mathematics

MDPI

Article

Numerical Methods That Preserve a Lyapunov Function for Ordinary Differential Equations

Yadira Hernández-Solano † **and Miguel Atencia** *,†

Departamento de Matemática Aplicada, Universidad de Málaga, 29071 Málaga, Spain
* Correspondence: matencia@uma.es
† These authors contributed equally to this work.

Abstract: The paper studies numerical methods that preserve a Lyapunov function of a dynamical system, i.e., numerical approximations whose energy decreases, just like in the original differential equation. With this aim, a discrete gradient method is implemented for the numerical integration of a system of ordinary differential equations. In principle, this procedure yields first-order methods, but the analysis paves the way for the design of higher-order methods. As a case in point, the proposed method is applied to the Duffing equation without external forcing, considering that, in this case, preserving the Lyapunov function is more important than the accuracy of particular trajectories. Results are validated by means of numerical experiments, where the discrete gradient method is compared to standard Runge–Kutta methods. As predicted by the theory, discrete gradient methods preserve the Lyapunov function, whereas conventional methods fail to do so, since either periodic solutions appear or the energy does not decrease. Moreover, the discrete gradient method outperforms conventional schemes when these do preserve the Lyapunov function, in terms of computational cost; thus, the proposed method is promising.

Keywords: geometric numerical integration; dynamical systems; Lyapunov function; stability; numerical methods; ordinary differential equations; discrete gradient

MSC: 65P40; 37M15

check for updates

Citation: Hernández-Solano, Y.; Atencia, M. Numerical Methods That Preserve a Lyapunov Function for Ordinary Differential Equations. *Mathematics* **2023**, *11*, 71. https://doi.org/10.3390/math11010071

Academic Editors: Hongyu Liu and Ioannis K. Argyros

Received: 8 October 2022
Revised: 2 November 2022
Accepted: 20 December 2022
Published: 25 December 2022

1. Introduction

The main aim of this paper is the study of numerical methods that preserve a Lyapunov function of a gradient dynamical system. The solutions, or integral curves, of a gradient system, follow trajectories that are tangent to a scalar function of the states, which is usually known as the Lyapunov function of the system. The flow of a gradient system has a rather simple qualitative behavior, e.g., all isolated minima of the Lyapunov function are asymptotically stable equilibria of the system. Dynamical analysis by Lyapunov's method is a well-established discipline, and plenty of references abound on the topic [1–4]. The Lyapunov function has the remarkable property that it is decreasing along trajectories of the dynamical system. Gradient systems are pervasive, both as models of physical systems and as representations of mathematical algorithms. For example, an ideal pendulum is a conservative system, namely the energy is a constant magnitude, but every actual mechanical system dissipates energy due to friction, until all potential and kinetic energies vanish, thus, energy acts as the Lyapunov function of the system. Remarkably, many mathematical algorithms are formulated in continuous time whose operation is based on the existence of the Lyapunov function, for example, in the fields of optimization, estimation, and control [5–9].

Numerical methods for the integration of Ordinary Differential Equations (ODEs) constitute a well-established field [10], and methods that provide rather accurate solutions for a wide variety of problems have long been known. However, no matter how small the approximation error of a numerical method is, it can lead to a solution that does not portray the qualitative features of the continuous model, when the integration extends through long time periods. A classic example is the Kepler problem [11], whose approximate solutions by conventional numerical methods do not respect the elliptical orbits describing the motion of the planets, as established by Kepler's first law. The inability of basic numerical methods to reflect crucial qualitative properties of dynamical systems led to the development of a new approach, namely *Geometric Numerical Integration* [12], which is an active line of research that links the methodology of dynamical systems analysis to the design of numerical methods [13] that preserve the qualitative properties of the continuous system. In this regard, the main objective is to consider the qualitative characteristics of the trajectories of the dynamical system, for example, energy decreasing, stability, and conservation of the Hamiltonian, among others. The task would then be the design of numerical methods so that the discrete trajectories of the method have the same properties as the exact solutions.

Within the field of Geometric Numerical Integration, there exists a substantial number of results concerning the study of systems with *first integrals* [14–16]. A differential geometric approach to this topic implies the discretization of Hamiltonian systems [17], since in a system with a conserved quantity, a symplectic form can always be defined. Among the number of methods defined for this class of systems, symplectic and projection methods have been well studied [12]. However, when it comes to the conservation of the Lyapunov function of a gradient dynamical system, the choice is limited—to the best of our knowledge—to three categories: discrete gradient methods [18], projection methods [19], and particular instances of Runge–Kutta methods [20]. The inattention to stability issues is striking since the dynamic analysis of ODEs is far from new in the field of numerical analysis. Indeed, the concept of A-stability [21] amounts to the preservation of the stability of the solution of linear scalar equations as test systems. In this regard, the conservation of the Lyapunov function can be viewed as a generalization of the concept of A-stability in a nonlinear context.

Projection methods [19] inherit the design of analogous methods for Hamiltonian systems, which are based upon projecting the approximate solution onto the manifold that the trajectories of the exact solutions lie in. Although the formulation of these methods is explicit in principle, they require solving the nonlinear equation that defines the projection at each time step. Moreover, from a conceptual point of view, we think that there is arguably something unsatisfactory in this differential geometric approach, since the Lyapunov function, unlike the Hamiltonian, does not define a manifold, i.e., the distribution of admissible trajectory directions is not integrable. For its part, the application of Runge–Kutta methods to gradient systems [20] led to proving that some Radau implicit methods, originally proposed for stiff and Hamiltonian systems, are also able to preserve the Lyapunov function under certain conditions and restrictions on the step size. Both projection methods and implicit Runge–Kutta integrators rely on non-constructive theorems, so they cannot guarantee the preservation of the Lyapunov function unless some ad-hoc adjustment of the step size is performed. In summary, although these methods are promising, their implementation is complicated and can lead to a substantial computational cost, so they are not suitable for all situations. It must be emphasized that, rather than advocating against other techniques, our results encourage further attention to discrete gradient methods, at least for particular applications. Nonetheless, some considerations on future lines of research for comparative assessments of all these methods are made in the conclusions.

Discrete gradient methods yield integrators for ODEs, based upon the fact that the equation of a gradient system can be written in linear-gradient form, i.e., as the product of a definite-negative matrix by the gradient of a Lyapunov function. Then, discrete gradient methods can be stated with a simple rationale: define an approximation of the definite-negative matrix and a discrete gradient, which has similar properties to that of the gradient of the Lyapunov function. By construction, these methods lead to an implicitly defined map that, when considered as a discrete dynamical system, preserves the Lyapunov function of the continuous system. Discrete gradient methods for systems with Lyapunov functions have been described mostly as an aside of methods for Hamiltonian systems [14,16,18,22].

Thus the development of discrete gradient methods for dissipative, rather than conservative, systems is limited, and examples of systematic application to real systems are hardly found in the literature, as far as we know. In previous work, we explored the application of discrete gradient methods to a particular system, namely Hopfield neural networks [23,24], which are computational methods used for optimization.

The main novelty of this paper is the contribution towards establishing a general systematic methodology for the development of discrete gradient methods, specifically tailored for systems with a Lyapunov function. After a review of the background about discrete gradient methods in Section 2, the contribution of this paper begins in Section 3, where we describe the methodology of implementation of discrete gradient methods, analyzing the order of the obtained method and illustrating its main properties by means of simple examples. Then, in Section 4, we present some systematic numerical experiments showing the performance of the proposed technique, and comparing its performance to standard Runge–Kutta methods. As a result, some favorable properties of the obtained method are brought to light. Finally, some conclusions and lines for further research are stated in Section 5.

2. Numerical Methods That Preserve the Lyapunov Function

In this section, we define and discuss key aspects of discrete gradient methods, after establishing the definitions that will be used in the paper.

2.1. Gradient Systems

First of all, we establish the notation for the dynamical system that must be dealt with, which is a finite-dimensional initial value problem (IVP), i.e., a system of ODEs with initial values:

$$\frac{d\vec{y}}{dt} = \vec{f}(\vec{y}) = \begin{pmatrix} f_1(\vec{y}) \\ \vdots \\ f_n(\vec{y}) \end{pmatrix}, \qquad \vec{y}(t_0) = \vec{y}_0 \in \mathbb{R}^n \tag{1}$$

Since we do not pursue existence and uniqueness issues; we take for granted all needed smoothness assumptions. The systems of interest are those that possess—at least—one asymptotically stable equilibrium (see e.g., [1,25] for definitions of stability concepts). An equilibrium or fixed point $\vec{y}^*$ fulfills $\vec{f}(\vec{y}^*) = \vec{0}$; thus, a trajectory that starts at $\vec{y}_0 = \vec{y}^*$ is the trivial trajectory $\vec{y}(t) = \vec{y}^*$. The statement that $\vec{y}^*$ is asymptotically stable amounts to saying that all trajectories $\vec{y}(t)$ that start in a certain neighbourhood $\mathcal{B}$ with $\vec{y}^* \in \mathcal{B}$ converge towards $\vec{y}^*$, i.e., $\lim_{t\to\infty} \vec{y}(t) = \vec{y}^*$ if $\vec{y}_0 \in \mathcal{B}$.

