symmetry

Article

MGPS: Midpoint-Series Group Preserving Scheme for Discretizing Nonlinear Dynamics

Zhenxing Xu [1,2] and Jinzhao Wu [1,2,3,*]

[1] Chengdu Institute of Computer Application, Chinese Academy of Sciences, Chengdu 610041, China; xuzhenxing17@mails.ucas.ac.cn
[2] School of Computing Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China
[3] Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Guangxi University for Nationalities, Nanning 530006, China
* Correspondence: himrwujinzhao@163.com

Abstract: In this article, we propose a new computational method for initial value problems in ordinary differential equations. The algorithm combines the merits of the group preserving scheme (GPS), which has the ability of avoiding possible spurious solutions utilizing the inherent symmetry group, the cone structure of the nonlinear dynamical system, and the classic midpoint rule. The error and stability analysis are included to demonstrate the convergence properties of the presented method. From the numerical experimental results we obtained, the algorithm can be said to be computationally effective and possesses better simulation ability generally. Meanwhile, it works well with the periodic Hamiltonian system.

Keywords: hybrid systems; ODEs; group preserving scheme; midpoint rule; symmetry group; MGPS

Citation: Xu, Z.; Wu, J. MGPS: Midpoint-Series Group Preserving Scheme for Discretizing Nonlinear Dynamics. *Symmetry* **2022**, *14*, 365. https://doi.org/10.3390/sym14020365

Academic Editor: Calogero Vetro

Received: 31 December 2021
Accepted: 8 February 2022
Published: 12 February 2022

1. Introduction

Systems of ordinary differential equations (ODEs) are most versatile for characterizing the behavior of dynamical systems. They are ubiquitous in fields such as science and engineering as well as physical processes, biology, etc. ODEs, partial differential equations (PDEs), as well as other types of equations can then be used to model systems in continuous time (e.g., integro-differential or delay equations). In discrete time, they can be depicted by state machines, Petri nets, finite automata, and so on. As an example, consider the most important hybrid system. Figure 1 illustrates a simple hybrid system: the self-regulating thermostat, which is well characterized by a hybrid automaton. There are two states, *on* and off, denoted by q_0 and q_1, respectively. The temperature is symbolized by the real-valued variable x. The two ODEs are used to portray the evolution of the continuous operation in the state q_0 cooling and q_1 state warming, separately. Initially, the system is in the q_0 state, i.e., off, and the temperature is $x = 20$, which drops depending on the flow condition $\dot{x} = -0.1x$. As the temperature falls below 19, the system state switches from q_0 to the *on* state q_1. The heater opens and the temperature increases depending on the flow condition $\dot{x} = 5 - 0.1x$. The state switches from q_1 to the off q_0 if the temperature exceeds 21. Subject to the invariant term $x \geq 18$, the heater is triggered when the temperature falls to 18 degrees at the least.

From this example, one crucial step for various tasks is solving the flow condition that characterizes the continuously dynamic evolution. Formally, $\phi_q : X \times \mathbb{R} \to X$ denotes a flow on $X \subset \mathbb{R}^n$ in state q, which is $\dot{x} = f(x)$ about variable X. $Inv_q \subseteq X$ is the domain of permitted evolution in q. $\Delta_i \in \mathbb{R}^+$ is the operation time. $\gamma_i : [0, \Delta_i] \to X$ denotes the curve. For all $t \in [0, \Delta_i]$, $\gamma_i(t) = \phi_{q_i}(\gamma_i(0), t)$ and $\gamma_i(t) \in Inv_{q_i}$. The subsequent sections cover how our proposed approach can be used to obtain the discretized flow pipes.

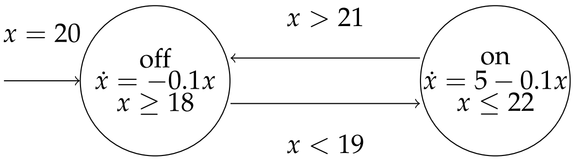

Figure 1. Hybrid automata for temperature controller.

Many successes have been obtained in linear systems with various mathematical models such as convex polyhedra [1], griddy polyhedra [2], zonetope [3], ellipsoidal [4], and support function [5] used for approximate simulations expressing the reachable set of hybrid systems. Many traditional tools have emerged for verifying linear hybrid systems, such as HYTECH [6], D/DT [7], CheckMate [8], PHAVER [9], SpaceEx [10], etc.

Nevertheless, for a large number of nonlinear continuous systems, no analytical solution exists and, even if it did, it would be prohibitively expensive to work out and understand its properties. When dealing with continuous and hybrid systems observed over real numbers, the term “exact” is insufficient because there may be numerical errors in observation, noise, and other nonidealities [9,11] over-approximating complex dynamics with simple rectangular flow constraints on the dotted variables, which is operative under some relatively simple nonlinear dynamics. However, the mass of the approximation may be too rough to meet the needs of security verification with comparatively high accuracy requirements.

Applying numerical analysis methods to discrete nonlinear flow constraints is also a proven method. The numerical approximation ”captures“ the behavior of the analytical solution, which is one of the most desirable properties of numerical schemes. Many classical numerical methods have been proposed for this task in the past, such as Euler and Heun, Runge–Kutta, and some others. Normally, the first-order numerical method is easy to implement and low in complexity, but the global error is large and unsuitable for systems with high accuracy requirements. Conversely, higher-order methods such as the Runge–Kutta method can produce numerical solutions with higher accuracy but at the expense of complexity. Furthermore, the spurious solution problem persists in these traditional numerical methods.

As a valuable class of numerical analysis schemes, geometric numerical integration methods serve a vital role since they respect the structure of the problem, which is of a geometric nature. In connection with the geometric characteristics of specific systems, mathematicians and physicists have formulated various numerical algorithms for structure preservation. In [12], a nonstandard finite-difference approach was developed for the Lotka–Volterra system, an essential predator–prey interaction mathematical model. Hamiltonian systems are dominated by the Hamiltonian equations and are prevalent in modern physics in an extensive variety of problems. To preserve the contact transformation property of the Hamilton equations, Ref. [13] devised methods of integration of Hamilton equations and presented many important findings on numerical symplectic integration for the first time, including the symplectic Euler method. Ref. [14] later concludes that the leapfrog method is symplectic. Refs. [15,16] systematically proposed the Hamiltonian equations and the Hamiltonian algorithm (i.e., the symplectic geometric algorithm), and opened up a new field of Hamiltonian algorithms. Ref. [17] discussed the discrete analogue of the gradient of a function and showed how discrete gradients can be used in the numerical integration of ODEs. Ref. [18] implemented a modified discrete gradient method for the purpose of preserving (almost completely) small oscillation periods of any time step. Ref. [19] proposed locally exact discrete gradient numerical integrators, which substitute a function for its step size and exactly preserve the energy integral. Ref. [20] goes further to develop the results of the above literature to the Nth order. Refs. [21,22] presented a new geometric integrator, which combines the implicit midpoint rule (IMR, i.e., second-order implicit Runge–Kutta method) [23] with an appropriate spatial discretization in the PDE cases. It can retain the energy properties of the solution [24], and the function $\mathbf{f}$ needs to be evaluated only once at each step (single-step method, unlike many other Runge–Kutta methods).

