(p+1)}).$$

Obviously, the local errors of $\widehat{\lambda}$ and $\lambda_\star$ coincide with the theoretical results in Lemma 2.1 and 2.2, respectively. Subsequently, we can also derive the invariant error as $g(y_{n+1}) = \mathcal{O}(h^{2(p+1)})$ by a direct substitution and Taylor's series expansion.

We numerically simulate the harmonic oscillator by the EIP methods with the underlying RK methods, i.e., the classic explicit ones of order 1 to 4 whose Butcher tabular are listed as follows:

$$
\begin{array}{c|c}
0 & \\
\hline
 & 1
\end{array}
\qquad
\begin{array}{c|cc}
0 & \\
1/2 & 1/2 \\
\hline
 & 0 & 1
\end{array}
\qquad
\begin{array}{c|ccc}
0 & \\
1/3 & 1/3 \\
2/3 & 0 & 2/3 \\
\hline
 & 1/4 & 0 & 3/4
\end{array}
\qquad
\begin{array}{c|cccc}
0 & \\
1/2 & 1/2 \\
1/2 & 0 & 1/2 \\
1 & 0 & 0 & 1 \\
\hline
 & 1/6 & 2/6 & 2/6 & 1/6
\end{array}
.
$$

For convenience, we denote the above four RK methods as RK1, RK2, RK3, RK4, respectively. In Table. 1, the errors of $|\widehat{\lambda} - \lambda_\star|$ and the corresponding orders are presented. For RK1 and RK3, the results are coincided with the analytical derivation, whereas for RK2 and RK4, we can observe a surprising result that the convergence order becomes $2(p+2)$ instead of $2(p+1)$ which is mainly because the simulated model is linear. As a consequence, the related energy errors also behave similarly in Table. 2, where the results labeled iter#1 correspond to the EIP method that can be viewed as a projection method with one step iteration. From Lemma 2.2, the invariant error of the standard projection method is of size $\mathcal{O}(h^{2^k(p+1)})$ with k being the iteration number for the nonlinear system (2.5). Therefore, we also provide results with $k = 2$ for RK1 and RK2, which again conform the theoretical analysis. While for RK3 and RK4, iteration twice will make the energy errors reach the machine accuracy quickly and consequently we can hardly get the convergence orders.

h	RK1		RK2		RK3		RK4	
	error	order	error	order	error	order	error	order
0.1/1	6.3506e-03		6.2524e-04		4.0849e-05		1.8688e-06	
0.1/2	6.2524e-04	3.3444	3.0048e-06	7.7010	2.8629e-07	7.1567	5.5259e-10	11.7237
0.1/4	4.5956e-05	3.7661	1.1909e-08	7.9790	1.2703e-09	7.8161	1.4148e-13	11.9314
0.1/8	3.0048e-06	3.9349	4.6563e-11	7.9987	5.1204e-12	7.9547	3.3307e-17	12.0525

Table 1: Errors of $|\widehat{\lambda} - \lambda_\star|$ and the corresponding orders for EIP methods with different underlying RK methods for the harmonic oscillator problem with $\omega = 10$, $y_0 = [1, 0]^\top$ at $t = 1$.

			$h_0/1$	$h_0/2$	$h_0/4$	$h_0/8$
RK1	iter#1	error	1.0354	7.0644e-02	4.7404e-03	3.0283e-04
		order	-	3.8735	3.8975	3.9684
	iter#2	error	1.7712e-02	1.9303e-04	1.0550e-06	4.5142e-09
		order	-	6.5197	7.5154	7.8686
RK2	iter#1	error	3.1922	7.0644e-02	3.0283e-04	1.1915e-06
		order	-	8.8198	7.8659	7.9896
	iter#2	error	5.6576e-01	1.9303e-04	4.5142e-09	7.1054e-14
		order	-	11.5171	15.3840	15.9552
RK3	iter#1	error	2.1230e-01	3.9722e-03	2.8561e-05	1.2701e-07
		order	-	5.7400	7.1198	7.8129
RK4	iter#1	error	3.4710e-01	1.8575e-04	5.5253e-08	1.4149e-11
		order	-	10.8678	11.7150	11.9312

Table 2: Energy errors and the corresponding orders for EIP methods with different underlying RK methods for the harmonic oscillator problem with $\omega = 10$, $y_0 = [1,0]^\top$ till $t = 1$. $h_0 = 0.1$ for RK1 and $h_0 = 0.2$ for the rest.

Remark 2.3. *According to the above simple example, we actually have illustrate the key strategy in [14, 62, 41] to make their methods explicit for quadratic invariant or general invariant after quadratization. Nevertheless, this strategy is only feasible for preservation of single invariant because there will be no analytical solutions of vector-valued λ for multiple invariants.*

Although the standard projection method can handle multiple invariants, it inevitably requires iterations to solve the nonlinear system (2.5) for vector-valued λ, which makes the standard projection method less efficient than the EIP method.

2.2 Implementation issue

As demonstrated in the algorithm of the EIP method, its implementation is completely explicit and the main effort is paid to calculate the parameter λ. In this subsection, with respect to single and multiple invariants, we will provide the expression of $\widehat{\lambda}$ in details for both ODEs and PDEs circumstances.

- Case I : single invariant

Suppose we have obtained $\widehat{y}_{n+1}$ through an explicit RK method Φ_h. For ODEs or one-dimensional PDEs after semi-discretization, $g(y) : \mathbb{R}^N \to \mathbb{R}$ and $\widehat{\lambda}$ is a scalar. Here, we use N as the notation of dimension instead of the aforementioned d for convenience. Consequently, the expression (2.3) of $\widehat{\lambda}$ can be simplified as

$$\widehat{\lambda} = -\frac{g(\widehat{y})}{\|\nabla g(\widehat{y})\|^2}, \tag{2.8}$$

where we have omitted the subscript of $\widehat{y}_{n+1}$ for simplicity. Specifically, when considering the canonical Hamiltonian system and let the Hamiltonian energy $H(y)$ be the targeted invariant, the formula (2.8) becomes

$$\widehat{\lambda} = -\frac{H(\widehat{y}) - H(y_0)}{\|\nabla H(\widehat{y})\|^2}, \tag{2.9}$$

and subsequently $y = \widehat{y} + \lambda \nabla H(\widehat{y})$, where the subscript of y_{n+1} is also omitted.

Remark 2.4. *In Ref. [9], the authors proposed a correction technique to prevent the accumulation of round-off errors for HBVMs. Actually, such technique belongs to the framework of the EIP method with*

7

the parameter calculated exactly as (2.9) and HBVMs as the one-step method Φ_h. As analyzed in the above section, the EIP method can make a non-conservative RK method invariants-preserving. Therefore, the additional employment of the EIP method as a correction technique for HBVMs may confuse the performance in the preservation of invariants. Nevertheless, for general conservative methods, the accumulation of round-off errors often renders the invariant errors growing linearly. How to design a technique to prevent it is still worthy of further study.

Due to the explicit form, the EIP method will be more competitive for PDEs of high dimensions. However, it is not straightforward to extend formula (2.3) for such cases, even if only one invariant needs to be preserved. Consider a two-dimensional problem with $\widehat{y}_{n+1} \in \mathbb{R}^{N \times N}$ for example, where N represents the grid size in both x and y directions. Then the gradient of single invariant $\nabla g(\widehat{y}_{n+1})$ is of dimension $N \times N$ in general. A direct substitution into formula (2.3) will suffer the disagreement of dimension. To overcome this obstacle, we define a function "vec" that represents the vectorization of matrix, i.e., $\text{vec}(A_{N \times N})$ results a vector of dimension $N^2 \times 1$, composed of the elements of A. Then we can obtain

$$\widehat{\lambda} = -\frac{g(\widehat{y})}{\|\text{vec}(\nabla g(\widehat{y}))\|^2}, \tag{2.10}$$

which can be generalized to three-dimensional cases in a straightforward manner.

- Case II : multiple invariants

For ODEs and one-dimensional semi-discrete PDEs with l invariants, the expression of λ retains the same as that in (2.3) with $\nabla g(\widehat{y}) \in \mathbb{R}^{N \times l}$. Different from (2.8) and (2.10) for single invariant where the denominators are scalars, one has to compute the inverse of a matrix of dimension $l \times l$. For high-dimensional PDEs after semi-discretization, the technique in (2.10) should also be employed, and the calculation of $\widehat{\lambda}$ yields

$$\widehat{\lambda} = -\left(\mathcal{G}^\top \mathcal{G}\right)^{-1} g(\widehat{y}_{n+1}), \tag{2.11}$$

where $\mathcal{G} = \left(\text{vec}\left(\nabla g_1(\widehat{y})\right), \text{vec}\left(\nabla g_2(\widehat{y})\right), \cdots, \text{vec}\left(\nabla g_l(\widehat{y})\right)\right) \in \mathbb{R}^{N^2 \times l}$ for two-dimensional problems. Accordingly, the implementation of the EIP method for multiple invariants is extremely simple, whereas most of existing conservative methods fail to simultaneously preserve two or more invariants.

3 Numerical experiments

In this section, we will present various numerical examples for both conservative ODEs and PDEs to verify the theoretical results and demonstrate the efficiency of the EIP method. The underlying one step method is uniformly taken as RK4 unless otherwise stated. All experiments are performed in Matlab, version 2016b, on a computer equipped with the Intel Core i7-6700 processor running at 3.40GHz. For the part of accuracy tests, when the exact solution is unknown, we take the following strategy to compute the convergence order.