One of the aims of qualitative analysis of ODEs is to prove that equilibrium is stable without computing the explicit solution, which can be accomplished by finding a suitable Lyapunov function V:

Definition 1. *Given the system in Equation (1), the function $V \in C^1(\mathbb{R}^n, \mathbb{R})$ is a Lyapunov function for the equilibrium $\vec{y}^*$ if the following conditions hold in a neighbourhood $\mathcal{B}$ such that $\vec{y}^* \in \mathcal{B}$:*

(a) $V(\vec{y}^*) = 0$ *and* $V(\vec{y}) > 0$ *if* $\vec{y} \neq \vec{y}^*$.

(b) $\frac{d}{dt}V(\vec{y}(t)) < 0$ *for all* $\vec{y} \in \mathcal{B} - \{\vec{y}^*\}$.

Condition (a) remains trivially equivalent if the minimum at $\vec{y}^*$ is achieved at any value $V(\vec{y}^*)$ other than zero, by considering the Lyapunov function $W = V - V(\vec{y}^*)$. Since this can be done without loss of generality, we no longer remark on this detail in the examples. Moreover, by the chain rule, condition (b) is equivalent to stating the following relation of the gradient and the ODE: $\frac{dV}{dt} = \nabla V(\vec{y}) \cdot \vec{f}(\vec{y}) \leq 0$ with V bounded below.

Note that it is obvious that $\frac{dV}{dt} = 0$ at an equilibrium $\vec{y}^*$ since $\vec{f}(\vec{y}^*) = \vec{0}$. The existence of a Lyapunov function characterizes the stability of an equilibrium [25]:

Theorem 1. *Let $\vec{y}^*$ be an equilibrium point of the system in Equation (1) and V a Lyapunov function in a neighborhood $\mathcal{B}$ of $\vec{y}^*$. Then, $\vec{y}^*$ is asymptotically stable.*

A Lyapunov function is often called the *energy* of the system; by analogy with dissipative physical systems where the energy decreases, thus, it may be used as a Lyapunov function. Rigorously speaking, the definition of the Lyapunov function does not require the inequality in condition (b) of Definition 1 to be strict, and, when the inequality is strict, we should specify that the system has *strict* Lyapunov function. In this paper, we always assume that the Lyapunov function is strict, so we do not make this distinction. Likewise, we loosely refer to a stable point, dropping the assumed precision that such stability is asymptotic. It is worth remarking that, although converse theorems guarantee the existence of a Lyapunov function when a stable equilibrium exists, there is no general method for finding the explicit expression of a Lyapunov function. In this paper, we assume that a Lyapunov function is explicitly known.

The main aim of this paper is to find numerical methods that preserve the Lyapunov function of a system given by Equation (1). Formally, we construct a discrete dynamical system defined by a time-stepping formula $\vec{z} = \varphi_h(\vec{y})$ such that $\vec{z}$ is a suitable approximation of $\vec{y}(t+h)$ if $\vec{y}$ is an approximation of $\vec{y}(t)$. The required preservation of the Lyapunov function V is subsumed by the condition $V(\vec{z}) < V(\vec{y})$ as long as $\vec{z} \neq \vec{y}$, which is the discrete counterpart of condition (b) in Definition 1: both inequalities express that the Lyapunov function decreases through time, either in a discrete or a continuous setting. It will also be of interest to determine if the time-stepping scheme produces a sequence that converges to some stable equilibrium of the original system, thus, reproducing asymptotic stability.

2.2. Discrete Gradient Methods

The history of stability preserving methods can be traced back at least three decades, to the seminal paper [26] and, later, the book [13]. It is, thus, well known that numerical methods may destroy the structural properties of the original ODE, but note that there is a hierarchy of how subtle this effect can be. On the one hand, an equilibrium may cease to be a fixed point of the discrete method, or it may become an unstable equilibrium. These spurious solutions can easily be detected by a (more or less) straightforward analysis of the method, including linearization around the equilibrium. More importantly, there are established criteria to construct (local) stability-preserving numerical methods. A much more severe problem arises when the equilibrium is still locally asymptotically stable, but the numerical method fails to decrease the Lyapunov function or, in other words, the basins of attraction change. This alteration of geometrical properties has a global nature; hence, its study is notoriously difficult. Discrete gradient methods guarantee that the Lyapunov function of the ODE decreases along sequences of points obtained by the numerical method so that, at least from the point of view of energy minimization, the geometric structure is preserved.

The rationale behind discrete gradient methods is a rather simple idea, namely to replace the derivative of the Lyapunov function with a finite increment. This idea is useful for discretizing the system because the ODE and the Lyapunov function are related: every ODE, as in Equation (1) for which a Lyapunov function V is known, can be rewritten in *linear-gradient* form [18]:

$$\frac{d\vec{y}}{dt} = L(\vec{y})\,\nabla V(\vec{y}) \tag{2}$$

where L is a negative-definite matrix and both L and V are continuously differentiable. Incidentally, it is worth mentioning that this decomposition is not unique, and the different ways to write $L(\vec{y})$ can be regarded as different metric structures [27].

Remark 1. *Care must be taken when negative-definiteness is considered for non-symmetric matrices since, in this case, negative eigenvalues of L do not guarantee the intended relation $v^\top Lv < 0$ for any vector $v \neq 0$. Let us, thus, emphasize that, along the paper, a matrix L is negative-definite if its symmetric part $L + L^\top$ is.*

After rewriting the ODE in Equation (1) in linear-gradient form, a discrete gradient method results from the choice of discrete analogs to the matrix L and the gradient ∇V:

Definition 2. *Given a differentiable function $V \in C^1(\mathbb{R}^n, \mathbb{R})$, the function $\overline{\nabla} V \in C^1(\mathbb{R}^{2\,n}, \mathbb{R}^n)$ is a discrete gradient of V if it satisfies:*

$$\begin{aligned} \overline{\nabla} V(\vec{y}, \vec{z}) \cdot (\vec{z} - \vec{y}) &= V(\vec{z}) - V(\vec{y}) \\ \overline{\nabla} V(\vec{y}, \vec{y}) &= \nabla V(\vec{y}) \end{aligned} \tag{3}$$

In fact, the second condition is implied by the first in the differentiable case [22], but we include it anyway to emphasize consistency.

Definition 3. *A discrete gradient method is a time-advancing numerical scheme defined by*

$$\frac{\vec{z} - \vec{y}}{h} = \widetilde{L}(\vec{y}, \vec{z}, h)\, \overline{\nabla} V(\vec{y}, \vec{z}) \tag{4}$$

where $\overline{\nabla} V$ is a discrete gradient of V and the matrix $\widetilde{L}(\vec{y}, \vec{z}, h)$ of continuously differentiable functions is negative-definite and satisfies the consistency condition

$$\widetilde{L}(\vec{y}, \vec{y}, 0) = L(\vec{y}) \tag{5}$$

The aim of a discrete gradient method is to compute $\vec{z} \approx \vec{y}(t+h)$ from the previous step $\vec{y} \approx \vec{y}(t)$ so the sequence $\vec{y}(t)$ is an approximation of the solution of the system given by Equation (2). It is trivial to prove that a discrete gradient method is consistent, as a consequence of the requirements on $\widetilde{L}$ and $\overline{\nabla} V$. Remarkably, the methods given by Equation (4) are implicit, at least in principle, since the next step $\vec{z}$ appears on the right-hand side of the formula.

3. Construction and Analysis of Discrete Gradient Methods

Once the parameters $\widetilde{L}$ and $\overline{\nabla} V$ have been set, a particular instance of discrete gradient method results by substituting this parameter choice into Equation (4). This is a critical design process as there is a wide range of definitions that are compatible with the conditions given by Equations (3) and (5). In this section, we aim to contribute a first step toward the analysis of different parameter choices, with no claim to be exhaustive

3.1. Metric Matrix

Our usage of the name *metric matrix* to refer to the negative-definite matrix $\widetilde{L}$, derives from the fact that every system in linear-gradient form is indeed a gradient system for some metric [27]. The matrix $\widetilde{L}$ is, in this formulation, the expression in some coordinates of the corresponding metric tensor, with a changed sign.

The range of freedom for choosing the matrix $\widetilde{L}$ within Definition 3 is very wide. To begin with, the trivial choice $\widetilde{L}(\vec{y}, \vec{z}, h) = L(\vec{y})$ is possible, where the dependence on the next step $\vec{z}$ is neglected. A less radical simplification results when dismissing the step size h in the definition of $\widetilde{L}$. We adopt this latter assumption throughout this paper, so we often write $\widetilde{L}(\vec{y}, \vec{z})$ for this matrix.

Assuming analyticity of the functions included in the matrix $\widetilde{L}$, it can be expanded as a Taylor series:

$$\widetilde{L}(\vec{y}, \vec{z}) = L(\vec{y}) + M_1(\vec{y})\,(z - y) + (z - y)^\top M_2(\vec{y})\,(z - y) + \ldots \tag{6}$$

where the only requisite for matrices of functions M_n is that the resulting matrix $\widetilde{L}$ is negative-definite. Note that writing higher-order terms would require the cumbersome multi-index notation or the introduction of tensors.