Unfortunately, the foregoing numerical approaches either suffer from so-called spurious solutions and ghost fixed points, or are feasible solely for certain systems. In order to avoid the flaws, ref. [25] firstly devised the group preserving scheme (GPS) by utilizing inherent symmetry group and the (null) cone structure of the nonlinear dynamical system. For the aim of improving the accuracy of GPS, an enhanced GPS is proposed in [26]. In this paper, combining the merits of the implicit midpoint rule and GPS, we construct a novelty midpoint group preserving scheme (MGPS). Compared with enhanced GPS, which is also a predictor–corrector method, MGPS overcomes some weaknesses of enhanced GPS.

- The predictor. In enhanced GPS, the Euler method with the same step size is used to calculate x_{n+1}, while MGPS uses GPS to calculate x_{n+1}, which can obtain the same benefits as the advantages of GPS over the Euler method.
- The corrector. When computing η_n^e, the enhanced GPS replaces x_n with $\bar{x}_n$; the MGPS remains unchanged and is still used to calculate η_n^e. This improves the accuracy while ensuring that the cone structure is held.

The remaining parts of this paper are structured below. Section 2 firstly introduces GPS and analyzes its stability property. Then, we devise the midpoint group preserving scheme (MGPS) and details of error and stability properties. Some numerical tests are conducted to assess the effectiveness of the proposed solver in Section 3. The numerical results are very encouraging. This demonstrates improved performance and higher accuracy compared with other well-known second-order methods present in the literature. Finally, Section 4 sums up this work and addresses the path for the future.

2. Discretization

Discretization of the nonlinear continuous dynamic evolution is a vital measure to understand the properties of the system while differential equations have no explicit solutions. Consider the following ordinary differential equation with the initial value

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t)), \mathbf{x}(t_0) = \mathbf{x}_0, \qquad t \in \mathbb{R}, \mathbf{x} \in \mathbb{R}^k \tag{1}$$

It is commonly utilized to demonstrate the attributes and behaviors of dynamic systems. $\mathbf{x}$ represents k-dimensional state vector, and $\mathbf{f}$ denotes a vector function about $\mathbf{x}$ and t. The Lipschitz condition is assumed to ensure the existence and uniqueness of the solution:

$$|\mathbf{f}(t, \mathbf{x}_1) - \mathbf{f}(t, \mathbf{x}_2)| \leq L(|\mathbf{x}_1 - \mathbf{x}_2|) \qquad \forall (t, \mathbf{x}_1), (t, \mathbf{x}_2) \in \mathbb{R} \times \mathbb{R}^k \tag{2}$$

Suppose that the running time delay Δ_i is uniformly partitioned into n steps. A sequence of time-value pairs $\{(t_j, u_j), j \in \{0, \ldots, n-1\}\}$ are generated along a solution curve. Take the Euler method as an example, we can obtain curve $\gamma_i(\frac{j}{n} \cdot \Delta_i) \in [\![Inv_{q_i}]\!]$ at each time point $t_j = \frac{j}{n} \cdot \Delta_i$, where $[\![Inv_{q_i}]\!]$ denotes the collection that meets Inv_{q_i}. The continuous trajectory is thus converted into a finite discrete-time sequence in this manner: $\eta = \langle q_1, \Delta_1 \cdot 0, \gamma_1(0) \rangle \xrightarrow{t} \langle q_1, \Delta_1 \cdot \frac{1}{n}, \gamma_1(\Delta_1 \cdot \frac{1}{n}) \rangle \xrightarrow{t} \cdots \xrightarrow{t} \langle q_1, \Delta_1 \cdot \frac{n-1}{n}, \gamma_1(\Delta_1 \cdot \frac{n-1}{n}) \rangle \xrightarrow{r_{q_1, q_2}} \langle q_2, \Delta_2 \cdot 0, \gamma_2(0) \rangle \xrightarrow{t} \cdots$.

Unfortunately, spurious solutions and ghost fixed points are broadly present in classical numerical methods and may yield unpredictable outcomes. The problem of such spurious solutions schemes is studied in [27] when first-order differential equations are computed employing the Runge–Kutta. Ref. [25] devised a group preserving scheme in the Minkowski space utilizing homogeneous coordinates to avoid the shortcomings, which serve as the foundation of our approach.

2.1. Group Preserving Scheme

The idea of GPS is that embedding Equation (1) into $k+1$ dimensional augmented dynamical system yields the equation below.

$$\frac{d}{dt}\begin{bmatrix}\mathbf{x}\\ \|\mathbf{x}\|\end{bmatrix}=\begin{bmatrix}\mathbf{0}_{k\times k} & \frac{\mathbf{f}(t,\mathbf{x})}{\|\mathbf{x}\|}\\ \frac{\mathbf{f}^{\mathbf{T}}(t,\mathbf{x})}{\|\mathbf{x}\|} & 0\end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \|\mathbf{x}\|\end{bmatrix} \tag{3}$$

The first equation in (3) is obviously identical to the original Equation (1), and with the addition of the second equation, we obtain an enhancement of $\mathbf{X} := (\mathbf{x}^{\mathbf{T}}, \|\mathbf{x}\|)^{\mathbf{T}}$ Minkowskian structure of state variables meeting the cone condition

$$\mathbf{X}^{\mathbf{T}}\mathbf{g}\mathbf{X}=0 \tag{4}$$

Herein, $\mathbf{g}$ denotes Minkowski metrics and is expressed by

$$\mathbf{g}=\begin{bmatrix}\mathbf{I_k} & \mathbf{0}_{k\times 1}\\ \mathbf{0}_{1\times k} & -1\end{bmatrix} \tag{5}$$

where $\mathbf{I_k}$ signifies the k-order identity matrix. The inference below leads to this conclusion:

$$\mathbf{X}^{\mathbf{T}}\mathbf{g}\mathbf{X}=\mathbf{x}\cdot\mathbf{x}-\|\mathbf{x}\|^2=\|\mathbf{x}\|^2-\|\mathbf{x}\|^2=0 \tag{6}$$

Then, as a result of the previously stated cone condition, Equation (3) has now become

$$\dot{\mathbf{X}}=\mathbf{A}(t,\mathbf{X})\mathbf{X} \tag{7}$$

where

$$\mathbf{A}(t,\mathbf{X}):=\begin{bmatrix}\mathbf{0}_{k\times k} & \frac{\mathbf{f}(t,\mathbf{x})}{\|\mathbf{x}\|}\\ \frac{\mathbf{f}^{\mathbf{T}}(t,\mathbf{x})}{\|\mathbf{x}\|} & 0\end{bmatrix} \tag{8}$$

It is a (local) Lie algebra of the proper orthochronous Lorentz group $\mathbf{SO}_o(k,1)$ that fulfills

$$\mathbf{A}^{\mathbf{T}}\mathbf{g}+\mathbf{g}\mathbf{A}=0 \tag{9}$$

This motivates the development of the so-called group preserving scheme with discretized mapping $\mathbf{G}$ that preserves exactly the following attributes [25]:

$$\begin{cases}\mathbf{G}^{\mathbf{T}}\mathbf{g}\mathbf{G}=\mathbf{g}\\ \det\mathbf{G}=1\\ G_0^0>0\end{cases} \tag{10}$$