Consider a pth-order numerical approximation of an exact solution y (a scalar variable for simplicity). The approximation depends on a small parameter h, either as the grid size or time step, and we denote it by y_h. By the h-expansion [34] of the global error, we have

$$y_h = y + Ch^p + \mathcal{O}(h^{p+1}),$$

where the number C is independent of h and typically depends on the exact solution. Next, we halve the step size h successively and obtain

$$y_{h/2} = y + C(h/2)^p + \mathcal{O}(h^{p+1}), \quad \text{and} \quad y_{h/4} = y + C(h/4)^p + \mathcal{O}(h^{p+1}).$$

The ratio of the difference between the above three numerical solutions is calculated by

$$\frac{y_h - y_{h/2}}{y_{h/2} - y_{h/4}} = \frac{1 - 2^{-p} + O(h)}{2^{-p} - 2^{-2p} + O(h)} = 2^p + \mathcal{O}(h). \tag{3.1}$$

Hence, after continually decreasing h and compute the logarithm of the left side of (3.1), we can get an estimate of p. For vector-valued solutions, the difference should be measured under some appropriate norms.

3.1 Experiment I: perturbed Kepler system

We first consider the perturbed Kepler system with Hamiltonian

$$H(q_1, q_2, p_1, p_2) = \frac{1}{2}(p_1^2 + p_2^2) - \frac{1}{\sqrt{q_1^2 + q_2^2}} - \frac{0.005}{2\sqrt{(q_1^2 + q_2^2)^3}}, \tag{3.2}$$

which describes the motion of a planet in the Schwarzschild potential for Einstein's general relatively theory. Besides the above Hamiltonian, this problem has another invariant, i.e., the angular momentum

$$L(q_1, q_2, p_1, p_2) = q_1 p_2 - q_2 p_1. \tag{3.3}$$

Most of existing invariant-preserving methods usually preserve only one of the above two invariants. For examples, the AVF method can be used to preserve the Hamiltonian (3.2) while the symplectic partitioned RK method can handle with the angular momentum (3.3) in general. In this experiment, we will not only verify the effectiveness of the proposed method through the accuracy test, but also show its flexibility in the conservation of either one or two invariants.

Let the initial conditions $q_1(0) = 1 - e$, $q_2(0) = 0$, $p_1(0) = 0$, $p_2(0) = \sqrt{(1 + e)/(1 - e)}$ where e represents the eccentricity and is taken as $e = 0.6$. Although one can derive the exact solution of the perturbed Kepler system based on the two invariants [33], we here choose formula (3.1) instead to compute the convergence order of the EIP methods. Denote **EIP-H**, **EIP-L** and **EIP-HL** as the methods designed to preserve H, L and both, respectively. The results of accuracy tests for those methods are listed in Table. 3, which uniformly show a convergence order of 4 as expected.

h	EIP-H		EIP-L		EIP-HL	
	error	order	error	order	error	order
0.02/1						
0.02/2	1.3935e-06		2.2607e-06		4.7011e-07	
0.02/4	8.6286e-08	4.0134	1.3647e-07	4.0501	3.2365e-08	3.8605
0.02/8	5.3636e-09	4.0079	8.3757e-09	4.0262	2.1111e-09	3.9384
0.02/16	3.3424e-10	4.0042	5.1865e-10	4.0134	1.3462e-10	3.9710

Table 3: Errors between two adjacent time steps and the corresponding orders for different EIP methods.

Different from the harmonic oscillator model, the Kepler system is fully nonlinear. We also present the convergence test for the invariant errors of different EIP methods to verify the result in Lemma 2.2. In Table 4, due to the barrier of machine accuracy, we only give the corresponding errors and orders when the underlying methods are chosen as RK1 and RK2, which is clearly coincided with our analysis. In addition, there is no super convergence result that occurs for RK2 in the harmonic oscillator case.

	h	EIP-H		EIP-L		EIP-HL	
		error	order	error	order	error	order
RK1	0.03/1	7.6344e-08		1.1704e-07		1.0251e-07	
	0.03/2	3.3388e-09	4.5151	7.3733e-09	3.9886	5.1644e-09	4.3110
	0.03/3	6.3275e-10	4.1021	1.5123e-09	3.9072	9.8071e-10	4.0972
	0.03/4	1.9642e-10	4.0664	4.8734e-10	3.9364	3.0466e-10	4.0638
RK2	0.03/1	1.5467e-12		2.3863e-11		5.7380e-11	
	0.03/2	1.9984e-14	6.2742	3.2674e-13	6.1905	7.7061e-13	6.2184
	0.03/3	1.8874e-15	5.8198	2.9088e-14	5.9656	6.8501e-14	5.9693
	0.03/4	3.3307e-16	6.0296	5.1070e-15	6.0473	1.2434e-14	5.9314

Table 4: Energy errors and the corresponding orders for EIP methods with RK1 and RK2 as the underlying methods till $t = 1$.

Moreover, the numerical orbits as well as the errors in Hamiltonian energy and angular momentum are presented in Figure. 1 with time step $h = 0.03$ for all simulations. It is clear that all three methods present orbits like an ellipse that rotates slowly around one of its foci. For the conservation of invariants, the **EIP-H** and **EIP-L** methods exactly preserve the related single invariant and keep another one growing linearly under a small order of magnitude. Obviously, the **EIP-HL** method gives the best results with the two invariants being preserved to the machine accuracy, although it spends a little more CPU time than the former two EIP methods. A detailed comparison on the computational efficiency with respect to single and multiple invariants-preserving EIP methods will be carried out in the following PDE case.
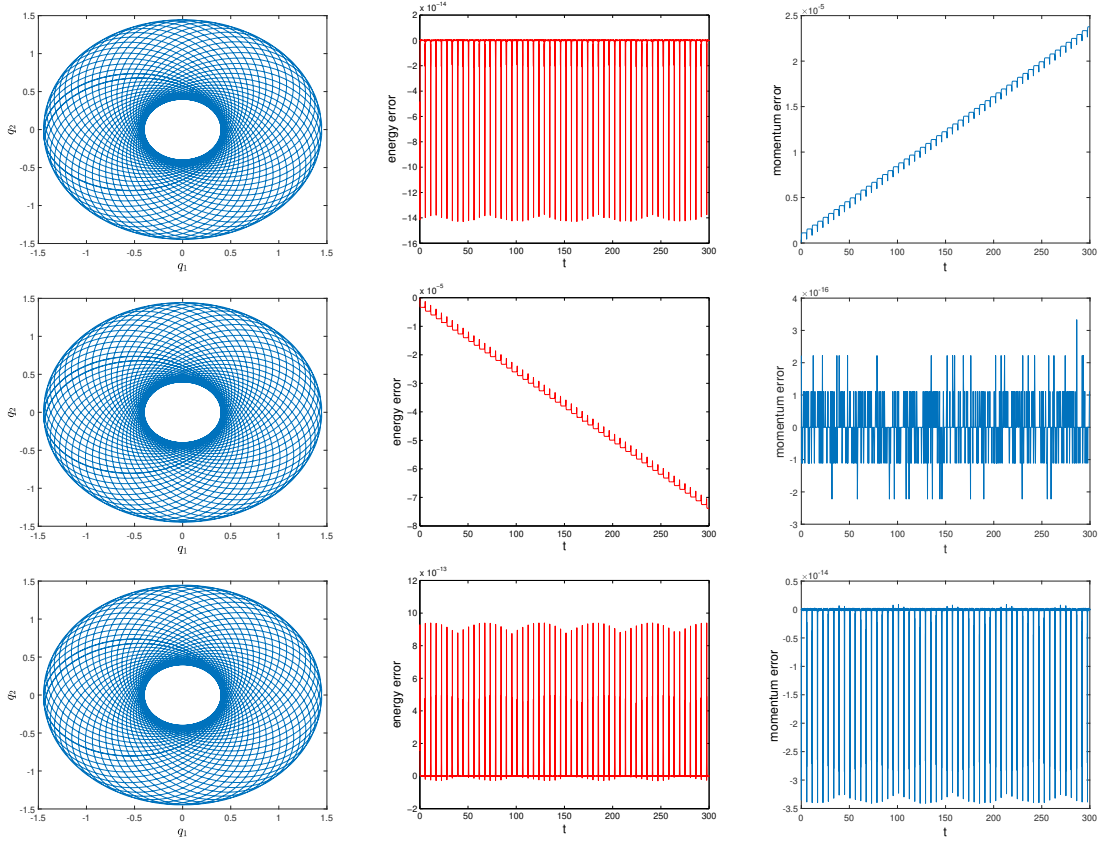
Figure 1: The results in numerical orbits (left), energy error (middle) and momentum error (right) by methods **EIP-H** (upper), **EIP-L** (middle) and **EIP-HL** (bottom), respectively.