We undertook the application of discrete gradient methods to scalar linear ODEs $\frac{dy}{dx} = -\lambda y$, mimicking the classical analysis of L-Stability. Such systems are obviously in linear-gradient form by setting constant $L(y) = -\lambda$ and the Lyapunov function $V(y) = \frac{1}{2}y^2$. As explained below, the discrete gradient is unique in the scalar case, and the trivial choice $\widetilde{L} = L$ leads to the well-known, second-order trapezoidal rule. With the aim of determining conditions for achieving higher order, we substituted the expansion in Equation (6) into a discrete gradient method and compared the result with the expansion of the exact solution $y(t) = e^{-\lambda t}$. The result was negative in the sense that no choice of matrices M_n can provide an order higher than two unless the matrix $\widetilde{L}$ is also dependent on the step size h. In fact, the only second-order method is the trapezoidal rule itself, arising from $M_n = 0$, i.e., the trivial choice $\widetilde{L} = L$. This discouraging result suggests that the search for optimal methods should not be guided exclusively by order conditions in general cases. Contrarily, we think that discrete gradient methods are well suited to particular classes of systems, where the preservation of dynamical properties is aimed rather than order alone.

Keeping the analysis within the one-dimensional case, but now allowing the matrix $L(y)$ of the continuous system to be non-constant, the above expansion leads to the second-order condition:

$$\left.\frac{\partial \widetilde{L}}{\partial z}\right|_{z=y} = \frac{1}{2}\frac{dL}{dy}$$

This result suggests some formulas where the role of the variables $\vec{y}, \vec{z}$ is symmetrical, such as

$$\begin{aligned}\widetilde{L}(\vec{y},\vec{z}) &= L\left(\frac{1}{2}(\vec{z}+\vec{y})\right)\\ \widetilde{L}(\vec{y},\vec{z}) &= \frac{1}{2}\left(L(\vec{z}) + L(\vec{y})\right)\\ \widetilde{L}(\vec{y},\vec{z}) &= L\left(\sqrt{\vec{y}\,\vec{z}}\right)\end{aligned} \tag{7}$$

Interestingly, only based on empirical arguments, we adopted the latter setting in our analysis of Hopfield neural networks [24].

The rigorous foundation of the analysis of discrete gradient methods according to the choice of the metric matrix $\widetilde{L}$ is a problem still open, to the best of our knowledge, and an interesting avenue for further research.

3.2. Discrete Gradient

With regard to the discrete gradient, there is a unique discrete gradient for one-dimensional systems, and it is given by

$$\overline{\nabla} V(\vec{y},\vec{z}) = \frac{V(\vec{z}) - V(\vec{y})}{\vec{z} - \vec{y}} \tag{8}$$

However, in higher dimensions, a wide variety of discrete gradients exist (see [18] and references therein). Some of the most commonly used include:

- The mean value discrete gradient:

$$\overline{\nabla} V(\vec{y},\vec{z}) = \int_0^1 \nabla V((1-t)\vec{y} + t\,\vec{z})$$

- The midpoint discrete gradient:

$$\overline{\nabla} V(\vec{y},\vec{z}) = \nabla V\left(\frac{1}{2}(\vec{y}+\vec{z})\right) + \frac{V(\vec{z}) - V(\vec{y}) - \nabla V\left(\frac{1}{2}(\vec{y}+\vec{z})\right)\cdot(\vec{z}-\vec{y})}{|\vec{z}-\vec{y}|^2}(\vec{z}-\vec{y})$$

- The coordinate increment discrete gradient also called the Itoh–Abe discrete gradient [28]:

$$\overline{\nabla} V(\vec{y},\vec{z}) = \begin{pmatrix} \dfrac{V(z_1,y_2,\ldots,y_n) - V(y_1,y_2,\ldots,y_n)}{z_1-y_1} \\ \dfrac{V(z_1,z_2,y_3,\ldots,y_n) - V(z_1,y_2,\ldots,y_n)}{z_2-y_2} \\ \vdots \\ \dfrac{V(z_1,\ldots,z_{n-2},z_{n-1},y_n) - V(z_1,\ldots,z_{n-2},y_{n-1},y_n)}{z_{n-1}-y_{n-1}} \\ \dfrac{V(z_1,\ldots,z_n) - V(z_1,\ldots,z_{n-1},y_n)}{z_n-y_n} \end{pmatrix} \tag{9}$$

where a particular ordering $y_1, y_2, \ldots, y_n$ of the coordinates of the vector $\vec{y} \in \mathbb{R}^n$ is assumed.

As mentioned in [18], both the mean value and the midpoint discrete gradient are second-order approximations to $\nabla V\left(\frac{1}{2}(\vec{y}+\vec{z})\right)$, whereas the coordinate increment discrete gradient only provides a first-order approximation to the midpoint of the segment between the points $\vec{y}$ and $\vec{z}$. However, it is not clear whether this property translates into a higher-order method or is somehow relevant in practice. The coordinate increment discrete gradient can be interpreted as a piecewise linear path joining $\vec{y}$ and $\vec{z}$, each piece parallel to one of the coordinate axes, rather than along the straight segment $\vec{y}-\vec{z}$. In this paper, we will focus on the *coordinate increment discrete gradient* because it is easier to implement computationally. The mean value discrete gradient requires the computation of n integrals, whereas the application of the midpoint discrete gradient leads to a rather complicated expression for the method. In contrast, the coordinate increment discrete gradient results in explicit methods for some systems, such as Hopfield neural networks, where the matrix L is diagonal, and the Lyapunov function is multilinear. Nevertheless, the analysis and comparison of different discrete gradients well deserves more attention.

In the rest of this section, we undertake a study of discrete gradient methods, first by a preliminary order analysis, then by constructing different methods for simple scalar systems (this methodology is inspired by [29]) and observing that a suitable choice of the matrix $\widetilde{L}$ allows in some cases for rewriting the method in explicit form.

3.3. Order Analysis

The order of the obtained numerical method can be studied by the usual systematic procedure [10]: comparing the Taylor series expansion around $h = 0$ of both the exact solution of the system of differential equations and the approximate solution obtained by the numerical method. Note that the discrete gradient method is consistent by construction [18], so it achieves at least order one, i.e., the error after a single step is given by $\vec{y}(t+h) - \vec{z} = C\,h^2 + O(h^3)$, where C is the error constant of the method. A straightforward—but tedious—computation yields the error constant of the second order term:

$$C_{GD} = \left(\frac{1}{2}J^c - J^d\Big|_{h=0}\right) f(\vec{y}) \tag{10}$$

where $J^c(\vec{y})$ is the Jacobian matrix of f at $\vec{y}$:

$$J^c = \frac{\partial f}{\partial \vec{y}} = \left[\frac{\partial f_i}{\partial y_j}\right]_{ij} = \left[\frac{\partial (L(\vec{y}) \cdot \nabla V(\vec{y}))_i}{\partial y_j}\right]_{ij} \qquad i,j = 1,\ldots,n \tag{11}$$

and J^d is the Jacobian of $\widetilde{L} \cdot \overline{\nabla} V$, i.e.,:

$$J^d = \frac{\partial \left(\widetilde{L} \cdot \overline{\nabla} V\right)}{\partial \mathbf{z}} = \left[\frac{\partial \left(\widetilde{L} \cdot \overline{\nabla} V\right)_i}{\partial \mathbf{z}_j}\right]_{ij} \qquad i,j = 1,\ldots,n \tag{12}$$

so that the condition $J^c = 2\, J^d\Big|_{h=0}$ would ensure that the obtained discrete gradient method is second order. In principle, a suitable choice of parameters $\widetilde{L}$ and $\overline{\nabla} V$ could lead to a higher-order method. When this paper was already in preparation, a systematic analysis of discrete gradient methods was published [22], although in the somewhat different context of Hamiltonian systems. Adapting this framework to gradient-like systems is an interesting avenue for future research. Nevertheless, it must be emphasized that the search for higher accuracy, without any other consideration, defeats the purpose of structure-preserving methods. In this paper, we will not further pursue the analysis of order and error, focusing on the preservation of the Lyapunov function and stability.

3.4. The Scalar Linear ODE

For the purpose of illustration, in this section, we show the mechanism of obtaining a discrete gradient method as described above. As a case in point, consider the scalar linear homogeneous ODE:

$$\frac{dy}{dt} = -a\,y, \qquad y(0) = y_0 \tag{13}$$

with $a > 0$. By direct integration, it is straightforward to compute the analytical solution $y(t) = y_0\, e^{-a\,t}$, which shows that the origin is asymptotically stable whenever $a > 0$, since $\lim_{t\to\infty} y(t) = 0$. We can also state that $V = \frac{1}{2}\, y^2$ is a Lyapunov function for this system because:

$$\alpha(y) = \frac{dV}{dt} = \frac{dV}{dy}\frac{dy}{dt} = y\,(-a\,y) = -a\,y^2 < 0 \tag{14}$$

for all $y \neq 0$. To construct a discrete gradient method, the equation is cast into linear-gradient form, thus, obtaining the definitions $L(y) = -a$, $\nabla V = y$. Therefore, the discrete gradient is

$$\overline{\nabla} V(y,z) = \frac{V(z) - V(y)}{z - y} = \frac{1}{2}\,\frac{z^2 - y^2}{z - y} = \frac{1}{2}\,(z + y) \tag{15}$$

and, with the trivial choice $\widetilde{L} = L = -a$, the discrete gradient method results:

$$z = y + h\,\widetilde{L}(y,z)\,\overline{\nabla} V(y,z) = y + \frac{h}{2}\,(-a\,z - a\,y) = y + \frac{h}{2}\,(f(z) + f(y)) \tag{16}$$

Now it is obvious that, in this case, the discrete gradient method turns out to be simply the trapezoidal rule, which is a second-order method. The fact that the trapezoidal rule preserves the stability of scalar linear ODEs for *any* step size h is already explained by the classical theory of numerical methods for stiff systems since it is well-known that the trapezoidal rule is A-stable, thus, nothing new seems to be provided by the proposal of discrete gradient methods. However, the point is that the choice of the matrix $\widetilde{L}$ is not unique, so a different definition $\widetilde{L}$, possibly depending on $\vec{z}$ and h, would lead to a different method. In addition, if we are not interested in preserving a particular Lyapunov function, but only the qualitative stability of the system, we could choose a different Lyapunov function, thus, leading to a different discrete gradient method.