G_0^0 denotes the 00-th element of $\mathbf{G}$, whereas det signifies the abbreviation for determinant. Provided that $\mathbf{X}_n$ is the value of $\mathbf{X}$ at $t=t_n$, it is required to work out $\mathbf{X}_{n+1}$ at $t=t_{n+1}$. It can be known from Equation (8) that $\mathbf{A}$ is not a constant matrix. Let $h=t_{n+1}-t_n$. Then, the GPS is formulated by

$$\mathbf{X}_{n+1}=\mathbf{G}(n)\mathbf{X}_n \tag{11}$$

$\mathbf{G}(n)\in\mathbf{SO}_o(k,1)$ denotes the group value at t_n. The rest of the issue is determining how to obtain the expression of $\mathbf{G}(n)$. The exponential mapping is adopted to calculate $\mathbf{G}(n)$ with the next expression:

$$\mathbf{G}(n)=exp[h\mathbf{A}(n)]=\begin{bmatrix}\mathbf{I}_k+\frac{(a_n-1)}{\|\mathbf{f}_n\|^2}\mathbf{f}_n\mathbf{f}_n^{\mathbf{T}} & \frac{b_n\mathbf{f}_n}{\|\mathbf{f}_n\|}\\ \frac{b_n\mathbf{f}_n^{\mathbf{T}}}{\|\mathbf{f}_n\|} & a_n\end{bmatrix} \tag{12}$$

where

$$a_n := cosh(\frac{h\|\mathbf{f}_n\|}{\|\mathbf{x}_n\|}), \quad b_n := sinh(\frac{h\|\mathbf{f}_n\|}{\|\mathbf{x}_n\|}) \tag{13}$$

The first line of the formula (11) is picked up and one obtains

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \eta^e \mathbf{f}_n \tag{14}$$

where

$$\eta^e := \frac{(a_n - 1)\mathbf{f}_n \cdot \mathbf{x}_n + b_n\|\mathbf{x}_n\|\|\mathbf{f}_n\|}{\|\mathbf{f}_n\|^2} \tag{15}$$

From $a_n > 1, \forall \mathbf{h} > 0$ and $\|\mathbf{f}_n\|\|\mathbf{x}_n\| \geq \mathbf{f}_n \cdot \mathbf{x}_n \geq -\|\mathbf{f}_n\|\|\mathbf{x}_n\|$, the inequalities can be concluded:

$$\begin{cases} \eta^e \leq \left[exp(\frac{h\|\mathbf{f}_n\|}{\|\mathbf{x}_n\|}) - 1\right]\frac{\|\mathbf{x}_n\|}{\|\mathbf{f}_n\|} \\ \eta^e \geq \left[1 - exp(-\frac{h\|\mathbf{f}_n\|}{\|\mathbf{x}_n\|})\right]\frac{\|\mathbf{x}_n\|}{\|\mathbf{f}_n\|} \end{cases} \tag{16}$$

Stable Analysis

This section will analyze the stability of the GPS (14). Regularly, the following test equation is used for analyzing numerical schemes's stability.

$$\mathbf{x}' = \lambda\mathbf{x}, \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{17}$$

whereby λ is a complex number. It is also assumed that $\mathbf{x}_0 \neq 0$, which would otherwise infer trivial zero solutions. Putting (17) into (14), one obtains

$$\begin{aligned} \mathbf{x}_1 &= \mathbf{x}_0 + \eta_0^e \mathbf{f}_0 = \mathbf{x}_0 + \eta_0^e(\lambda\mathbf{x}_0) = (1 + \lambda\eta_0^e)\mathbf{x}_0 \\ \mathbf{x}_2 &= \mathbf{x}_1 + \eta_1^e \mathbf{f}_1 = \mathbf{x}_1 + \eta_1^e(\lambda\mathbf{x}_1) = (1 + \lambda\eta_0^e)(1 + \lambda\eta_1^e)\mathbf{x}_0 \end{aligned}$$

Inductively,

$$\mathbf{x}_{n+1} = \mathbf{x}_0 \prod_{i=1}^{n} (1 + \lambda\eta_i^e) \tag{18}$$

is derived. The analytical solution $\mathbf{x} = \mathbf{x}_0 e^{\lambda t}$ ultimately converges to 0 as t goes to infinity under the restriction $Re(\lambda) < 0$. In numerical terms, it is corresponding to the situation where $\mathbf{x}_n$ becomes 0 when n becomes infinite in Equation (18). In order to meet this condition, it needs

$$\prod_{i=1}^{n} |(1 + \lambda\eta_i^e)| < 1 \tag{19}$$

Besides, the following Equations (20)–(25) could be clearly deduced.

$$\mathbf{x} \cdot \mathbf{x} = \|\mathbf{x}\|^2 \tag{20}$$

$$\|\mathbf{f}\| = \sqrt{(\lambda\mathbf{x})(\lambda\mathbf{x})} = |\lambda|\|\mathbf{x}\| \tag{21}$$

$$\mathbf{f} \cdot \mathbf{f} = \|\mathbf{f}\|^2 = |\lambda|^2\|\mathbf{x}\|^2 \tag{22}$$

$$\mathbf{f} \cdot \mathbf{x} = \lambda\mathbf{x} \cdot \mathbf{x} = \lambda\|\mathbf{x}\|^2 \tag{23}$$

$$\|\mathbf{x}\| \cdot \|\mathbf{f}\| = \|\mathbf{x}\| \cdot \|\lambda\mathbf{x}\| = |\lambda|\|\mathbf{x}\|^2 \tag{24}$$

$$b_n - a_n = -e^{\frac{h\|\mathbf{f}_n\|}{\|\mathbf{x}_n\|}} = -e^{-h|\lambda|} \tag{25}$$

Hence, Equation (15) can be transformed into

$$\begin{aligned}\eta_n^e &= \frac{(a_n-1)\lambda\|\mathbf{x}_n\|^2 + b_n|\lambda|\|\mathbf{x}_n\|^2}{|\lambda|^2\|\mathbf{x}_n\|^2} \\ &= \frac{(a_n-1)\lambda + b_n|\lambda|}{|\lambda|^2}\end{aligned} \tag{26}$$

In practice, $\lambda \in \mathbb{R}$ is often restricted; then, when $\lambda < 0$, Equation (26) could be transformed into

$$\begin{aligned}\eta_n^e &= \frac{(a_n-1)\lambda - b_n\lambda}{(-\lambda)(-\lambda)} \\ &= \frac{b_n-(a_n-1)}{-\lambda} \\ &= \frac{1-e^{-h|\lambda|}}{-\lambda} \\ &= \frac{1-e^{h\lambda}}{-\lambda}\end{aligned} \tag{27}$$

Consequently, Equation (18) is deduced

$$\mathbf{x}_{n+1} = \mathbf{x}_0\prod_{i=1}^{n}(1+e^{h\lambda}-1) = \mathbf{x}_0(e^{h\lambda})^n \tag{28}$$

Thus, $e^{h\lambda} < 1$ for $\forall h > 0$, which implies that the interval of absolute stability contains $(-\infty, 0)$.