It is mentioned in Remark 2.1 that the EIP is a practical invariants-preserving method provided a high-order underlying RK method is utilized, e.g., a fourth-order one used above. For lower-order RK methods, the numerical behaviors may not be satisfied, whereas for higher-order ones the computational complexity will be significantly increased since for order $p > 5$ the stages of explicit RK methods must be greater than p [34] which makes it less cost-effective, especially for high-dimensional PDEs. To demonstrate this observation, we further present the results with the second- and fifth-order RK methods (RK2 and RK5 in short) as the underlying RK methods, where RK2 is listed above and RK5 is taken as the classic Fehlberg method with 6 stages [34]. From Figure. 2, we can find that the orbit related to RK2 is wrong and the errors in energy and momentum are bounded but with magnitude much greater than the round-off error. While RK5 produces a correct orbit and the invariants errors exhibits even better than that in Figure. 1 by RK4 which coincides with the theoretical result. However, considering the requirement of practical computations, the magnitude of invariants errors by RK4 is already very satisfied and can be viewed as a conservation of invariants to round-off error. Therefore, from the perspective of cost performance we recommend RK4 as the underlying RK method of the EIP method. The following numerical tests will give a strong support of
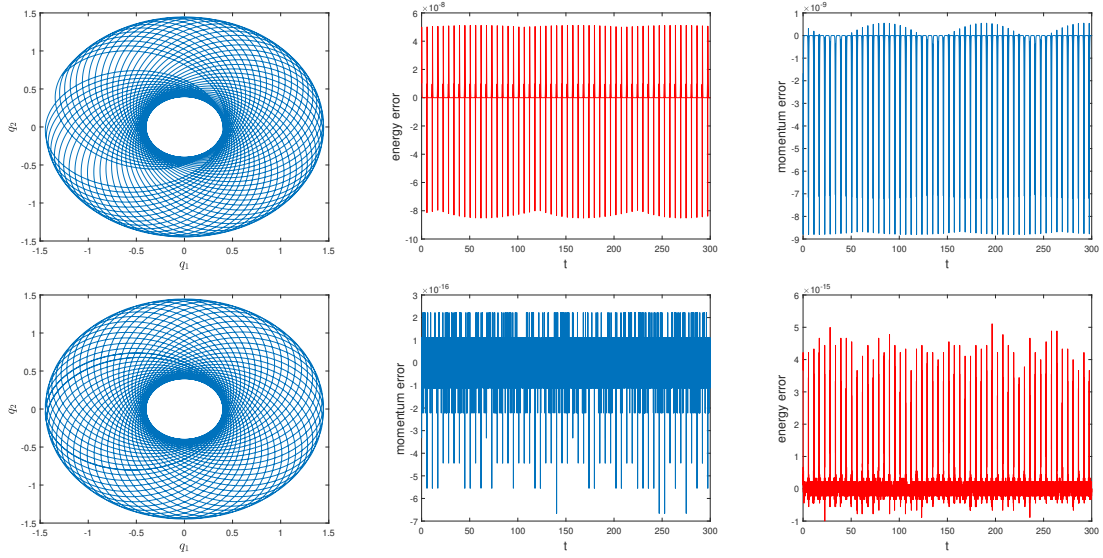
the choice.

Figure 2: The results in numerical orbits (left), energy error (middle) and momentum error (right) by the method **EIP-HL** with RK2 (upper) and RK5 (bottom) as the underlying methods, respectively.

3.2 Experiment II: solar system

Next, we apply the EIP method to simulate a more realistic problem in celestial mechanics. Consider the motion of the solar system that describes eight planets and Pluto orbiting around the sun. As a generalization of the above Kepler problem, the corresponding Hamiltonian energy becomes

$$H(p,q) = \frac{1}{2} \sum_{i=1}^{10} \frac{1}{m_i} p_i^\top p_i - \sum_{i=2}^{10} \sum_{i=1}^{i-1} \frac{Gm_i m_j}{\|q_i - q_j\|}, \tag{3.4}$$

where m_i represents the mass, and p, q are supervectors composed by the momenta and position vectors $q_i, p_i \in \mathbb{R}^3$, respectively. Besides the energy, the solar system also admits the conservation of the angular momentum $L = \sum_{i=1}^{10} q_i \times p_i$, which actually consists of three first integrals, namely

$$
\begin{aligned}
L_x &= \sum_{i=1}^{10} \left(p_i(3)q_i(2) - p_i(2)q_i(3) \right), \\
L_y &= \sum_{i=1}^{10} \left(p_i(1)q_i(3) - p_i(3)q_i(1) \right), \\
L_z &= \sum_{i=1}^{10} \left(p_i(2)q_i(1) - p_i(1)q_i(2) \right),
\end{aligned}
\tag{3.5}
$$

where $p_i(k), q_i(k)$ means the k-th component of momenta and position with respect to the i-th planet.

In the following experiments, to demonstrate the advantage of the proposed method in simultaneously preserving multiple invariants, we only focus on the numerical behaviors of the **EIP-HL** method. The initial datum of the solar system are taken from NASA JPL Ephemeris in the Appendix, and the simulation is carried out over 2000 years with $h = 0.002\text{yr}$[1]. This long-term simulation does require not only the

---

[1]Notice that the velocity in the Appendix is recorded in second and therefore the time step should also be transformed in second for practical computation.

superior stability, but also the high precision of numerical algorithms. For comparison, we also present the numerical results by the popular second-order Störmer-Verlet method (**SV**) [33]. Since **SV** is a symplectic partitioned RK method, it can automatically preserve the three angular momentum to round-off errors and keep the energy errors bounded in a small order of magnitude in Figure. 3. While the **EIP-HL** method can achieve the preservation of all the four invariants exactly, and the errors behaves like random walk due to the machine accuracy.
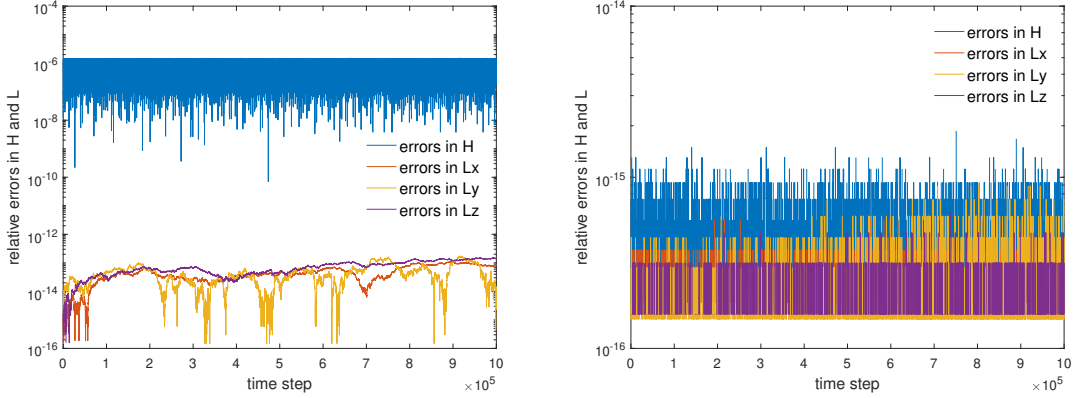
Figure 3: Relative invariant errors by **SV** (left) and the **EIP-HL** method (right).

The high-accuracy advantage of the **EIP-HL** method for the solar system is revealed by the numerical orbits of planets in Figure. 4, including the orbits of outer planets (and Pluto), the orbits of inner planets and the enlarged Mercury's orbit projected onto the xy-plane. We can hardly tell the difference from the outer orbits generated by **SV** and the **EIP-HL** method, no matter in the shapes of orbits or in the final positions of planets. However, it becomes clear from the inner orbits that the Mercury's orbit by **SV** is no long an ellipse. Instead, it suffers an undesirable precession effect, which can be observed more apparently from the projected picture on the xy-plane. The reason is mainly because of the low-order accuracy and the subsequent accumulation of phase errors, although its symplecticity guarantees the long-term stability. While for the high-order **EIP-HL** method both inner orbits and the enlarged Mercury's orbit exhibit correct elliptical shapes and the positions of planets are further improved.
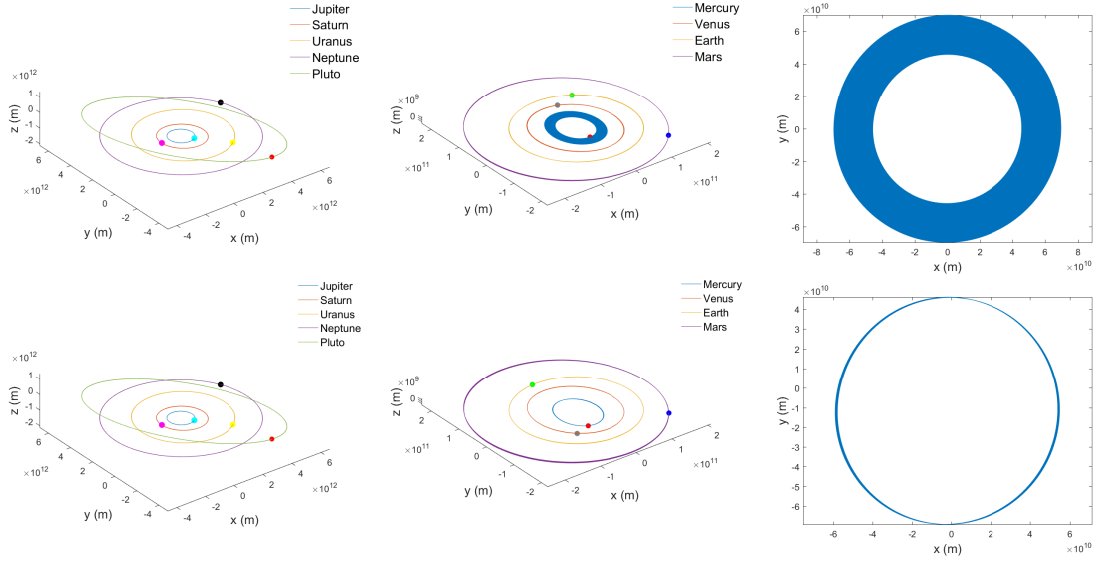
Figure 4: Orbits of outer planets and Pluto (left), orbits of inner four planets (middle) and the innermost orbit of Mercury (right) by **SV** (upper) and the **EIP-HL** method (bottom), respectively.

3.3   Experiment III: 2D charged particle dynamics

As the underlying one step method, RK4 has the same order of accuracy as the presented EIP methods. Nevertheless, the numerical behaviors can be distinct. For illustration, we consider an example of 2D charged particle dynamics.