3.5. The Logistic Equation

Consider next the IVP given by the generalization of the usually called logistic differential equation:

$$\frac{dy}{dt} = a\,y\,(1-y)\,, \qquad y(0) = y_0 \tag{17}$$

By straightforward integration, the exact solution can be computed:

$$y(t) = \frac{1}{1 + \left(\dfrac{1}{y_0} - 1\right) e^{-at}} \tag{18}$$

for any initial condition $y_0 \neq 0$, whereas the trivial solution $y(t) = 0$ involves a fixed point. We also choose $y_0 > 0$ to avoid the need to consider unbounded solutions. There are several ways to check that the equilibrium $y^* = 1$ is asymptotically stable, for example, the Jacobian of the ODE given by Equation (17) is negative at $y = 1$ or the limit when $t \to \infty$ of the exact solution given by Equation (18) is 1.

The construction of a discrete gradient method as in Equation (4) requires, first, writing the system in linear-gradient form from the knowledge of a Lyapunov function V; and then choosing the method parameters, $\widetilde{L}(y,z,h)$ and $\overline{\nabla}V(y,z)$, while fulfilling the conditions that guarantee the consistency of the method. Interestingly, even such a simple example as the logistic ODE can lead to completely different discrete gradient methods.

Firstly, observe that the function $V = \dfrac{1}{2}(1-y)^2$ fulfills the conditions required by Definition (1) to be a Lyapunov function. In particular, its time derivative is

$$\frac{dV}{dt} = \nabla V \cdot f = -(1-y)\,ay\,(1-y) = -ay\,(1-y)^2 < 0 \tag{19}$$

whenever $y > 0, y \neq 1$. Therefore, V is a Lyapunov function of Equation (17) at $y^* = 1$ that is valid for any initial value $y_0 > 0$. Then, the ODE can be rewritten in linear-gradient form as in Equation (2) by defining $L(y) = -a\,y$, so that the system is expressed as

$$\frac{dy}{dt} = a\,y\,(1-y) = L(y)\,\nabla V \tag{20}$$

with $L(y)$ negative-definite for $y > 0$, as required. Then, the discrete gradient is defined by the unique choice existing in the scalar case:

$$\begin{aligned} \overline{\nabla}V(y,z) &= \frac{V(z)-V(y)}{z-y} = \frac{1}{2}\,\frac{(1-z)^2-(1-y)^2}{z-y} = \\ &= \frac{1}{2}\,\frac{-2\,(z-y)+(z^2-y^2)}{z-y} = \frac{-2+(z+y)}{2} \\ &= -\left(1-\frac{z+y}{2}\right) \end{aligned} \tag{21}$$

The last equality of Equation (21) has been included to point out a plausible interpretation of the discrete gradient as a sort of *midpoint* gradient, since it is identical to the gradient of V, replacing the variable y with the average $\dfrac{z+y}{2}$. With regard to the choice of $\widetilde{L}(y,z,h)$, there are several consistent options. For simplicity, we adopt the trivial setting

$\widetilde{L} = L$. Therefore, if we substitute the chosen parameters in Equation (4), the method is obtained:

$$\begin{aligned} z &= y + h\,\widetilde{L}(y,z,h)\cdot\overline{\nabla}V(y,z) = y - a\,h\,y\left(\frac{-2+z+y}{2}\right) \\ &= y + a\,h\,y - \frac{a\,h}{2}\,y\,z - \frac{a\,h}{2}\,y^2 \end{aligned} \tag{22}$$

which after straightforward algebra yields an explicit expression for z:

$$z = \frac{\left(1 + a\,h - \dfrac{a\,h}{2}\,y\right)\,y}{1 + \dfrac{a\,h}{2}\,y} \tag{23}$$

In this particular case, the choice of $\widetilde{L}$ has allowed for obtaining an explicit method. However, the procedure has some generality, at least restricted to one-dimensional ODEs: it can be proved that if the Lyapunov function V is quadratic and the matrix $\widetilde{L}$ is trivially set to $\widetilde{L} = L$, the discrete gradient method can be cast into explicit form.

Remark 2 (Relation to known methods)**.** *Note that apparently Equation (23) cannot be derived as a conventional Runge–Kutta method (although proving this, in general, would require some work). In contrast, the* nonlocal *substitution $y^2 \to yz$ and the use of the discrete gradient remind us of nonstandard finite difference schemes [30], while providing a systematic methodology for their construction.*

Consider now the function $V = -\frac{1}{2}y^2 + \frac{1}{3}y^3$ as a candidate for the Lyapunov function of the same system, and observe that it fulfills the conditions required by Definition (1). In particular, the time derivative is

$$\frac{dV}{dt} = \nabla V \cdot f = -y\,(1-y)\,ay\,(1-y) = -ay^2\,(1-y)^2 < 0 \tag{24}$$

whenever $y \neq 0, 1$. Therefore V is a Lyapunov function of Equation (17) for the stable equilibrium point $y^* = 1$, which is valid for any initial value $y_0 > 0$. Then, the ODE can be rewritten in linear-gradient form as in Equation (2) by defining $L(y) = -a$ so that the linear gradient form $\frac{dy}{dt} = ay\,(1-y) = L(y)\,\nabla V$ holds too with these new parameters and $L(y)$ is negative-definite, as required. The one-dimensional discrete gradient has the same form as before, but the Lyapunov function V is different, to begin with, leading to

$$\begin{aligned} \overline{\nabla}V(y,z) = \frac{V(z)-V(y)}{z-y} &= \frac{-\frac{1}{2}(z^2-y^2) + \frac{1}{3}(z^3-y^3)}{z-y} = \\ &= -\frac{1}{2}(z+y) + \frac{1}{3}(z^2+zy+y^2) \end{aligned} \tag{25}$$

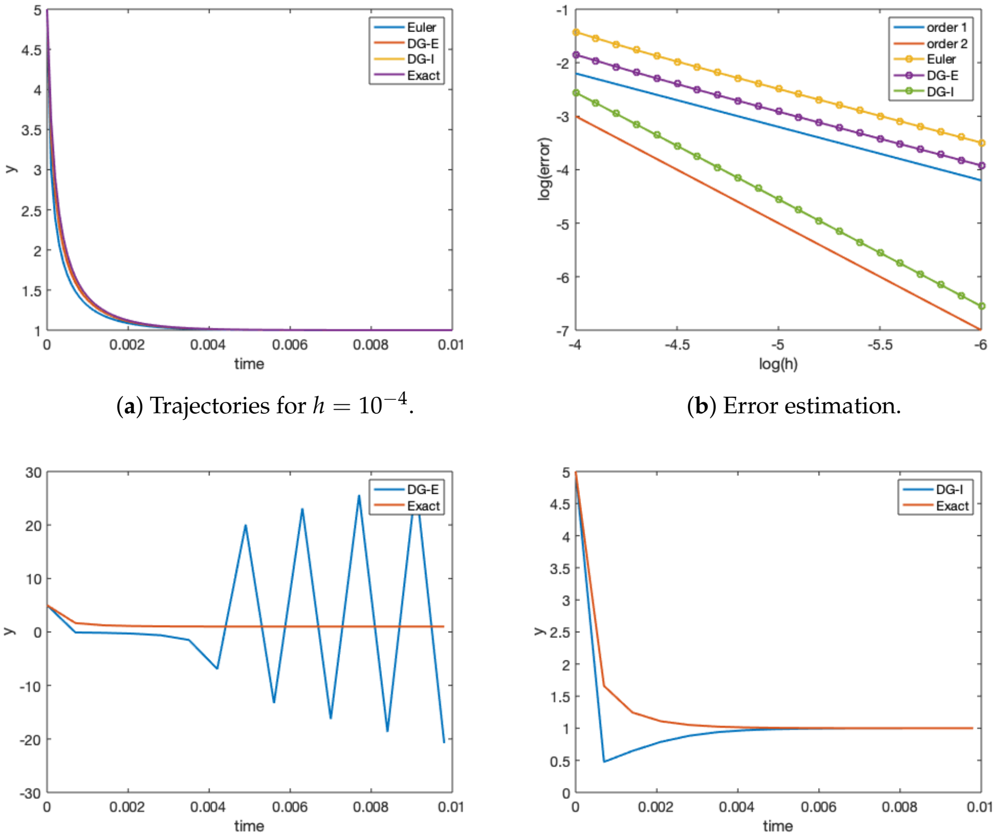
With regard to the choice of $\widetilde{L}(y,z,h)$, for simplicity we again adopt the trivial setting $\widetilde{L} = L$. Therefore, if we substitute the chosen parameters in Equation (4), the new method is obtained:

$$z = y + h\,\widetilde{L}(y,z,h)\cdot\overline{\nabla}V(y,z) = y - ah\left[-\frac{1}{2}(z+y) + \frac{1}{3}(z^2+zy+y^2)\right] \tag{26}$$