2.2. *Midpoint Group Preserving Scheme*

As the well-known second-order Runge–Kutta method, the implicit midpoint rule is a predictor–corrector numerical algorithm. The predicted value at $t_{n+\frac{1}{2}}$ is approximated by Euler's method with a half step [23]

$$x_{n+1} = x_n + h\mathbf{f}(t_{n+\frac{1}{2}}, \frac{\mathbf{x}_n+\mathbf{x}_{n+1}}{2}), \quad t_{n+\frac{1}{2}} = t_n + \frac{h}{2} \tag{29}$$

The improvement of the midpoint method is to as accurately as possible obtain some of the curvature that will occur in the solution before calculating $\mathbf{x}_{n+1}$. Integrating the notion of GPS (14), step size h is replaced with η_n^e. Thus, the midpoint group preserving scheme can be formulated naturally

$$\begin{cases}\mathbf{x}_{n+1} = \mathbf{x}_n + \eta_n^e\mathbf{f}_n \\ \mathbf{x}_{n+1} = \mathbf{x}_n + \eta_n^e\mathbf{f}(t_{n+\frac{1}{2}}, \frac{\mathbf{x}_n+\mathbf{x}_{n+1}}{2})\end{cases} \tag{30}$$

where η_n^e is the same as the definition of (15). In the following subsections, we will discuss its error estimator and stability analysis.

2.2.1. Error Analysis

Firstly, we consider the error associated with the proposed algorithm (30). Let the local truncation error be e_n; then,

$$\mathbf{x}(t_{n+1}) = \mathbf{x}(t_n) + \eta_n^e\mathbf{f}(t_{n+\frac{1}{2}}, \frac{\mathbf{x}(t_n)+\mathbf{x}(t_{n+1})}{2}) + e_n \tag{31}$$

We slightly modify the right-hand side and write

$$\mathbf{x}(t_{n+1}) = \mathbf{x}(t_n) + \eta_n^e\mathbf{f}(t_{n+\frac{1}{2}}, \mathbf{x}(t_{n+\frac{1}{2}})) + e_{n_1} + e_n \tag{32}$$

Then, by Lipschitz condition (2),

$$\begin{aligned}|e_{n_1}| &= \eta_n^e|\mathbf{f}(t_{n+\frac{1}{2}}, \frac{\mathbf{x}(t_n)+\mathbf{x}(t_{n+1})}{2}) - \mathbf{f}(t_{n+\frac{1}{2}}, \mathbf{x}(t_{n+\frac{1}{2}}))| \\ &\leq \frac{1}{2}\eta_n^e L|\mathbf{x}(t_n) + \mathbf{x}(t_{n+1}) - 2\mathbf{x}(t_{n+\frac{1}{2}})| \end{aligned} \tag{33}$$

and expanding $\mathbf{x}(t_{n+1})$ and $\mathbf{x}(t_{n+\frac{1}{2}})$ in Taylor series about point t_n with $\mathbf{x} = \mathbf{x}(t_n)$, we have

$$|e_{n_1}| \leq \frac{1}{2}\eta_n^e L|\mathbf{x} + (\mathbf{x} + h\mathbf{x}') - 2(\mathbf{x} + \frac{1}{2}h\mathbf{x}') + \mathcal{O}(h^2)| \tag{34}$$

In practice, since $|\eta_n^e - h| = \mathcal{O}(h)$, the right side of (34) is $\mathcal{O}(h^3)$. Now, from (32), since $\mathbf{f}(t, x(t)) = x'(t)$, we obtain

$$\begin{aligned}e_{n_1} + e_n &= \mathbf{x}(t_{n+1}) - \mathbf{x}(t_n) - \eta_n^e\mathbf{x}'(t_{n+\frac{1}{2}}) \\ &= h\mathbf{x}' + \frac{1}{2}h^2\mathbf{x}'' - \eta_n^e(\mathbf{x}' + \frac{1}{2}h\mathbf{x}'') + \mathcal{O}(h^3) \\ &= \mathcal{O}(h^3)\end{aligned} \tag{35}$$

Hence,

$$|e_n| \leq |e_{n_1}| + \mathcal{O}(h^3) \leq ch^3 \tag{36}$$

Thus, local truncation error e_n is bounded.

Next, let the global truncation error $E_n = \mathbf{x}(t_n) - \mathbf{x}_n$. The recurrence relation for En is obtained by subtracting (30) from (31)

$$E_{n+1} = E_n + \eta_n^e(\mathbf{f}(t_{n+\frac{1}{2}}, \frac{\mathbf{x}(t_n)+\mathbf{x}(t_{n+1})}{2}) - \mathbf{f}(t_{n+\frac{1}{2}}, \frac{\mathbf{x}_n + \mathbf{x}_{n+1}}{2})) + e_n \tag{37}$$

The Lipschitz condition implies

$$\begin{aligned}|E_{n+1}| &\leq |E_n| + \frac{1}{2}\eta_n^e L|\frac{\mathbf{x}(t_n)+\mathbf{x}(t_{n+1})}{2} - \frac{\mathbf{x}_n + \mathbf{x}_{n+1}}{2}| + |e_n| \\ &\leq |E_n| + \frac{1}{2}\eta_n^e L(|E_n| + |E_{n+1}|) + ch^3\end{aligned} \tag{38}$$

Thereby,

$$|E_{n+1}| \leq \frac{1+\frac{1}{2}\eta_n^e L}{1-\frac{1}{2}\eta_n^e L}|E_n|) + ch^3 \leq \frac{1+\frac{1}{2}(h + c_n\mathcal{O}(h^2))L}{1-\frac{1}{2}(h - c_n\mathcal{O}(h^2))L} + ch^3 \quad c_n > 0 \tag{39}$$

Let $R = max\{c_0\mathcal{O}(h^2), c_1\mathcal{O}(h^2), \cdots, c_n\mathcal{O}(h^2)\}$ and $r = min\{c_0\mathcal{O}(h^2), c_1\mathcal{O}(h^2), \cdots, c_n\mathcal{O}(h^2)\}$; then,

$$|E_{n+1}| \leq \frac{1+\frac{1}{2}(h+R)L}{1-\frac{1}{2}(h-r)L} + ch^3 \tag{40}$$

Now, the relations of the form

$$|E_{n+1}| \leq A|E_n| + B, \quad n \geq 0, \quad E_0 = 0 \tag{41}$$

provide the estimate

$$|E_{n+1}| \leq B(1 + A + A^2 + \cdots + A^n) = B\frac{A^{n+1}-1}{A-1} \tag{42}$$

Since $\frac{1}{2} \leq 1 - \frac{1}{2}(h-r)L \leq 1$ for small h, we have

$$A - 1 = \frac{(h + \frac{1}{2}(R+r))L}{1 - \frac{1}{2}(h-r)L} \geq (h + \frac{1}{2}(R+r))L \tag{43}$$

$$A^{n+1} = (1 + \frac{(h + \frac{1}{2}(R+r))L}{1 - \frac{1}{2}(h-r)L})^{n+1} \leq (1 + 2(h + \frac{1}{2}(R+r))L)^{n+1} \leq e^{2(h+\frac{1}{2}(R+r))L(n+1)} \tag{44}$$

whence

$$|E_{n+1}| \leq \frac{1}{(h + \frac{1}{2}(R+r))L} e^{2(h+\frac{1}{2}(R+r))L(n+1)} ch^3 \leq c_1 h^3 \tag{45}$$

with a constant c_1 that only depends on L and t, i.e., we have uniform convergence in any finite interval [0, t].