In a given electromagnetic field $(\boldsymbol{E}(\boldsymbol{x}), \boldsymbol{B}(\boldsymbol{x}))$, the motion of a charged particle is governed by the Lorentz force law and can be described by

$$m\ddot{\boldsymbol{x}} = q(\boldsymbol{E} + \dot{\boldsymbol{x}} \times \boldsymbol{B}), \quad \boldsymbol{x} \in \mathbb{R}^3 \tag{3.6}$$

where $\boldsymbol{x}$ is the position of the charged particle, m is the mass, and q denotes the electric charge. For convenience, we assume that $\boldsymbol{B}, \boldsymbol{E}$ are static and thus $\boldsymbol{B} = \nabla \times \boldsymbol{A}$ and $\boldsymbol{E} = -\nabla\varphi$ with $\boldsymbol{A}$ and φ the potentials. Let the conjugate momentum $\boldsymbol{p} = m\dot{\boldsymbol{x}} + q\boldsymbol{A}(\boldsymbol{x})$. The system (3.6) has a canonical symplectic structure [36] with the Hamiltonian given by

$$H(\boldsymbol{x}, \boldsymbol{p}) = \frac{1}{2m}\left(\boldsymbol{p} - q\boldsymbol{A}(\boldsymbol{x})\right) \cdot \left(\boldsymbol{p} - q\boldsymbol{A}(\boldsymbol{x})\right) + q\varphi(\boldsymbol{x}), \tag{3.7}$$

which however cannot be split into the form $H(\boldsymbol{x}, \boldsymbol{p}) = T(\boldsymbol{p}) + V(\boldsymbol{x})$. A separable formulation of (3.6) can be found by recasting it with transformation $(\boldsymbol{x}, \boldsymbol{p}) \to (\boldsymbol{x}, \boldsymbol{v}) : \boldsymbol{x} = \boldsymbol{x}, \boldsymbol{v} = \boldsymbol{p}/m - q\boldsymbol{A}(\boldsymbol{x})/m$, as

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{v}, \\ \dot{\boldsymbol{v}} &= \frac{q}{m}\left(\boldsymbol{E}(\boldsymbol{x}) + \boldsymbol{v} \times \boldsymbol{B}(\boldsymbol{x})\right), \end{aligned} \tag{3.8}$$

where the corresponding energy becomes $H(\boldsymbol{x}, \boldsymbol{v}) = m\boldsymbol{v} \cdot \boldsymbol{v}/2 + q\varphi(\boldsymbol{x})$. Several structure-preserving methods have been proposed for numerically solving (3.6), including symplectic methods [63, 56, 37], volume-preserving methods [49, 36, 28] and energy-conserving methods [44, 10], with respect to the structures (3.7) or (3.8). Among those methods, explicit ones are only valid for the separable formulation (3.8). To the best of our knowledge no explicit structure-preserving method, especially no energy-conserving method, exists based on the canonical form (3.7). Hence, in the following experiments, we will utilize the proposed EIP method to construct a first explicit energy-conserving method based on this form. One can follow a similar process to derive the EIP method according to the separable system (3.8) and we omit it here.

First, we consider the 2D dynamics of the charged particle in a static, non-uniform electromagnetic field

$$\boldsymbol{B} = \nabla \times \boldsymbol{A} = \boldsymbol{e}_z, \quad \boldsymbol{E} = -\nabla\varphi = \frac{10^{-2}}{R^3}(x\boldsymbol{e}_x + y\boldsymbol{e}_y),$$

where the potentials are chosen as $\boldsymbol{A} = [-y/2, x/2, 0]^\top, \varphi = \frac{10^{-2}}{R}, R = \sqrt{x^2 + y^2}$. In this example, the physical quantities are normalized by the system size a, the characteristic magnetic field B_0, and the gyro-frequency $\omega_0 \equiv qB_0/m$ of the particle. Besides the energy (3.7), another invariant of this case is given by the angular momentum

$$L(\boldsymbol{x}, \boldsymbol{p}) = xp_y - xp_x. \tag{3.9}$$

Therefore, we construct the method **EIP-HL** that preserves both energy and angular momentum. Starting from the initial conditions $\boldsymbol{x}_0 = [0, -1, 0]^\top$, $\boldsymbol{v}_0 = [0.1, 0.01, 0]^\top$, and taking the step size $h = \pi/10$, which is the $1/20$ of the characteristic gyro-period, we run the **EIP-HL** method for 2.7×10^5 steps. The particle's exact orbit is a spiraling circle with a constant radius. For comparison, we also apply RK4 for this problem. Figure. 5 presents the orbits generated by these two methods and the corresponding errors in the invariants. Though RK4 has a same fourth-order accuracy, the numerical error accumulation after 2.7×10^5 steps gives rise to a complete wrong solution orbit during the 100th-turn, where gyro-motion is numerically dissipated. This can be also confirmed from the errors in two invariants. While the **EIP-HL** method can not only provide a correct and stable gyro-motion over such a long-term simulation but also preserve the energy and angular momentum to round-off errors.
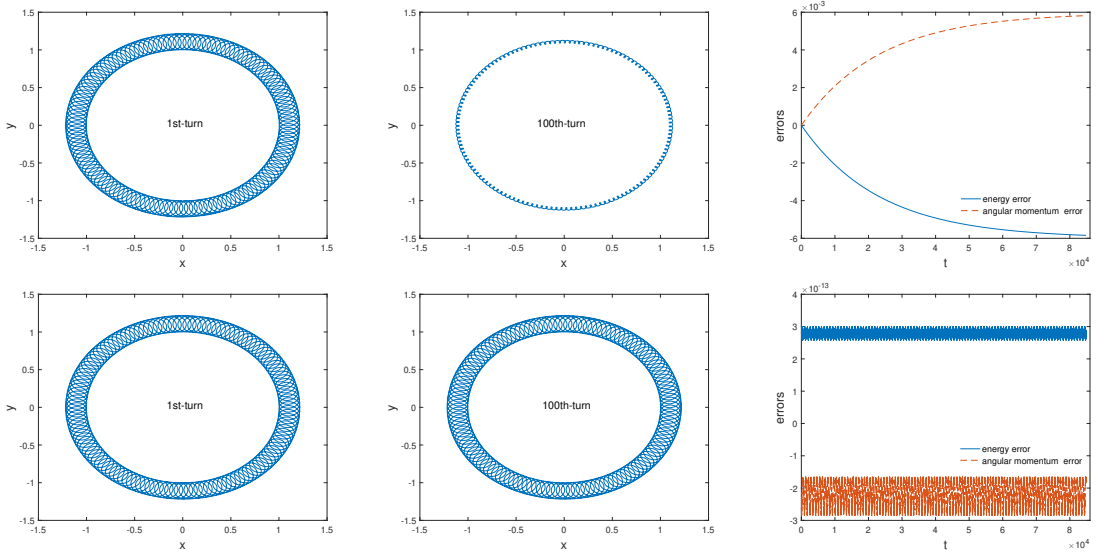

Figure 5: Numerical orbits and errors in two invariants solved on the interval $[0, 2.7 \times 10^5 h]$ by RK4 (upper) and the **EIP-HL** method (bottom).

Next, we test our method for the 2D dynamics in an axisymmetric tokamak geometry without inductive electric field where

$$\boldsymbol{B} = -\frac{2y + xz}{2R^2}\boldsymbol{e}_x + \frac{2x - yz}{2R^2}\boldsymbol{e}_y + \frac{R-1}{2R}\boldsymbol{e}_z,$$

and

$$\boldsymbol{A} = \left[\frac{xz}{2R^2} - \frac{\left((1-R)^2 + z^2\right)y}{4R^2}\right]\boldsymbol{e}_x + \left[\frac{yz}{2R^2} + \frac{\left((1-R)^2 + z^2\right)x}{4R^2}\right]\boldsymbol{e}_y - \frac{1}{2}\log(R)\boldsymbol{e}_z.$$

14

The initial conditions are taken as $\boldsymbol{x}_0 = [1.05, 0, 0]^\top, \boldsymbol{v}_0 = [0, 4.816\text{e-}4, -2.059\text{e-}3]^\top$. The exact orbit projected on (R, z) space is a banana orbit, and it will turn into a transit orbit when the initial velocity is changed to $\boldsymbol{v}_0 = [0, 2 \times 4.816\text{e-}4, -2.059\text{e-}3]^\top$. The time step is also set to $h = \pi/10$. The results produced by RK4 and the **EIP-HL** method are given in Figure. 6. It can be observed that the banana orbit by RK4 gradually transformed into a circulating orbit and the transit orbit deviates to the right side which is mainly due to its numerical dissipation. Again, the **EIP-HL** method can provide correct orbits as well as an exact conservation of the invariants.

From the above two tests, we can conclude that the EIP method essentially improve the numerical performance of the standard fourth-order RK method.
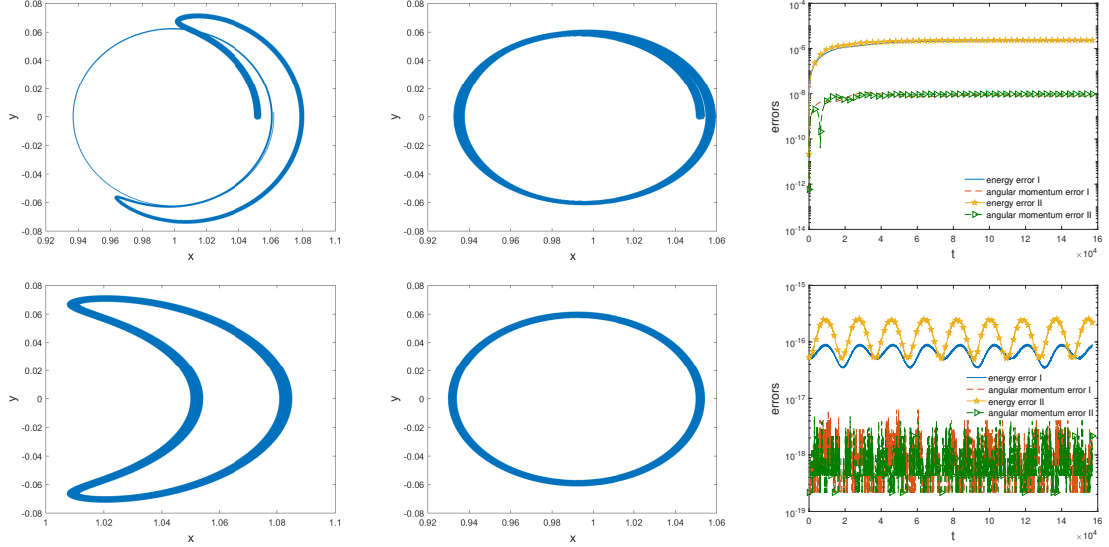

Figure 6: Numerical orbits (banana orbit and transit orbit) and errors in two invariants solved on the interval $[0, 5 \times 10^5 h]$ by RK4 (upper) and the **EIP-HL** method (bottom). The legends ended with I and II in the error figures correspond to the banana orbit and transit orbit, respectively.