In this case, we obtain an implicit method. In order to apply Newton's method to obtain the solutions, we can rewrite the method as a function of z as shown below:

$$F(z) = \left(\frac{1}{3}ah\right)z^2 + \left(1 - \frac{1}{2}ah + \frac{1}{3}ahy\right)z + \left(-1 - \frac{1}{2}ah + \frac{1}{3}ahy\right)y = 0 \tag{27}$$

We have implemented the explicit discrete gradient method (DG-E) given by Equation (23) and the implicit scheme (DG-I) from Equation (27). Both are applied to the same logistic ODE, choosing the parameter as $a = 1000$ and the initial value $y_0 = 5$. The resulting trajectories are shown in Figure 1 for different values of the step size h. When h is small enough, all methods provide qualitatively correct solutions, as shown in Figure 1a. Aside from both discrete gradient methods derived above, the Euler rule has been included for comparison. To have a glimpse at the approximation accuracy achieved by each method, the global error has been computed by subtracting the discrete sequence from the exact solution and averaging over all the computed steps. The obtained results for 20 different values of the step size in the interval $h \in [10^{-6}, 10^{-4}]$ are shown in Figure 1b, in logarithmic scale. Two straight lines with slopes 1 and 2 are added to ease the comparison. It is clear that both the Euler rule and DG-E are first-order methods. Unexpectedly, DG-I turns out to be a second-order method, even though the construction procedure has been identical. As said above, order analysis of discrete gradient methods is an interesting avenue for further research.



(**a**) Trajectories for $h = 10^{-4}$.

(**b**) Error estimation.

(**c**) Trajectory for DG-E with $h = 7 \times 10^{-4}$.

(**d**) Trajectory for DG-I with $h = 7 \times 10^{-4}$.

Figure 1. Solutions for the logistic equation obtained by the Euler method, the explicit method in Equation (23) (DG-E), the implicit method in Equation (27) (DG-I), and the exact solution.

The picture changes radically when the step size is increased, even modestly to $h = 7 \times 10^{-4}$. To begin with, the trajectory computed by the Euler rule blows up to infinity, so it is not represented. Remarkably, the problem is not that of insufficient order: we tested an implicit Runge–Kutta method of order 2 (the basis of the *ode23s* function in the *Matlab ODE Suite*), and it also produced unbounded solutions. This is a significant finding as methods designed for stiff differential equations are often assumed to reproduce better the qualitative behavior, which is not the case here. Regarding the explicit discrete gradient method DG-E, its trajectory remains bounded, at least within the computed range, but the qualitative behavior is completely wrong, as shown in Figure 1c. Instead of convergence to the equilibrium, undamped oscillations appear that destroy stability. In contrast, the correct behavior is ultimately achieved by DG-I with the same step size, despite an initial transient, plotted in Figure 1d.

The apparent contradiction between the proved preservation of the Lyapunov function and the oscillatory solution provided by DG-E is explained by the local nature of the chosen Lyapunov function $V = \frac{1}{2}(1-y)^2$. The condition $\frac{dV}{dt} < 0$ checked in Equation (19) only holds for $y < 0$. This fact is dismissed in the original system since the region $y < 0$ cannot be reached from a positive initial value. However, the discretization does take a step so large that the solution becomes negative. Another approach that offers insight into the different behavior of the discretizations is the analysis of the basins of attraction, i.e., the sets of initial values such that trajectories converge towards the equilibrium. Basins of attraction of ODEs are continuous, so the restriction $y > 0$ for the validity of the first Lyapunov function is irrelevant for the continuous system. In contrast, basins of attraction of discrete dynamical systems, such as the one defined by a numerical method, may be formed by disconnected sets. If the discrete steps of the numerical method drive the system to a region where $\frac{dV}{dt} < 0$ is no longer true, the discrete gradient construction does not enforce stability. This suggests the first rule that must guide the construction of discrete gradient methods: find a Lyapunov function for which the requirement $\frac{dV}{dt} < 0$ holds universally (except for $\nabla V = 0$, of course), or at least, throughout a domain as large as possible.

4. Numerical Experiments

In this section, we show the result of several numerical experiments designed to show the satisfactory performance of the designed discrete gradient method, assessed in terms of its ability to preserve the qualitative properties of the dynamical system. We are primarily interested in preserving the stability of the system, which will be evidenced by decreasing values of the considered Lyapunov function along solution trajectories of the numerical approximation. As a suitable case study, we first propose the Duffing equation [19], for which a Lyapunov function is known.

The proposed method is compared with three conventional methods: the explicit Euler rule, a second-order Runge–Kutta method (RK2) that forms the basis of the *ode23s* function in the *Matlab ODE Suite*), and a fourth-order Runge–Kutta method (RK4), which the *Matlab ode45* function is based upon. Note that *ode23s* is an implicit method, well suited to stiff equations; thus, it is a strong competitor when the preservation of qualitative features is considered, whereas *ode45* is an explicit method design with a higher order of accuracy in mind. To carry out a fair comparison among methods, all experiments are carried out with a fixed step size. Needless to say, our work on the implementation of discrete gradient methods will eventually comprise variable step size mechanisms for error control.

For the sake of brevity, we have left aside a number of methods that could eventually have the same stability properties as discrete gradient methods. We have already mentioned the specialized Runge–Kutta methods that preserve the Lyapunov function [20], with the shortcoming that these favorable properties can only be proved for small enough step size. It is also worth mentioning the family of Rosenbrock–Wanner schemes [31], which were proposed mainly in the context of Differential-Algebraic Equations. It has been proved that these methods have favorable stability properties when implemented with complex coefficients [32]. Of particular interest for our work, is that these methods have been applied to the ODEs that result from the spatial discretization of Partial Differential Equation [33]. The behavior of all these alternative methods is often strongly dependent on time step-adjusting mechanisms, which we tried to avoid in this work, to keep the exposition as simple as possible.

All experiments have been performed with Matlab installed on a laptop equipped with an Intel Core i5-10210U processor at a base frequency of 1.6 GHz, and 8 GB of RAM. Since our aim was a proof of concept and comparison between different methods, we have not carried out an extensive optimization of either the code or the implementation.

The chosen methods are applied to the Duffing equation that can be written as a first-order system of ODEs $\frac{d\vec{y}}{dt} = f(\vec{y})$ according to:

$$\begin{aligned} \frac{dy_1}{dt} &= y_2 \\ \frac{dy_2}{dt} &= y_1 - b\,y_1^3 - a\,y_2 \end{aligned} \tag{28}$$

with $b \neq 0$ and $a > 0$. The system has three fixed points: $P_0 = (0,0)$, $P_1 = \left(\sqrt{1/b}, 0\right)$, and $P_2 = \left(-\sqrt{1/b}, 0\right)$. A straightforward linearization shows that P_0 is a saddle point, whereas P_1 and P_2 are stable equilibria. It is known that a Lyapunov function is defined by

$$V(y_1, y_2) = \frac{1}{2}\left(y_2^2 - y_1^2 + \frac{b}{2}\,y_1^4\right) \tag{29}$$

which has (local) minima at P_1 and P_2, since the gradient vanishes and the Hessian of V is positive definite at both these points. The gradient of V is the vector field:

$$\nabla V = \begin{pmatrix} -y_1 + b\,y_1^3 \\ y_2 \end{pmatrix} \tag{30}$$

that leads to the energy-decreasing condition:

$$\frac{dV}{dt} = \nabla V \cdot f(y) = -a\,y_2^2 \leq 0$$

Then, the system can be cast into the linear-gradient form, i.e.:

$$\frac{d\vec{y}}{dt} = \begin{pmatrix} 0 & 1 \\ -1 & -a \end{pmatrix}\begin{pmatrix} -y_1 + b\,y_1^3 \\ y_2 \end{pmatrix} \tag{31}$$

which entails the definition of the negative-definite matrix L:

$$L = \begin{pmatrix} 0 & 1 \\ -1 & -a \end{pmatrix} \tag{32}$$

Our implementation starts by computing the coordinate increment discrete gradient for the particular system given by Equation (28):

$$\overline{\nabla} V(y,z) = \frac{1}{2}\left(\begin{matrix} (z_1 + y_1)(-1 + \frac{b}{2}(z_1^2 + y_1^2)) \\ z_2 + y_2 \end{matrix}\right)$$