2.2.2. Stability Analysis

Just as previously discussed in Section 2, according to Equation (17), we obtain

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \eta_n^e \lambda (\mathbf{x}_n + \frac{\eta_n^e}{2} \lambda \mathbf{x}_n) \tag{46}$$

$$= \mathbf{x}_n (1 + \eta_n^e \lambda + \frac{1}{2} (\eta_n^e \lambda)^2) \tag{47}$$

Similarly, if $\lambda \in \mathbb{R}$ is adopted and $\lambda < 0$; then, substituting Equation (27) into Equation (30)

$$\mathbf{x}_{n+1} = \mathbf{x}_n (\frac{e^{h\lambda} + 1}{2}) \tag{48}$$

For any $h > 0$, $0 < \frac{e^{h\lambda}+1}{2} < 1$; thus, it also has intervals of absolute stability containing $(-\infty, 0)$.

2.3. Scalability

As a matter of fact, along this line of thought, more classes of midpoint-like group preserving schemes can be extended.

$$\begin{cases} \mathbf{x}_{n+1} = \mathbf{x}_n + \eta^e_{n,\frac{1}{2}} \mathbf{f}_n \\ \bar{\mathbf{x}}_{n+1} = \mathbf{x}_n + \eta^e_{n,\frac{1}{2}} \mathbf{f}_n(\mathbf{x}_{n+1}) \\ \mathbf{x}_{n+1} = \mathbf{x}_n + \eta_n^e \mathbf{f}(t_{n+\frac{1}{2}}, \frac{\mathbf{x}_n + \bar{\mathbf{x}}_{n+1}}{2}) \end{cases} \tag{49}$$

Comparing Equations (30) and (49), it is intuitively clear that the prediction function has been further optimized, while the correction function remains in the same format. This optimization is worthwhile as evidenced by the following examples 3 and 4. This scheme is shorthanded as OMGPS hereafter.

3. Experiments

In this section, the experiments will be conducted in two types of examples: one for regular continuous dynamical systems (Examples 1 and 2) and the other for two special types of systems, Kolmogorov systems (Example 3) and Hamiltonian systems (Example 4). The proposed method and other second-order methods are adopted and compared. All tests are performed in MATLAB R2020a.

3.1. Example 1

The following nonlinear differential equation is use to describe population growth in a nonlinear hybrid system [11]

$$\begin{cases} \dot{x_1} = (2000 - x_2 - 5x_1)x_1 \\ \dot{x_2} = (4x_1 - 2600 - 4x_2)x_2 \end{cases} \tag{50}$$

where initial value $\mathbf{x}_0 = \{900, 150\}$ and time $t = 0.008$. Its true solutions are simulated using explicit Euler scheme with a step size of $h = 0.00001$, and other numerical methods have the same step size $h = 0.001$. Figure 2 shows their computation results. It is evident that GPS deviates from the real curve by a large margin. The simulation outcomes of midpoint, Heun, enhanced-GPS, and MGPS are quite close to each other. By comparison clearly, OMGPS is on top of them. This is also in accordance with the optimization point of Section 2.3.

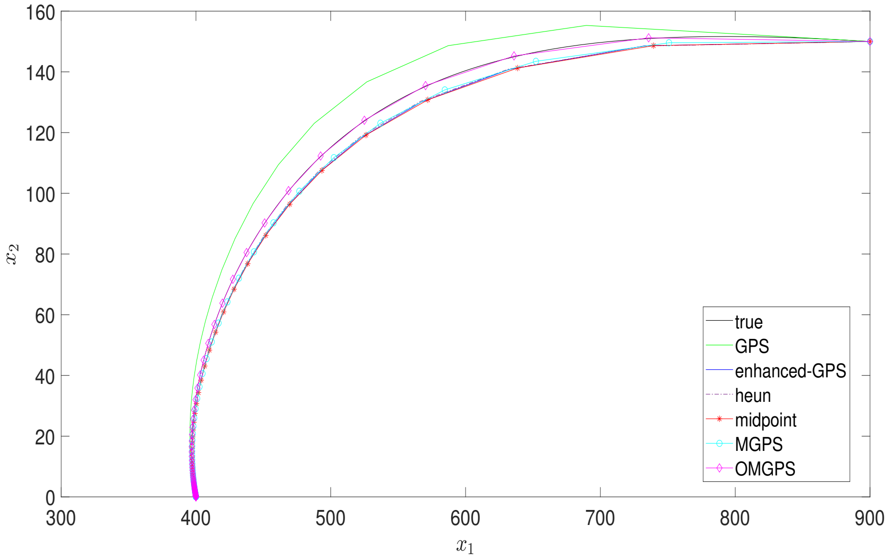

Figure 2. Phase plots for Example 1 with $\mathbf{x}_0 = \{900, 150\}$ and $h = 0.0001$.

3.2. Example 2

The second example is taken from the literature [26], which is a two-dimensional continuous system. Starting from initial set $\mathbf{x}_0 = \{1, 0\}$, the system evolves from time 0 to 25.

$$\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = -x_1 + \dfrac{1}{3}x_1^3 - x_2 \end{cases} \tag{51}$$

Similarly, applying explicit Euler method with $h = 0.0001$ simulates the true curve. $h = 0.5$ is adopted for the remaining methods. The simulation results are exhibited in Figure 3. Apparently, The conclusion reached is identical to the first example.

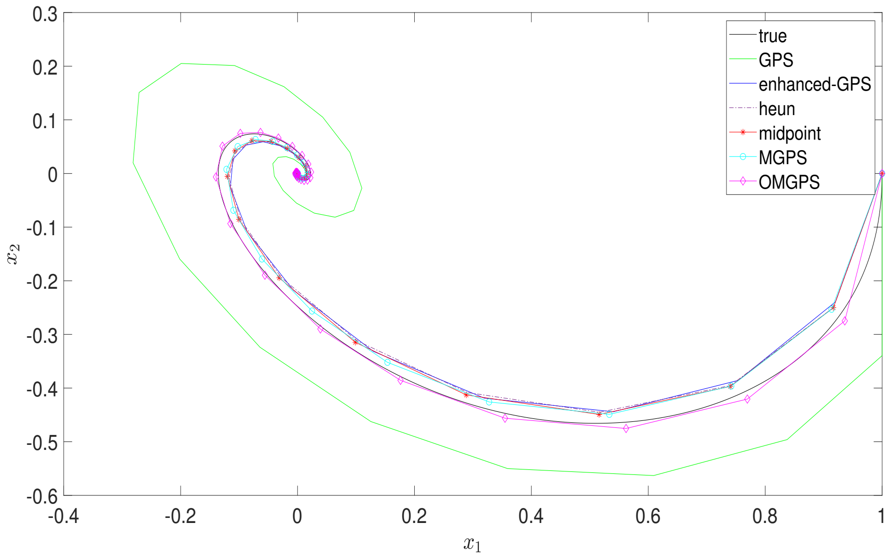

Figure 3. Phase plots for Example 2 with $\mathbf{x}_0 = \{1, 0\}$ and $h = 0.5$.