3.4 Example IV: the rotating Gross-Pitaevskii equation

Finally, we apply the EIP method to the PDE case to demonstrate its computational efficiency and superior behaviors in invariants preservation. Notice that it is straightforward to extend the EIP method from ODEs to PDEs in the detailed implementation, with the only requirement that the semi-discretization of our targeted PDE is still a conservative system.

Consider the dimensionless time-dependent Gross-Pitaevskii equation (GPE) with a rotating Bose-Einstein condensate (BEC)

$$i\partial_t \psi(t, \boldsymbol{x}) = \left(-\frac{1}{2}\Delta + V(\boldsymbol{x}) - \Omega L_z + \beta|\psi(t, \boldsymbol{x})|^2 \right) \psi(t, \boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{D} \subset \mathbb{R}^d, \ t \in (0, T], \tag{3.10}$$

for $d = 2, 3$, where $\psi(t, \boldsymbol{x})$ is a complex-valued condensate wave function, $V(\boldsymbol{x})$ is a real-valued potential function. The parameter β is the nonlinearity strength representing the interaction between atoms of the condensate. Ω is the angular velocity and L_z is the z-component of the angular momentum defined by

$$L_z = -i(x\partial_y - y\partial_x).$$

In the following experiments, we test the EIP method for the GPE under periodic boundary conditions in both 2D and 3D cases, and the spatial discretization is uniformly taken as the Fourier pseudospectral method (see [18, 52] and references therein). Consider the 2D case for illustration, the resulting semi-discrete

scheme can be written as

$$i\frac{d}{dt}\psi_{jk} = \left(-\frac{1}{2}\Delta_h + V_{jk} - \Omega L_z^h + \beta|\psi_{jk}|^2\right)\psi_{jk}, \quad 0 \le j \le J-1, 0 \le k \le K-1, \tag{3.11}$$

where i, j are the grid indexes and J, K are the partition numbers with respect to x and y directions. ψ_{jk} corresponds to the approximation at the grid point. The discretization of Laplace and angular momentum are defined as

$$\Delta_h\psi_{jk} = (D_2^x\psi + \psi D_2^y)_{jk}, \quad L_z^h\psi_{jk} = -i(X\psi D_1^y - D_1^x\psi Y)_{jk},$$

where D_m^x, D_m^y, $m = 1, 2$ are the mth-order spectral differentiation matrices for x and y directions, respectively. $X = \text{diag}(x_0, x_1, \cdots, x_{J-1})$, and $Y = \text{diag}(y_0, y_1, \cdots, y_{K-1})$. It has be proved in [21] that the above semi-discrete scheme (3.11) possesses the mass conservation law

$$\frac{d}{dt}M(t) = 0, \quad \text{with} \quad M(t) = \|\psi\|_h^2, \tag{3.12}$$

and the energy conservation law

$$\frac{d}{dt}E(t) = 0, \quad \text{with} \quad E(t) = \frac{1}{2}\|\nabla_h\psi\|_h^2 + (V, |\psi|^2) - \Omega(L_z^h\psi, \psi) + \frac{\beta}{2}\|\psi\|_h^4. \tag{3.13}$$

Here, the inner product and the discrete norm for the 2D case are defined as

$$(u, v) = h_x h_y \sum_{j=0}^{J-1}\sum_{k=0}^{K-1} u_{jk}\bar{v}_{jk}, \quad \|u\|_h = \sqrt{(u, u)}, \quad \|\nabla_h u\|_h = \sqrt{-(\Delta_h u, u)},$$

where h_x, h_y are the spatial grid sizes. In the following tests, we denote **EIP-M**, **EIP-E** and **EIP-ME** for the EIP methods to preserve the mass, energy and both, respectively.

3.4.1 Computational efficiency

When $V(\boldsymbol{x}) = 0$ and $\Omega = 0$, the GPE (3.10) reduces to the classic nonlinear Schrödinger (NLS) equation, which admits an analytical solution as

$$\psi = A\exp\left(i(\kappa_1 x + \kappa_2 y - \omega t)\right), \quad \omega = \frac{1}{2}(\kappa_1^2 + \kappa_2^2) + \beta A^2.$$

Thus, we can test the computational efficiency of the EIP methods conveniently. For comparisons, we also present three kinds of conservative schemes for the NLS equation as follows:

- fully-implicit Crank-Nicolson scheme (**CN**) [57, 29]:

$$i\frac{\psi_{jk}^{n+1} - \psi_{jk}^n}{h} = \left(-\frac{1}{2}\Delta_h + V_{jk} - \Omega L_z^h\right)\psi_{jk}^{n+1/2} + \frac{\beta}{2}\left(|\psi_{jk}^{n+1}|^2 + |\psi_{jk}^n|^2\right)\psi_{jk}^{n+1/2};$$

- linearly-implicit central difference scheme (**LIC**) [61]:

$$i\frac{\psi_{jk}^{n+1} - \psi_{jk}^{n-1}}{2h} = \left(-\frac{1}{2}\Delta_h + V_{jk} - \Omega L_z^h\right)\frac{\psi_{jk}^{n+1} + \psi_{jk}^{n-1}}{2} + \frac{\beta}{2}|\psi_{jk}^n|^2\left(\psi_{jk}^{n+1} + \psi_{jk}^{n-1}\right);$$

- linearly-implicit Crank-Nicolson scheme based on the SAV approach (**SAV/CN**) [53, 24].

Though **LIC** and **SAV/CN** are both linearly implicit, the corresponding algebraic systems are distinct. **LIC** has a variable coefficient matrix while **SAV/CN** has a constant one, which represent two typical forms of linearly implicit methods.

First, we present the convergence test for the EIP method and the above three methods. We can see from Figure. 7 that all the methods exhibit correct order of accuracy (the line in the left plot overlay each other for the fourth-order EIP methods). Among the three implicit methods, **CN** is most time consuming due to the nonlinear iterations, while the computational cost of **SAV/CN** is the cheapest. Though the underlying method of the EIP methods is RK4 with four stages in the Butcher tabular which seems to have more function evaluations, due to their full explicitness, they uniformly give a better performance on the computational efficiency than the presented three methods. Furthermore, CPU times of the single invariant-preserving methods, i.e., **EIP-M** and **EIP-E**, are comparable while the **EIP-ME** method spends a little more time than the formers under the same solution error.
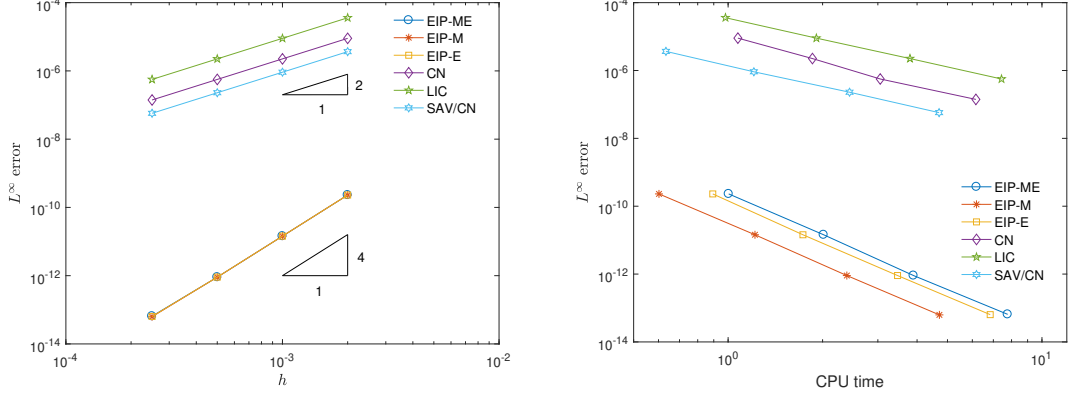

Figure 7: Accuracy tests (left) and computational efficiency (right) for various methods.

3.4.2 Dynamics of a rotating BEC

In the following experiments, we apply the EIP methods on the simulations of dynamics of a rotating BEC for the GPE (3.10). Firstly, we test the convergence of the EIP methods with different angular velocity Ω based on the approach (3.1). We set $\mathcal{D} = [-2, 2]^2$, $V(\boldsymbol{x}) = \frac{1}{2}(x^2 + y^2)$ and $\beta = 1$. The initial condition is taken as $\psi_0 = \frac{2}{\sqrt{\pi}}(x + iy)\exp(-8(x^2 + y^2))$. Table. 5 lists the accuracy results for the three methods where $h_0 = 0.0002$ is the initial time step. It is clearly that all the methods can also achieve an excepted convergence order, independent of the angular velocity Ω.

	h	EIP-M		EIP-E		EIP-ME	
		L^∞-error	order	L^∞-error	order	L^∞-error	order
	h_0						
	$h_0/2$	9.0794e-11		9.0712e-11		9.0696e-11	
$\Omega = 0$	$h_0/4$	5.6754e-12	3.9998	5.6727e-12	3.9992	5.6722e-12	3.9990
	$h_0/8$	3.5474e-13	3.9999	3.5454e-13	4.0000	3.5443e-13	4.0004
	h_0						
	$h_0/2$	9.3456e-11		9.3455e-11		9.3448e-11	
$\Omega = 0.5$	$h_0/4$	5.8442e-12	3.9992	5.8441e-12	3.9992	5.8439e-12	3.9992
	$h_0/8$	3.6541e-13	3.9994	3.6528e-13	3.9999	3.6539e-13	3.9994
	h_0						
	$h_0/2$	1.1316e-10		1.1315e-10		1.1309e-10	
$\Omega = 0.9$	$h_0/4$	7.0722e-12	4.0001	7.0715e-12	4.0000	7.0698e-12	3.9996
	$h_0/8$	4.4213e-13	3.9996	4.4224e-13	3.9991	4.4220e-13	3.9989

Table 5: Errors between two adjacent time steps and the corresponding orders with three different Ω for the EIP methods. The partition numbers in x and y directions are set to $J = K = 128$ and the computational time is $t = 0.5$.