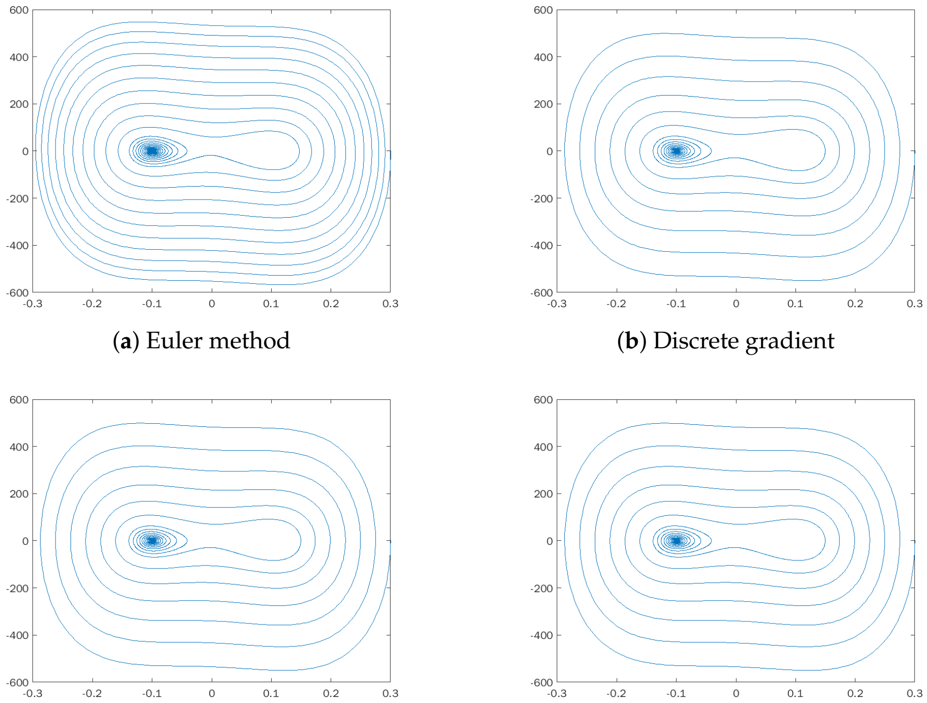
whereas we adopt the simplest approximation $\widetilde{L} = L$. Then, the discrete gradient method results:

$$z_1 = y_1 + \frac{h}{2}(z_2 + y_2)$$

$$z_2 = \frac{h}{2}\left[-(z_1 + y_1)\left(-1 + \frac{b}{2}(z_1^2 + y_1^2)\right) - a(z_2 + y_2)\right]$$

This implicit equation for $\vec{z}$ will be solved by Newton iteration until convergence at each time step.

All the experiments have been carried out considering $y_0 = (0.3, 0)$ as the initial point. We have designed three types of experiments. Firstly, we show the phase portrait that is obtained by applying each of the methods for different values of the step size h and compare it with the exact solution. Contrarily to the simple systems of the previous section, we do not have the benefit of an analytical solution, but we consider that the approximation obtained by Euler's method with $h = 10^{-8}$ is exactly up to machine precision. The results of this set of experiments are shown in Figures 2–4. It can be seen how the behavior of the discrete gradient method reproduces the phase portrait of the exact solution regardless of the step size. In contrast, Euler's rule does not converge with step sizes greater than 10^{-5}. As for the Runge–Kutta methods, both the order two and order four schemes fail when working with $h = 10^{-3}$. Both explicit methods, Euler and RK4, produce trajectories that blow up towards unbounded values; thus, they are not shown in the figures. This is the case for both methods with $h = 10^{-3}$ in Figure 4 and the Euler's method with $h = 10^{-4}$ in Figure 3. Despite Euler's rule producing a bounded trajectory that converges to the stable equilibrium for a small enough step size, the phase portrait is not correct. It is noticeable in Figure 2a) that the turns of the trajectory are closer than in other plots, suggesting that the numerical method is introducing a spurious dissipation.



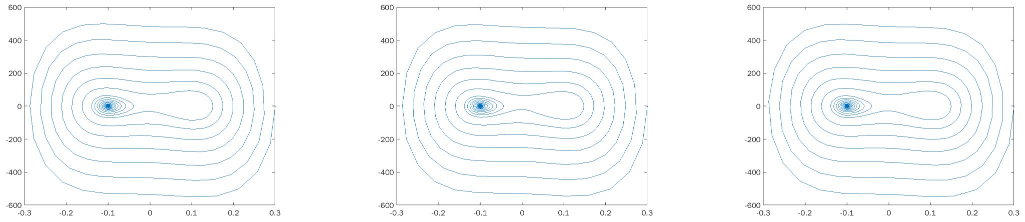
(**a**) Euler method

(**b**) Discrete gradient

(**c**) RK of order 2

(**d**) RK of order 4

Figure 2. Phase portrait for $h = 10^{-5}$.

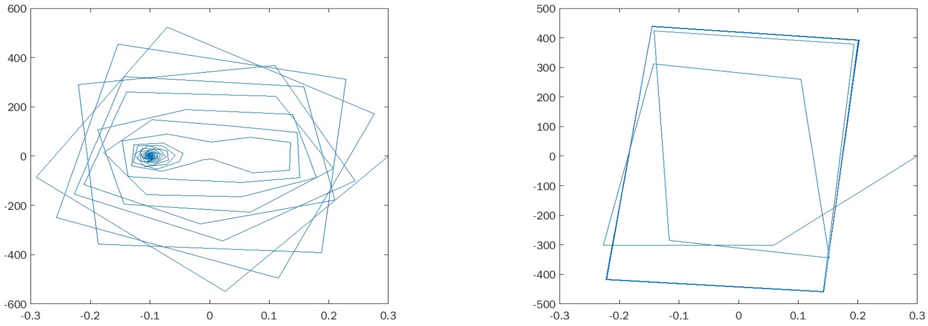


(**a**) Discrete gradient

(**b**) RK of order 2

(**c**) RK of order 4

Figure 3. Phase portrait for $h = 10^{-4}$.

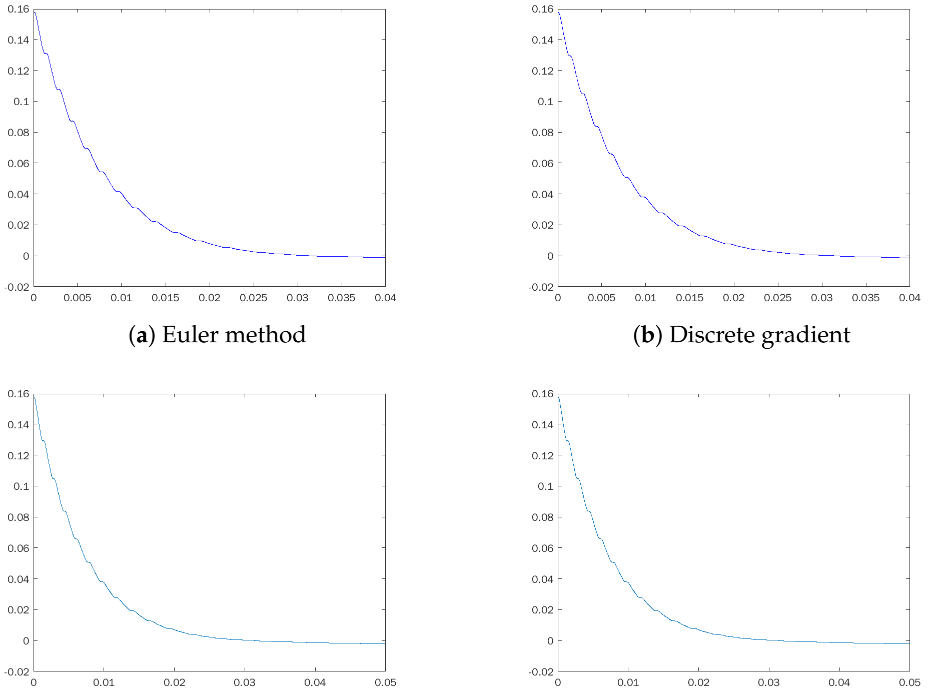


(**a**) Discrete gradient

(**b**) RK of order 2

Figure 4. Phase portrait for $h = 10^{-3}$.

On the other hand, taking into account that the fundamental objective of the designed method is the conservation of the Lyapunov function, we have designed another set of experiments focused on showing the behavior of the Lyapunov function with respect to time. Table 1 shows the values of the maximum increment of V for each method and each step size used. We also plot in Figures 5–8 the trajectories of the value of V for different step sizes. In general, it can be seen on the graphs that the Lyapunov function is decreasing along trajectories of the discrete gradient method, as expected by construction. The small positive increments shown in the table are within the range of machine precision, so they are attributed to rounding rather than the numerical method. In contrast, much larger increases in the Lyapunov function are visible in Figure 6 when using Euler's method with $h = 10^{-5}$, even though for this step size, the trajectories of the solution converge to the equilibrium. For large step sizes such as $h = 10^{-3}$, only the implicit RK2 among conventional methods provides bounded trajectories. However, the evolution of V shown in Figure 8 reveals, even more clearly than the phase portrait, that the behavior of the system is qualitatively corrupted. Periodic oscillations of V prove that the system is not approaching equilibrium and the Lyapunov function is no longer decreasing.