3.3. Example 3

The Lotka–Volterra system is an important mathematical model for depicting predator–prey interactions in the dynamics of biological systems. The Lotka–Volterra equations can be written simply as follows:

$$\begin{cases} \dot{x_1} = ax_1 - bx_1x_2 \\ \dot{x_2} = -cx_2 + dx_1x_2 \end{cases} \tag{52}$$

It was mentioned earlier that [12] proposes a nonstandard finite-difference scheme (abbreviated here as NSFD) for this system, and to facilitate comparison, the same parameters as in [12] are employed here for the test, namely, $a = b = c = d = 1$. As can be seen in Figure 4, the numerical integration results of MGPS form a closed curve, which is consistent with the results of NSFD and is very close to the true value, which is simulated using the Euler method with a step size of 0.0001. The Figure 5 exhibits more clearly the error accuracy of several methods, and the superiority of MGPS is noticeable.

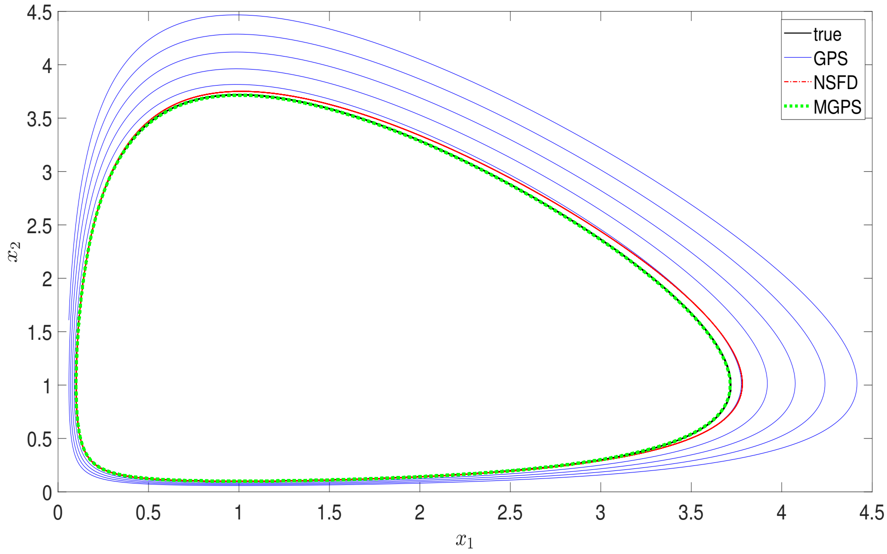

Figure 4. Phase plots for Example 3 with $\mathbf{x}_0 = \{0.1, 1\}$ and $h = 0.01$.

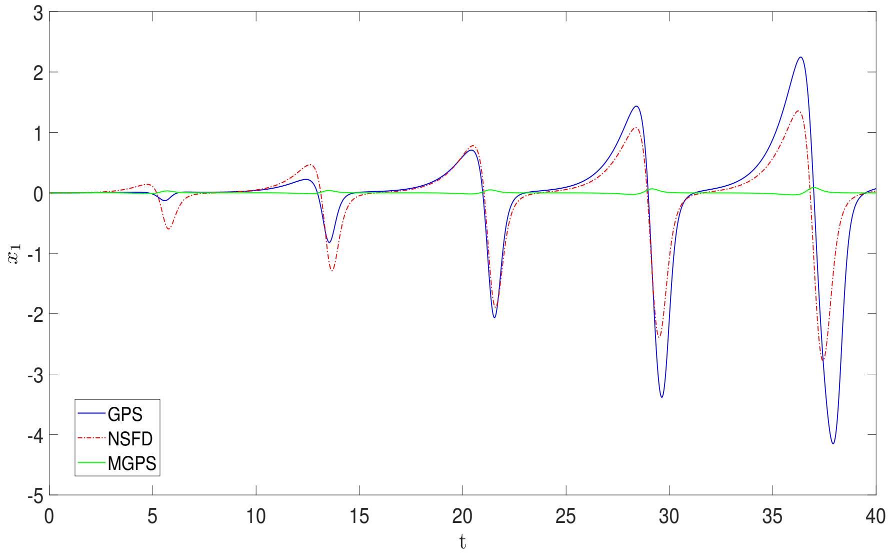

Figure 5. Error plots about x_1 of Example 3 with $\mathbf{x}_0 = \{0.1, 1\}$ and $h = 0.01$.

3.4. Example 4

The last example is oriented towards Hamiltonian systems and comes from the literature [19], which demonstrates the GR family of methods(GR, MOD_GR, GR_LEX, GR_SLEX). From this, we chose GR_SLEX for comparison. The Hamiltonian system is the simple pendulum $H = \frac{p^2}{2} - cosx$. The corresponding ODE is

$$\begin{cases} \dot{x} = H_p = p \\ \dot{p} = -H_x = -sinx \end{cases} \tag{53}$$

Here, two sets of experiments were conducted adopting the same parameters as those listed above in the literature—that is, $h = 0.25$ and $h = 0.01$. The rest of parameters are $x_0 = 0, p_0 = 1.8, t = 20$. In Figure 6, GPS performs very poorly and yields erroneous results. In Figure 7, however, the true value is better simulated, but the accuracy in comparison with the other two methods is inferior. This illustrates the fact that GPS is less stable in the Hamilton system. When dealing with this type of problem with GPS, a relatively small h is recommended. In both sets of experiments, MGPS and GR_SLEX are reliably capable of modeling the true values. Figures 8 and 9 express that GR_SLEX has better accuracy. With the step size becoming smaller, the precision gap of MGPS and GR_SLEX is narrowing, and it can be seen from Figure 9 that MGPS nearly has the same level of error accuracy as GR_SLEX. Regarding the running efficiency, we executed two sets of trials—one with 20,000 iterations and the other with 200,000 iterations, to minimize the effect of other factors—and averaged the running time after three runs for each group. The outcome of the experimental data is placed in Appendix A and shows that when the number of iterations is not large, the running times are comparable, and vice versa, GR_SLEX is superior. Although *MGPS* is not computed serially, this strength cannot be fully exploited at low dimensions.