Next, we consider the dynamics of vortex lattices in rotating BECs with $\Omega = 3.5$, $\beta = 1000$. The domain $\mathcal{D} = [-10, 10]^2$, the partition numbers $J = K = 256$ and $h = 0.0002$. The initial datum is chosen as the L^2-normalized ground state eigenvector of the Gross-Pitaevskii operator $G_0(v) := \left(-\frac{1}{2}\Delta + V_0(\boldsymbol{x}) - \Omega L_z + \beta|v|^2\right) v$ where the potential function V_0 is set to a quadratic-plus-quartic potential [3] as

$$V_0(\boldsymbol{x}) = \frac{1-\alpha}{2}(\gamma_x^2 x^2 + \gamma_y^2 y^2) + \frac{\kappa}{4}(\gamma_x^2 x^2 + \gamma_y^2 y^2)^2, \tag{3.14}$$

with $\gamma_x = \gamma_y = 1$, $\alpha = 1.2$ and $\kappa = 0.3$. The practical computation of the ground state is done by the Matlab package GPELab [1, 2]. This stationary state is a circular ring with many uniformly distributed vortices (see the first plot in Figure. 8). To simulate the dynamics in this ring BEC, we perturb the parameter $\kappa = 0.3$ to 0.7 and generate a typical example of a fast rotating BEC which demands high-precision numerical methods to capture the movement of each vortex. Figure. 8 displays the snapshots of solutions obtained by the **EIP-M** method, and the solutions computed by the **EIP-E** and **EIP-ME** methods look the same. We can observe a complex dynamics in the ring BEC. All the vortices exhibit a clockwise rotation and the number of which is also conserved during the simulation. We further demonstrate the long-time behavior by carrying out a larger time period $t = 10$. As we can see from Figure. 9, the error in the mass or energy invariant is preserved exactly by the **EIP-M** or **EIP-H** method respectively, while both errors reach machine accuracy by the **EIP-ME** method.
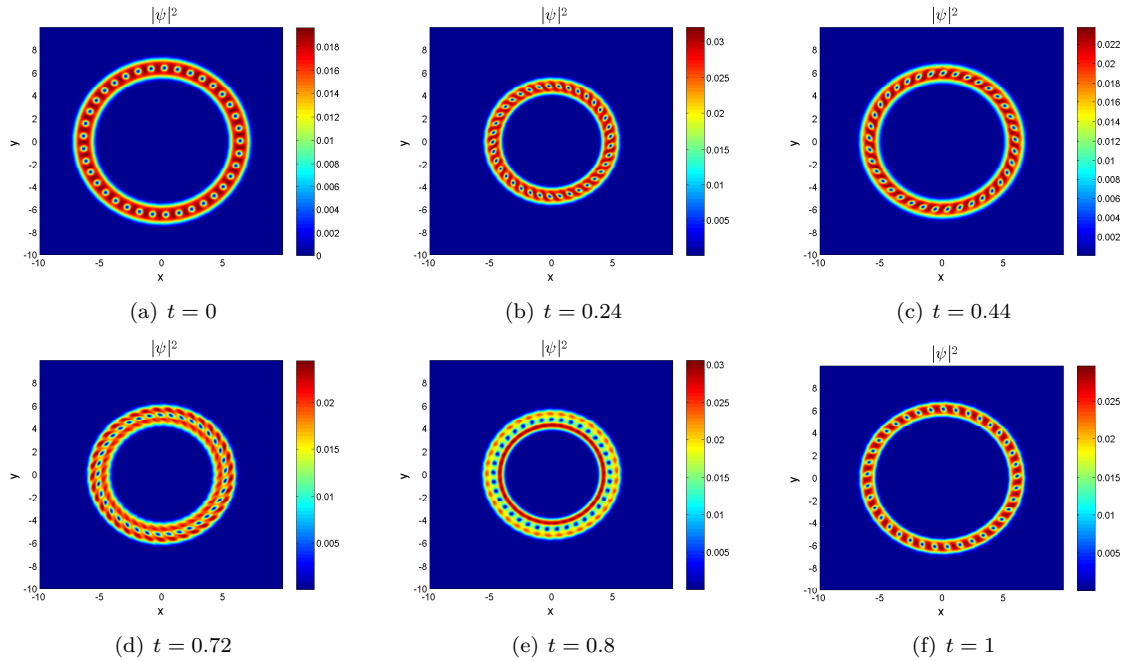
(a) $t = 0$ (b) $t = 0.24$ (c) $t = 0.44$

(d) $t = 0.72$ (e) $t = 0.8$ (f) $t = 1$

Figure 8: Snapshots of the density $|\psi|^2$ computed by the **EIP-M** method for the ring BEC.

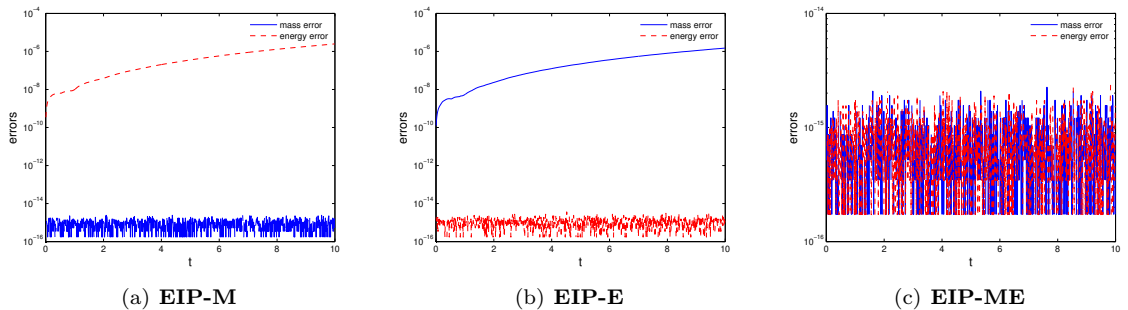(a) **EIP-M**   (b) **EIP-E**   (c) **EIP-ME**

Figure 9: Errors in two invariants computed by the EIP methods for the ring BEC.

Finally, we consider the EIP methods for the dynamics of vortex lines in a 3D rotating BEC. The ground state solution is obtained similarly as the above 2D experiment with computational domain $\mathcal{D} = [-10, 10] \times [-10, 10] \times [-15, 15]$, and the parameters are chosen as $\Omega = 0.7$, $\beta = 400$. The partition numbers $J = K = 64$ and the time step $h = 0.005$. The potential function V_0 in the Gross-Pitaevskii operator becomes a quadratic potential as

$$V_0(\boldsymbol{x}) = \frac{1}{2}(\gamma_x^2 x^2 + \gamma_y^2 y^2 + \gamma_z^2 z^2),$$

with $\gamma_x = \gamma_y = 1, \gamma_z = 1/2$. The 10^{-3}-isosurface of the modulus of the corresponding ground state solution is drawn in the first plot of Figure. 10 where four vortex lines can be observed clearly. By perturbing the angular velocity $\Omega = 0.7$ to 0.9, we can generate dynamics of the four vortex lines by the **EIP-M** method, and the other methods have similar results. Figure. 10 also presents the snapshots of 10^{-3}-isosurfaces at different times where the four vortex lines rotate clockwise around the z- axis from above. This can be confirmed again by the slices along the surface $z = 0$ in Figure. 11. Moreover, as demonstrated in Figure. 12 the errors in the mass and energy exhibit a similar tendency as that in the 2D dynamics of vortex lattices.
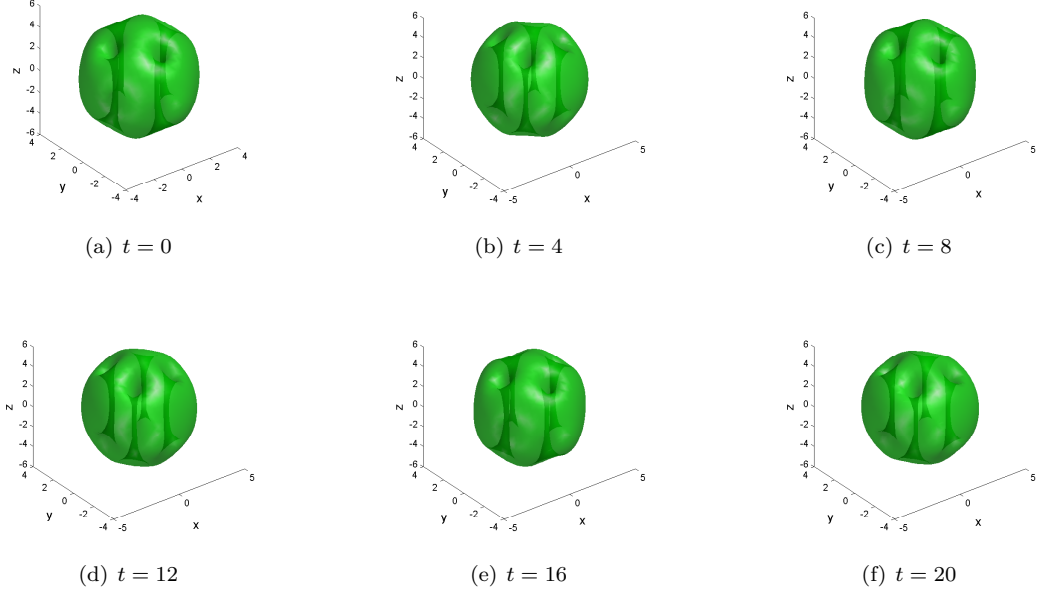
(a) $t = 0$ (b) $t = 4$ (c) $t = 8$