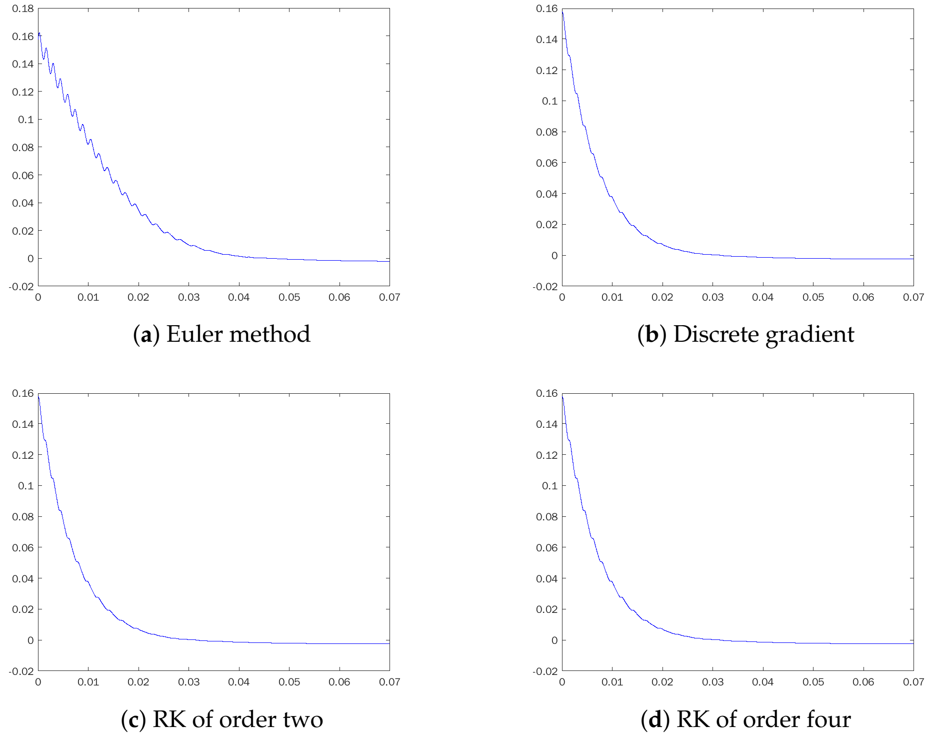
(**a**) Euler method

(**b**) Discrete gradient

(**c**) RK of order two

(**d**) RK of order four

Figure 5. Lyapunov function $h = 10^{-6}$.



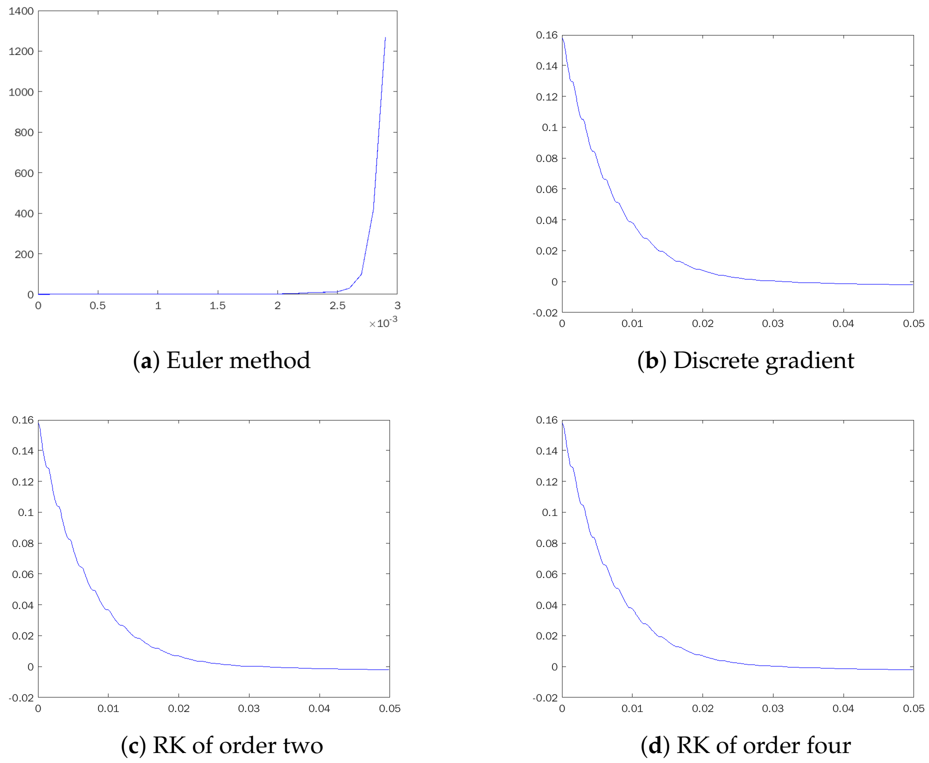
(**a**) Euler method

(**b**) Discrete gradient

(**c**) RK of order two

(**d**) RK of order four

Figure 6. Lyapunov function $h = 10^{-5}$.



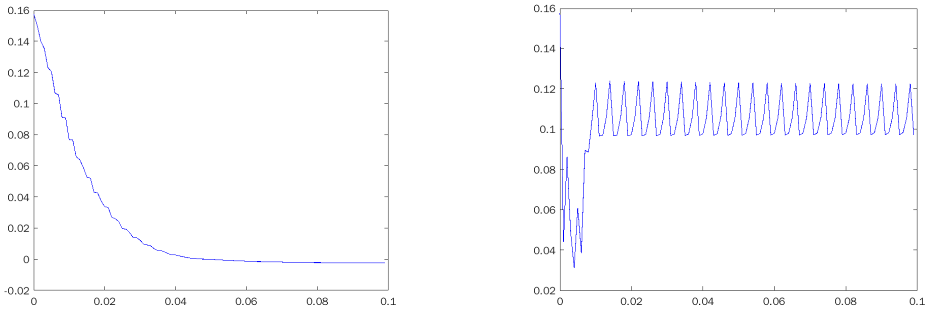
(**a**) Euler method

(**b**) Discrete gradient

(**c**) RK of order two

(**d**) RK of order four

Figure 7. Lyapunov function for $h = 10^{-4}$.



(**a**) Discrete gradient (**b**) RK of order two

Figure 8. Lyapunov function for $h = 10^{-3}$.

Table 1. Results of numerical experiments for the Duffing ODE.

Step Size h	Method	Comp. Time	max ΔV
10^{-3}	Euler	-	∞
	RK4	-	∞
	RK2	0.0146	0.0510
	GD	0.0069	1.3010×10^{-18}
5×10^{-4}	Euler	0.0027	∞
	RK4	0.0533	8.6736×10^{-19}
	RK2	0.0295	0.0091
	GD	0.0104	1.3010×10^{-18}
10^{-4}	Euler	0.0027	∞
	RK4	0.3558	8.6736×10^{-19}
	RK2	0.1299	8.6736×10^{-19}
	GD	0.0399	1.3010×10^{-18}
5×10^{-5}	Euler	0.0027	∞
	RK4	0.6885	1.3010×10^{-18}
	RK2	0.2429	1.3010×10^{-18}
	GD	0.1270	1.3010×10^{-18}
10^{-5}	Euler	0.6185	2.8800×10^{-4}
	RK4	3.2979	1.3010×10^{-18}
	RK2	1.2253	1.3010×10^{-18}
	GD	0.5220	1.3010×10^{-18}
5×10^{-6}	Euler	1.3717	7.2×10^{-5}
	RK4	7.0309	1.3010×10^{-18}
	RK2	2.4288	1.3010×10^{-18}
	GD	1.2380	1.3010×10^{-18}
10^{-6}	Euler	3.4330	2.88×10^{-6}
	RK4	24.2176	1.7347×10^{-18}
	RK2	11.5226	1.3010×10^{-18}
	GD	4.7109	1.7347×10^{-18}

Even when competitor conventional methods converge to a stable equilibrium, the proposed method is favorable in terms of computational cost. This is illustrated in Figure 9, showing the real computation time for the different step sizes. The computing times are also shown in Table 1 for each combination of step size and method.

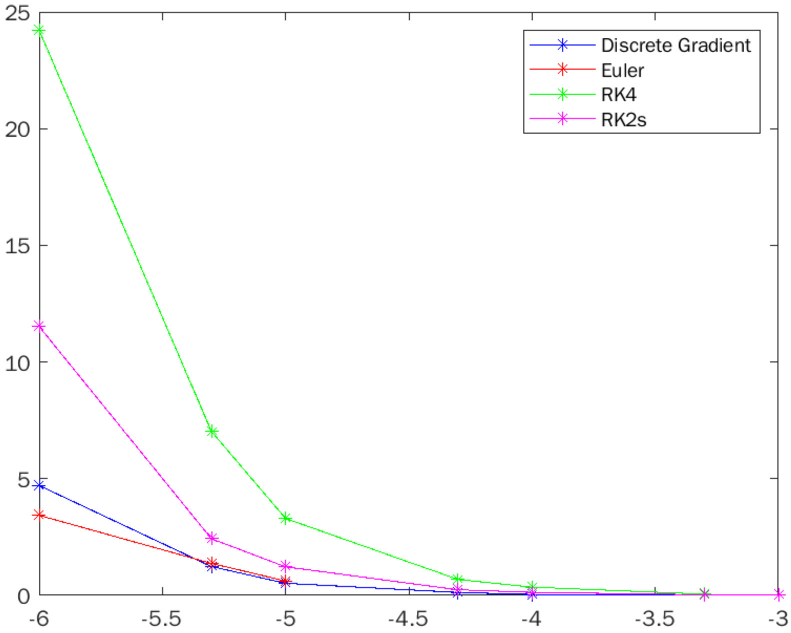


Figure 9. Computational cost for different values of h.