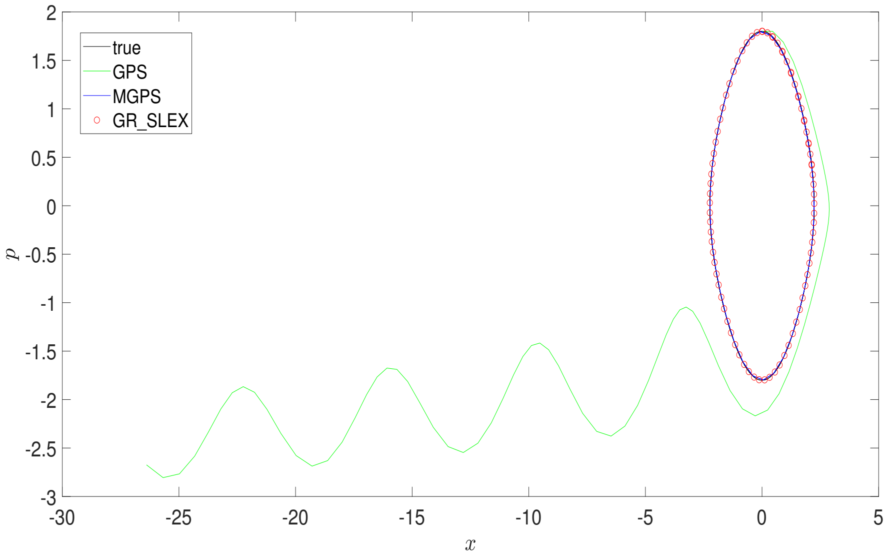

Figure 6. Phase plots for Example 4 with $x_0 = 0$, $p_0 = 1.8$, and $h = 0.25$.

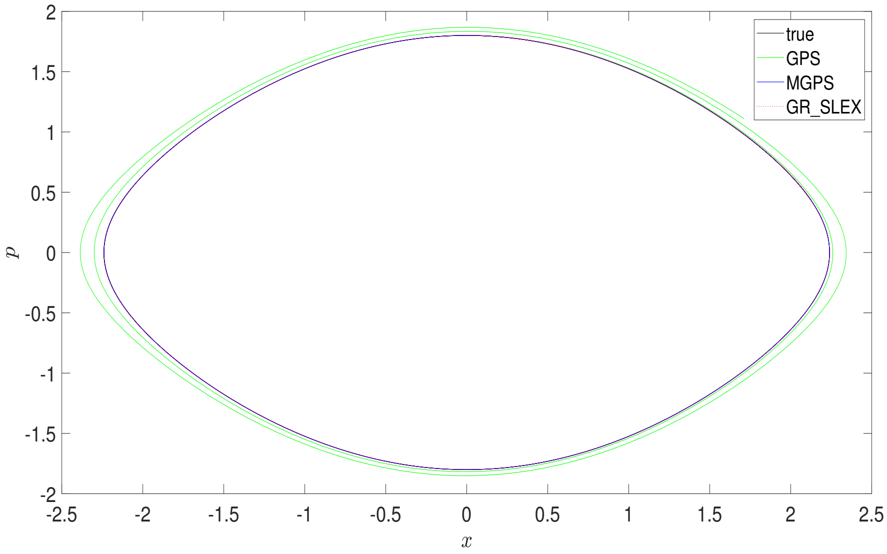

Figure 7. Phase plots for Example 4 with $x_0 = 0$, $p_0 = 1.8$, and $h = 0.01$.

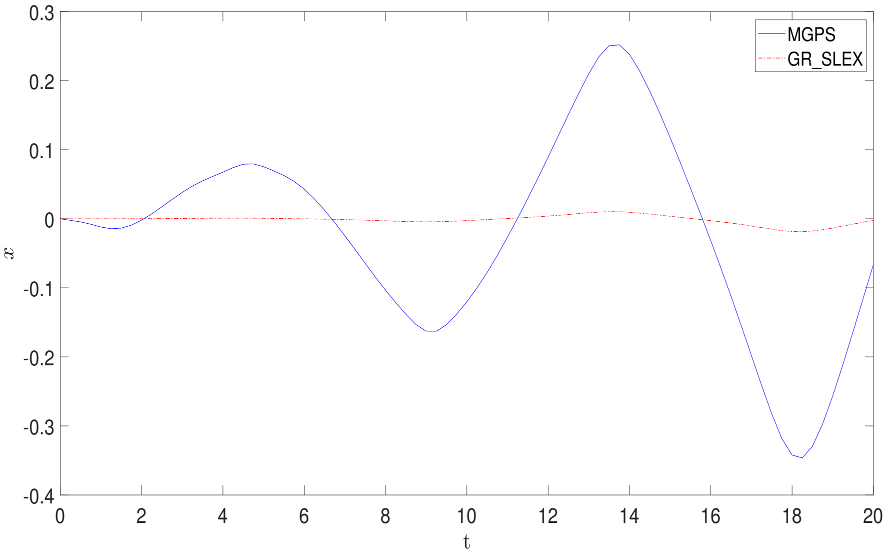

Figure 8. Error plots about x of Example 4 with $x_0 = 0$, $p_0 = 1.8$, and $h = 0.25$.

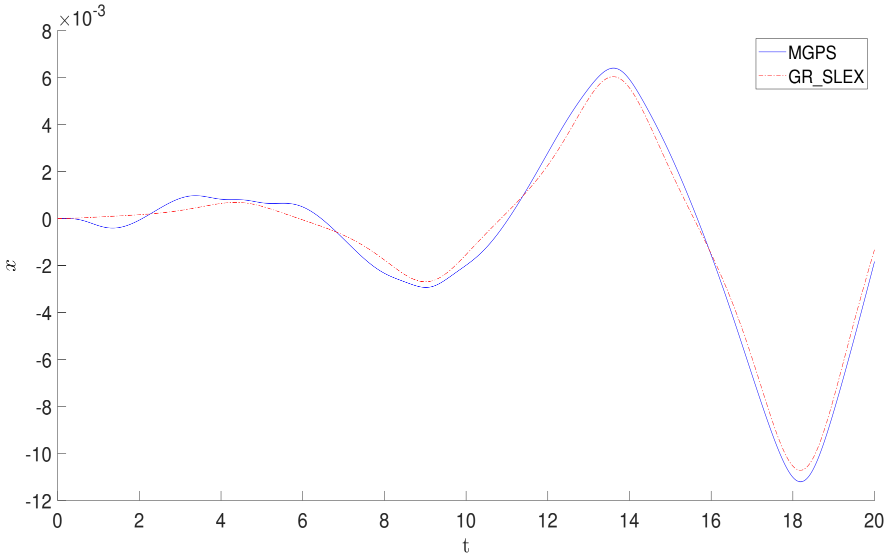

Figure 9. Error plots about x of Example 4 with $\mathbf{x}_0 = 0$, $p_0 = 1.8$, and $h = 0.01$.

4. Conclusions

Computing flow pipes is a highly significant problem in the verification of hybrid systems. This paper constructs an easy-to-implement and extensible midpoint-series group preserving scheme from a numerical analysis point of view, making use of the classical midpoint method and the symmetry of the group structure. Yet, the verification of hybrid systems faces several more problems than this one. How to effectively detect the termination of continuous evolution is another very important issue. To the best of our knowledge, there is no viable numerical approach. How to better solve this problem and the solution of high-dimensional nonlinear dynamic equations will be the focus of our future work.