(d) $t = 12$ (e) $t = 16$ (f) $t = 20$

Figure 10: Snapshots of 10^{-3}-isosurfaces of $|\psi|^2$ at different times computed by the **EIP-M** method for the dynamics of vortex lines.
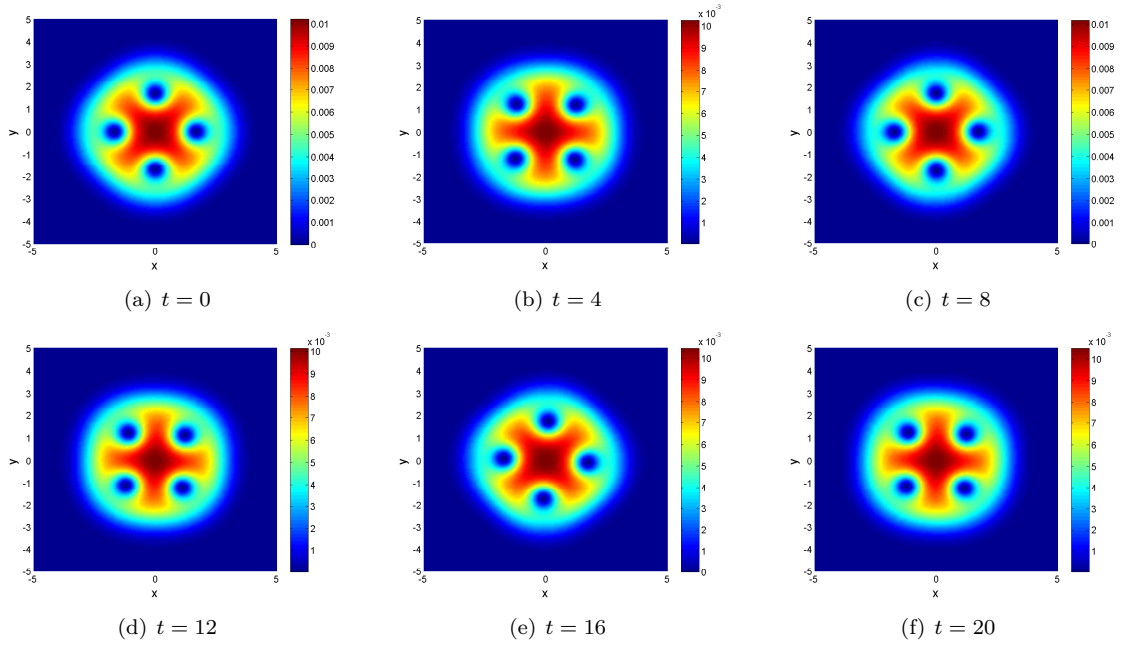

(a) $t = 0$ (b) $t = 4$ (c) $t = 8$

(d) $t = 12$ (e) $t = 16$ (f) $t = 20$

Figure 11: Slices along the surface $z = 0$ of $|\psi|^2$ at different times computed by the **EIP-M** method for the dynamics of vortex lines, intercepted on the domain $[-5, 5]^2$.
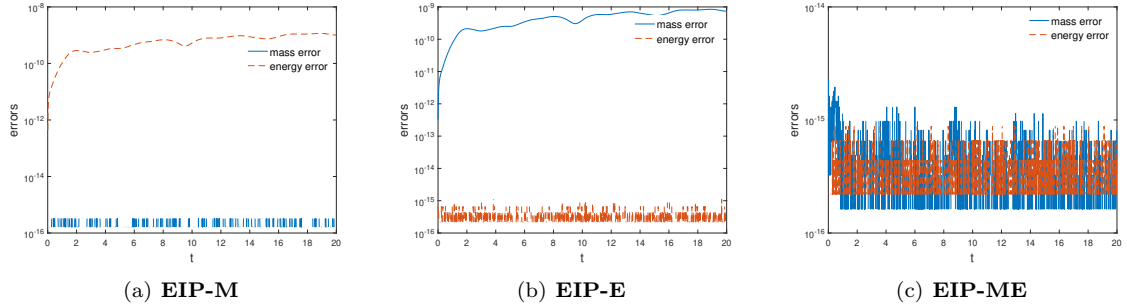
Figure 12: Errors in two invariants computed by the EIP methods for the dynamics of vortex lines.

4 Conclusion

In this paper, we propose a novel explicit and practically invariants-preserving method for conservative systems, which can be viewed as a further simplification of the standard projection method. We prove that such simplification does not affect the numerical behaviors, that is, the order of accuracy retains the same as the underlying Runge-Kutta method and the invariants are preserved to round-off errors in practice. The detailed implementations are also provided to show the flexibility in preservation of single and multiple invariants, and in the generalization to high-dimensional problems. Extensive numerical experiments are carried out for both ODEs and PDEs to verify the theoretical analysis and demonstrate the efficiency and conservative properties of our method.

Notice that the underlying method is taken as an explicit Runge-Kutta method throughout the entire discussion. However, the proposed invariants-preserving method is rather general such that any existing efficient numerical methods can also be brought into our framework to achieve invariants preservation in practice and may essentially improve the numerical performance of the original methods.

Acknowledgments

References

[1] X. Antoine and R. Duboscq. GPELab, a Matlab toolbox to solve Gross-Pitaevskii equations I: computation of stationary solutions. *Comput. Phys. Commun.*, 00:1–37, 2014.

[2] X. Antoine and R. Duboscq. GPELab, a Matlab toolbox to solve Gross-Pitaevskii equations II: dynamics and stochastic simulations. *Comput. Phys. Commun.*, 00:1–29, 2015.

[3] X. Antoine and R. Duboscq. Modeling and computation of Bose-Einstein condensates: Stationary states, nucleation, dynamics, stochasticity. In *Nonlinear Optical and Atomic Systems*, volume 2146 of *Lecture Notes in Mathematics*. Springer, 2015.

[4] P. Betsch and P. Steinmann. Inherently energy conserving time finite elements for classical mechanics. *J. Comput. Phys.*, 160:88–116, 2000.

[5] L. Brugnano, M. P. Calvo, J. I. Montijano, and L. Rández. Energy-preserving methods for Poisson systems. *J. Comput. Appl. Math.*, 236:3890–3904, 2012.

[6] L. Brugnano and F. Iavernaro. Line integral methods which preserve all invariants of conservative problems. *J. Comput. Appl. Math.*, 236:3905–3919, 2012.

[7] L. Brugnano and F. Iavernaro. *Line Integral Methods for Conservative Problems.* Chapman and Hall/CRC, Boca Raton, FL, 2016.

[8] L. Brugnano, F. Iavernaro, and D. Trigiante. Hamiltonian Boundary Value Methods (Energy Preserving Discrete Line Integral Methods). *J. Numer. Anal. Industr. Appl. Math.*, 5:17–37, 2010.

[9] L. Brugnano, F. Iavernaro, and D. Trigiante. A two-step, fourth-order method with energy preserving properties. *Comput. Phys. Commun.*, 183:1860–1868, 2012.

[10] L. Brugnano, J. Montijiano, and L. Rández. High-order energy-conserving line integral methods for charged particle dynamics. *J. Comput. Phys.*, 396:209–227, 2019.

[11] J. Cai and J. Shen. Two classes of linearly implicit local energy-preserving approach for general multi-symplectic Hamiltonian PDEs. *J. Comput. Phys.*, 401:108975, 2020.

[12] W. Cai, C. Jiang, Y. Wang, and Y. Song. Structure-preserving algorithms for the two-dimensional sine-Gordon equation with Neumann boundary conditions. *J. Comput. Phys.*, 395:166–185, 2019.

[13] W. Cai, H. Li, and Y. Wang. Partitioned averaged vector field methods. *J. Comput. Phys.*, 370:25–42, 2018.

[14] M. Calvo, D. Hernández-Abreu, J. Montijiano, and L. Rández. On the preservation of invariants by explicit Runge-Kutta methods. *SIAM J. Sci. Comput.*, 28:868–885, 2006.

[15] M. Calvo and J. Sanz-Serna. High-order symplectic Runge-Kutta-Nyström methods. *SIAM J. Sci. Comput.*, 14:1237–1252, 1993.

[16] C. Chen, J. Hong, and D. Jin. Modified averaged vector field methods preserving multiple invariants for conservative stochastic differential equations. *BIT Numer. Math.*, 60:917–957, 2020.

[17] C. Chen, J. Hong, C. Sim, and K. Sonwu. Energy and quadratic invariants preserving (EQUIP) multi-symplectic methods for Hamiltonian wave equations. *J. Comput. Phys.*, 418:109599, 2020.

[18] J. Chen and M. Qing. Multi-symplectic Fourier pseudospectral method for the nonlinear Schrödinger equation. *Electr. Trans. Numer. Anal.*, 12:193–204, 2001.

[19] D. Cohen and E. Hairer. Linear energy-preserving integrators for Poisson systems. *BIT*, 51:91–101, 2011.

[20] G. Cooper. Stability of Runge-Kutta methods for trajectory problems. *IMA J. Numer. Anal.*, 7:1–13, 1987.

[21] J. Cui, W. Cai, and Y. Wang. A linearly-implicit and conservative Fourier pseudo-spectral method for the 3D Gross-Pitaevskii equation with angular momentum rotation. *Comput. Phys. Commun.*, 253:107160, 2020.