Author Contributions: Conceptualization, Z.X. and J.W.; methodology, Z.X.; software, Z.X.; validation, Z.X. and J.W.; formal analysis, Z.X.; investigation, Z.X.; resources, Z.X.; data curation, Z.X.; writing—original draft preparation, Z.X.; writing—review and editing, J.W.; visualization, Z.X.; supervision, J.W.; project administration, J.W.; funding acquisition, J.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grant No. 61772006, the Science and Technology Major Project of Guangxi under Grant No. AA17204096, the Key Research and Development Project of Guangxi under Grant No. AB17129012, and the Special Fund for Bagui Scholars of Guangxi.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Running time data for MGPS and GR_SLEX. The CPU is an Intel(R) Core(TM) i7-10750H CPU @ 2.60 GHz 2.59 GHz with 32 G of RAM. $t = 20$ h.

Number of Iterations (and Step Size)	Scheme Name	Running Time (s)	Average Running Time (s)
20,000 (h = 0.0001)	MGPS	66.4450	67.3003
		68.4420	
		67.0140	
	GR_SLEX	65.6050	67.844
		71.1950	
		66.7320	
200,000 (h = 0.00001)	MGPS	8.7665×10^3	8.5724×10^3
		8.4467×10^3	
		8.5039×10^3	
	GR_SLEX	7.9818×10^3	7.9215×10^3
		7.8997×10^3	
		7.8831×10^3	

References

1. Chutinan, A. Hybrid System Verification Using Discrete Model Approximations. Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 1999.
2. Asarin, E.; Bournez, O.; Dang, T.; Maler, O.; Pnueli, A. Effective synthesis of switching controllers for linear systems. *Proc. IEEE* **2000**, *88*, 1011–1025. [CrossRef]
3. Girard, A. Reachability of uncertain linear systems using zonotopes. In *Hybrid Systems: Computation and Control*; Morari, M., Thiele, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2005; Volume 3414, pp. 291–305.
4. Kurzhanskiy, A.A.; Varaiya, P. Ellipsoidal techniques for reachability analysis of discrete-time linear systems. *IEEE Trans. Autom. Control* **2007**, *52*, 26–38. [CrossRef]
5. Le Guernic, C.; Girard, A. Reachability analysis of hybrid systems using support functions. In *Computer Aided Verification*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 540–554.
6. Henzinger, T.A.; Ho, P.H.; Wong-Toi, H. HYTECH: A model checker for hybrid systems. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 1998; Volume 1254, pp. 460–463.
7. Asarin, E.; Bournez, O.; Dang, T.; Maler, O. Approximate Reachability Analysis of Piecewise Linear Dynamical Systems. In *Proceedings of the HSCC 00: Hybrid Systems—Computation and Control*; Springer: Berlin/Heidelberg, Germany, 2000; Volume 1790, pp. 20–31.
8. Chutinan, A.; Krogh, B.H. Verification of Polyhedral-Invariant Hybrid Automata Using Polygonal Flow Pipe Approximations. In *International Workshop on Hybrid Systems: Computation and Control*; Springer: Berlin/Heidelberg, Germany, 2000; Volume 1569, pp. 76–90.
9. Frehse, G. PHAVer: Algorithmic verification of hybrid systems past HyTech. *Int. J. Softw. Tools Technol. Transf.* **2008**, *10*, 263–279. [CrossRef]
10. Frehse, G.; Le Guernic, C.; Donzé, A.; Cotton, S.; Ray, R.; Lebeltel, O.; Ripado, R.; Girard, A.; Dang, T.; Maler, O. Spaceex: Scalable verification of hybrid systems. In *Computer Aided Verification*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 379–395.
11. Henzinger, T.A.; Ho, P.H.; Wong-Toi, H. Algorithmic analysis of nonlinear hybrid systems. *IEEE Trans. Autom. Control* **1998**, *43*, 540–554. [CrossRef]
12. Mickens, R.E. A nonstandard finite-difference scheme for the Lotka-Volterra system. *Appl. Numer. Math.* **2003**, *45*, 309–314. [CrossRef]

13. De Vogelaere, R. *Methods of Integration which Preserve the Contact Transformation Property of the Hamilton Equations*; Technical Report, Unpublished; University of Notre Dame. Dept. of Mathematics: Notre Dame, IN, USA, 1956.
14. Ruth, R.D. A canonical integration technique. *IEEE Trans. Nucl. Sci.* **1983**, *30*, 2669–2671. [CrossRef]
15. Feng, K. On difference schemes and symplectic geometry. In Proceedings of the 5th International Symposium on Differential Geometry and Differential Equations, Beijing, China, August 1984; pp. 42–58.
16. Feng, K. Difference schemes for Hamiltonian formalism and symplectic geometry. *J. Comput. Math.* **1986**, *4*, 279–289.
17. McLachlan, R.I.; Quispel, G.R.W.; Robidoux, N. Geometric integration using discrete gradients. *Philos. Trans. R. Soc. Lond. Ser. Math. Phys. Eng. Sci.* **1999**, *357*, 1021–1045. [CrossRef]
18. Cieśliński, J.L.; Ratkiewicz, B. Long-time behaviour of discretizations of the simple pendulum equation. *J. Phys. Math. Theor*. **2009**, *42*, 105204. [CrossRef]
19. Cieśliński, J.L.; Ratkiewicz, B. Improving the accuracy of the discrete gradient method in the one-dimensional case. *Phys. Rev. E* **2010**, *81*, 016704. [CrossRef] [PubMed]
20. Cieśliński, J.L.; Ratkiewicz, B. Energy-preserving numerical schemes of high accuracy for one-dimensional Hamiltonian systems. *J. Phys. A Math. Theor.* **2011**, *44*, 155206. [CrossRef]
21. d'Aquino, M.; Serpico, C.; Miano, G. Geometrical integration of Landau–Lifshitz–Gilbert equation based on the mid-point rule. *J. Comput. Phys*. **2005**, *209*, 730–753. [CrossRef]
22. Shepherd, D. Numerical Methods for Dynamical Micromagnetics. Ph.D. Thesis, The University of Manchester, Manchester, UK, 2015.
23. Hairer, E.; Norsett, S.P.; Wanner, G. *Solving Ordinary Differential Equations I: Nonstiff Problems*; Springer: Berlin/Heidelberg, Germany, 1993.
24. Shepherd, D.; Miles, J.; Heil, M.; Mihajlović, M. An Adaptive Step Implicit Midpoint Rule for the Time Integration of Newton's Linearisations of Non-Linear Problems with Applications in Micromagnetics. *J. Sci. Comput*. **2019**, *80*, 1058–1082. [CrossRef]
25. Liu, C.S. Cone of non-linear dynamical system and group preserving schemes. *Int. J. Non-Linear Mech.* **2001**, *36*, 1047–1068. [CrossRef]
26. Zhang, H.; Wu, J.; Lu, J.; Tang, J. Safety verification of finite real-time nonlinear hybrid systems using enhanced group preserving scheme. *Clust. Comput.* **2016**, *19*, 2189–2199. [CrossRef]
27. Chen, B.; Solis, F. Discretizations of nonlinear differential equations using explicit finite order methods. *J. Comput. Appl. Math.* **1998**, *90*, 171–183. [CrossRef]