[22] M. Dahlby and B. Owren. A general framework for deriving integral preserving numerical methods for PDEs. *SIAM J. Sci. Comput.*, 33:2318–2340, 2011.

[23] N. Del Buono and C. Mastroserio. Explicit methods based on a class of four stage fourth order Runge-Kutta methods for preserving quadratic laws. *J. Comput. Appl. Math.*, 140:231–243, 2002.

[24] Y. Fu, W. Cai, and Y. Wang. A structure-preserving algorithm for the fractional nonlinear Schrödinger equation based on the SAV approach. *arXiv:1911.07379*, 2019.

[25] D. Furihata. Finite difference schemes for $\partial u/\partial t = (\partial/\partial x)^\alpha \delta G/\delta u$ that inherit energy conservation or dissipation property. *J. Comput. Phys.*, 156:181–205, 1999.

[26] D. Furihata. Finite-difference schemes for nonlinear wave equation that inherit energy conservation property. *J. Comput. Appl. Math.*, 134:37–57, 2001.

[27] D. Furihata and T. Matsuo. *Discrete Variational Derivative Method. A Structure-Preserving Numerical Method for Partial Differential Equations.* Chapman Hall/CRC, Boca Raton, FL, 2011.

[28] Y. Gong, J. Gao, and Y. Wang. High order Gauss-Seidel schemes for charged particle dynamics. *Discrete Contin. Dyn. Syst. Ser. B*, 23:573–585, 2018.

[29] Y. Gong, Q. Wang, Y. Wang, and J. Cai. A conservative Fourier pseudo-spectral method for the nonlinear Schrödinger equation. *J. Comput. Phys.*, 328:354–370, 2017.

[30] Y. Gong, J. Zhao, X. Yang, and Q. Wang. Fully discrete second-order linear schemes for hydrodynamic phase field models of binary viscous fluid flows with variable densities. *SIAM J. Sci. Comput.*, 40:B138–B167, 2018.

[31] O. Gonzalez. Time integration and discrete Hamiltonian systems. *J. Nonlinear Sci.*, 6:449–467, 1996.

[32] E. Hairer. Energy preserving variant of collocation methods. *J. Numer. Anal. Ind. Appl. Math.*, 5:73–84, 2010.

[33] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations.* Springer-Verlag, Berlin, 2nd edition, 2006.

[34] E. Hairer, S. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I.* Springer-Verlag, Berlin, 3rd edition, 2008.

[35] A. Harten, P. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation law. *SIAM Rev.*, 25:35–61, 1983.

[36] Y. He, Y. Sun, J. Liu, and H. Qin. Higher order volume-preserving schemes for charged particle dynamics. *J. Comput. Phys.*, 305:172–184, 2016.

[37] Y. He, Z. Zhou, Y. Sun, J. Liu, and H. Qin. Explicit K-symplectic algorithms for charged particle dynamics. *Phys. Lett. A*, 381:568–573, 2017.

[38] T. Itoh and K. Abe. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *J. Comput. Phys.*, 77:85–102, 1988.

[39] C. Jiang, Y. Gong, W. Cai, and Y. Wang. A linearly implicit structure-preserving scheme for the Camassa-Holm equation based on multiple scalar auxiliary variables approach. *J. Sci. Comput.*, 83:20, 2020.

[40] C. Jiang, Y. Wang, and Y. Gong. Arbitrarily high-order energy-preserving schemes for the Camassa-Holm equation . *Appl. Numer. Math.*, 151:85–97, 2020.

[41] C. Jiang, Y. Wang, and Y. Gong. Explicit high-order energy-preserving methods for general Hamiltonian partial differential equations. *arXiv:2001.00774*, 2020.

[42] H. Li, Y. Wang, and M. Qin. A sixth order averaged vector field method. *J. Comput. Math.*, 34:479–498, 2016.

[43] S. Li and L. Vu-Quoc. Finite difference calculus invariant structure of a class of algorithms for the nonlinear Klein-Gordon equation. *SIAM. J. Numer. Anal.*, 32:1839–1875, 1995.

[44] T. Li and B. Wang. Efficient energy-preserving methods for charged-particle dynamics. *Appl. Math. Comput.*, 361:703–714, 2019.

[45] T. Matsuo. New conservative schemes with discrete variational derivatives for nonlinear wave equations. *J. Comput. Appl. Math.*, 203:32–56, 2007.

[46] T. Matsuo and D. Furihata. Dissipative or conservative finite-difference schemes for complex-valued nonlinear partial differential equations,. *J. Comput. Phys.*, 171:425–447, 2001.

[47] R. McLachlan, G. Quispel, and N. Robidoux. Geometric integration using discrete gradients. *Phil. Trans. R. Soc. Lond. A*, 357:1021–1045, 1999.

[48] Z. Qiao, S. Sun, T. Zhang, and Y. Zhang. A new multi-component diffuse interface model with Peng-Robinson equation of state and its scalar auxiliary variable (SAV) approach. *Commun. Comput. Phys.*, 26:1597–1616, 2019.

[49] H. Qin, S. Zhang, J. Xiao, J. Liu, Y. Sun, and W. Tang. Why is Boris algorithm so good? *Phys. Plasma*, 20:084503, 2013.

[50] M. Qin and W. Zhu. Canonical Runge-Kutta-Nyström (RKN) methods for second order ordinary differential equations. *Comput. Math. Appl.*, 22:85–95, 1991.

[51] G. Quispel and D. McLaren. A new class of energy-preserving numerical integration methods. *J. Phys. A: Math. Theor.*, 41:045206, 2008.

[52] J. Shen, T. Tang, and L.-L. Wang. *Spectral Methods-Algorithms, Analysis and Applications*. Springer-Verlag, Berlin, 2011.

[53] J. Shen, J. Xu, and J. Yang. The scalar auxiliary variable (SAV) approach for gradient flows. *J. Comput. Phys.*, 353:407–416, 2018.

[54] J. Shen, J. Xu, and J. Yang. A new class of efficient and robust energy stable schemes for gradient flows. *SIAM Rev.*, 61:474–506, 2019.

[55] W. Tang and Y. Sun. Time finite element methods: a unified framework for numerical discretizations of ODEs. *Appl. Math. Comput.*, 219:2158–2179, 2012.

[56] M. Tao. Explicit high-order symplectic integrators for charged particles in general electromagnetic fields. *J. Comput. Phys.*, 327:245–251, 2016.

[57] T. Wang, B. Guo, and Q. Xu. Fourth-order compact and energy conservative difference schemes for the nonlinear Schrödinger equation in two dimensions. *J. Comput. Phys.*, 243:382–399, 2013.

[58] X. Wu, B. Wang, and W. Shi. Efficient energy-preserving integrators for oscillatory Hamiltonian systems. *J. Comput. Phys.*, 235:587–605, 2013.

[59] X. Yang. Linear, first and second-order, unconditionally energy stable numerical schemes for the phase field model of homopolymer blends. *J. Comput. Phys.*, 327:294–316, 2016.

[60] X. Yang, J. Zhao, and Q. Wang. Numerical approximations for the molecular beam epitaxial growth model based on the invariant energy quadratization method. *J. Comput. Phys.*, 333:104–127, 2017.

[61] F. Zhang, V. Pérez-Garacía, and L. Vázquez. Numerical simulation of nonlinear Schrödinger systems: A new conservative scheme. *App. Math. Comput.*, 71:165–177, 1995.

[62] H. Zhang, X. Qian, J. Yan, and S. Song. Highly efficient invariant-conserving explicit Runge-Kutta schemes for nonlinear Hamiltonian differential equations. *J. Comput. Phys.*, 418:109598, 2020.

[63] R. Zhang, J. Liu, Y. Tang, Q. Hong, J. Xiao, and B. Zhu. Canonicalization and symplectic simulation of the gyrocenter dynamics in time-independent magnetic fields. *Phys. Plasma*, 21:032504, 2014.

Appendix A Initial datum of the solar system

Table 6: Initial datum of planets in the solar system.

Planet	Initial position	Initial velocity	G*mass
Mercury	1.563021412664830e+10	-5.557001175482630e+04	2.203209e+13
	4.327888220902108e+10	1.840863017229157e+04	
	2.102123103174893e+09	6.602621285552567e+03	
Venus	-9.030189258080004e+10	-1.907374632532257e+04	3.248586e+14
	5.802615456116644e+10	-2.963461693326599e+04	
	6.006513603716755e+09	6.946391255404438e+02	
Earth	-1.018974476358996e+11	-2.201749257051057e+04	3.986004e+14
	1.065689158175689e+11	-2.071074857788741e+04	
	-3.381951053601424e+06	1.575245213712245e+00	
Mars	-2.443763125844157e+11	-3.456935754608896e+03	4.282830e+13
	4.473211564076996e+10	-2.176307370133160e+04	
	6.935657388967808e+09	-3.711433859326417e-02	
Jupiter	-2.3516546827532200e+11	-1.262559929908801e+04	1.266865e+17
	7.421837640432589e+11	-3.332552395475581e+03	
	2.179850895804323e+09	2.962741332356101e+02	
Saturn	-1.011712827283427e+12	6.507898648442419e+03	3.793120e+16
	-1.077496255617324e+12	-6.640809674126991e+03	
	5.901251900068215e+10	-1.434198106014633e+02	
Uranus	2.934840841770302e+12	-1.433852081777671e+03	5.793966e+15
	6.048399137411513e+11	6.347897341634990e+03	
	-3.576451387567792e+10	4.228261484335974e+01	
Neptune	4.055112581124043e+12	2.275119229131818e+03	6.835107e+15
	-1.914578873112663e+12	4.942356914027413e+03	
	-5.400973716179796e+10	-1.548950389954096e+02	
Pluto	9.514009594170194e+11	5.431808363374300e+03	8.72400e+11
	-4.776029500570151e+12	-2.387056445508962e+01	
	2.358627841705075e+11	-1.551877289694926e+03	