We briefly review the application of discrete gradient methods to a different class of systems, namely those with orbital stability [25]. In this case, the attractor is not a single point, but a compact subset. Trajectories with initial values within the attractor remain confined to it, which is, thus, termed an invariant set. The Lyapunov function is constant throughout the attractor, whereas its value is higher at any point outside the attractor. As an example of such orbitally stable systems, we propose the following ODE [34,35]:

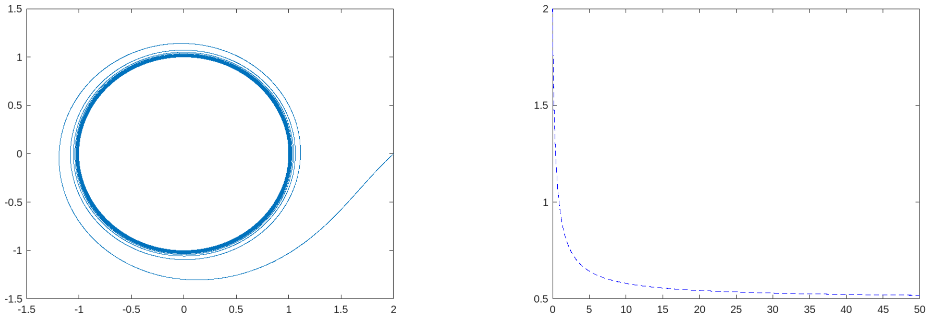
$$\begin{aligned} \frac{dy_1}{dt} &= -y_2 - y_1\left(1-\sqrt{y_1^2+y_2^2}\right)^2 \\ \frac{dy_2}{dt} &= y_1 - y_2\left(1-\sqrt{y_1^2+y_2^2}\right)^2 \end{aligned} \tag{33}$$

with the Lyapunov function $V(\vec{y}) = \frac{1}{2}(y_1^2+y_2^2)$. We can check that the time derivative of the Lyapunov function is:

$$\frac{dV}{dt} = -(y_1^2+y_2^2)\left(1-\sqrt{y_1^2+y_2^2}\right)^2 \leq 0$$

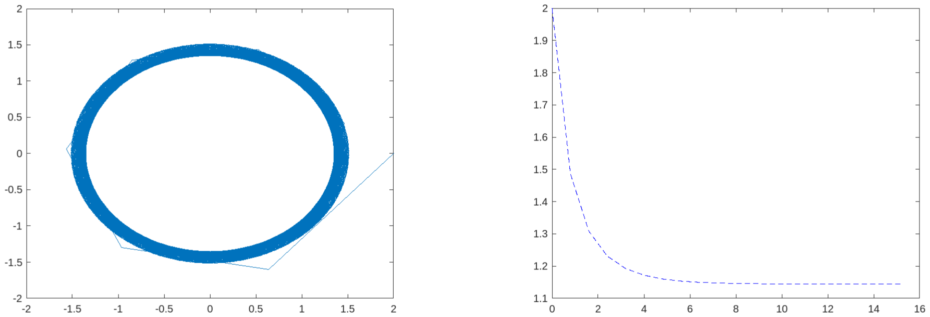
and $\frac{dV}{dt} = 0$ at the origin and on the circle of unit radius. Consequently, trajectories that start outside the circle approach the circle, whereas trajectories that start inside the circle are attracted to the origin. A continuous trajectory that starts outside cannot traverse the circle, but a discretization step risks *jumping* to the interior where the attractor nature of the circle is lost.

It has been shown [34,35] that conventional numerical methods produce trajectories that fall into the circle, thus, exhibiting a completely wrong behavior. We have implemented a discrete gradient method for the system in Equation (33) with the easiest choice $\widetilde{L} = L$ and the coordinate increment discrete gradient. We have set the initial point $\vec{y}_0 = (2,0)$. The results are shown in Figure 10, where a relatively small step size ($h = 10^{-3}$) has been set. It is clear that the trajectory approaches smoothly the circle and gets trapped by it, showing the invariant nature of the attractor. When we implement a large step size $h = 0.8$, such large discretization steps cannot reproduce faithfully the circle, as shown in Figure 11. However, the qualitative behavior is correct, and the trajectory does not fall into the origin, but remains orbiting, thus, reproducing the dynamical properties of the continuous system.



(**a**) Phase portrait for $h = 10^{-3}$ (**b**) Lyapunov function for $h = 10^{-3}$

Figure 10. Discretization of a system with orbital stability by the discrete gradient method and $h = 10^{-3}$.



(**a**) Phase portrait for $h = 0.8$ (**b**) Lyapunov function for $h = 0.8$

Figure 11. Discretization of a system with orbital stability by the discrete gradient method and $h = 0.8$.

5. Conclusions

We have presented a methodology for the implementation of numerical integrators that preserve a Lyapunov function of a dynamical system, namely discrete gradient methods. The analysis is performed on the proposed method, establishing that it is, in principle, a first-order method, although the second-order term is computed, revealing the conditions for the method parameters under which a second-order method would be obtained. As a proof of concept, a discrete gradient method is applied to the logistic equation, revealing the variety of choices that can lead to different numerical schemes with qualitatively different behaviors. The proposed method has been applied to the integration of the Duffing equation, which is regarded as a suitable test system: different parameter sets lead to oscillatory and stiff systems, whereas the preservation of the Lyapunov function is more important than the accuracy of individual trajectories. Numerical experiments are also carried out to confirm the ability of discrete gradient methods to preserve the Lyapunov function, and the failure of standard Runge–Kutta codes for a wide range of step size values, since Lyapunov function increments occur, thus, stability is lost.

The proposed methodology has a number of shortcomings, first and foremost, the need to know the explicit expression of a Lyapunov function. This is out of the scope of the present work and must be guided by physical considerations. Interestingly, our work proves that different Lyapunov functions for the same system lead to completely different numerical methods. The interaction between results in the context of the application and theoretical results on the methods themselves should lead to further advances in this direction. Another significant limitation is the lack of a general form of the time-stepping formula, which must be derived ad-hoc once given the Lyapunov function. This fact supports the notion that discrete gradient methods will find their role in the integration of particular classes of systems, rather than lead to a commercial code of general applicability.

We are currently engaged in further research to extend the results of this paper in several directions. First, we are developing order conditions to obtain higher-order methods. Preliminary results show that this is possible, at least for order two, by defining the matrix $\widetilde{L}$ dependant not only on $\vec{y}$ and $\vec{z}$ but also on h. Another promising line considers composition and splitting techniques. The long-term objective would be to establish a systematic order theory for designing discrete gradient methods of arbitrary orders, in line with the recent paper [22]. We are also trying to generalize the conditions for obtaining explicit methods, based on the original, implicit formulation.

This work suggests that general-purpose integrators are unable to keep pace with methods specifically designed to preserve the Lyapunov function. Thus we are extending our experiments to compare discrete gradient methods to both projection methods and Radau algorithms. In particular, it has been argued [20] that Radau methods are favorable due to their superior damping of high frequencies. In our experiments, we have detected that some discrete gradient methods possess an enhanced ability to deal with highly oscillatory systems. This question undoubtedly deserves deeper attention. It also must be taken into account that the results of this paper are a proof of concept, and much more can be done regarding the implementation refinements of discrete gradient methods. The obvious advance is the inclusion of an error control device, which could derive from detecting the lack of convergence of the Newton iteration. An improved discrete gradient method could be a serious competitor in applications where preserving the qualitative dynamical behavior is more important than the stringent accuracy of individual trajectories. For such systems, the integrators that preserve the Lyapunov function for arbitrary step sizes, such as discrete gradient methods, are endorsed as first-line methods by our results.

Author Contributions: Both authors have contributed equally to all phases of this work. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially supported by Project PID2020-116898RB-I00 from the Ministerio de Ciencia e Innovación of Spain and Project UMA20-FEDERJA-045 from the Programa Operativo FEDER de Andalucía.

Data Availability Statement: Not applicable.

Acknowledgments: The thorough reading and useful remarks of the anonymous reviewers are gratefully acknowledged.

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

References

1. Hirsch, M.W.; Smale, S. *Differential Equations, Dynamical Systems, and Linear Algebra*; Academic Press: Cambridge, MA, USA, 1974.
2. Slotine, J.J.; Li, W. *Applied Nonlinear Control*; Prentice Hall: Englewood Cliffs, NJ, USA, 1991.
3. Vidyasagar, M. Are Analog Neural Networks Better Than Binary Neural Networks? *Circuits Syst. Signal Process.* **1998**, *17*, 243–270. [CrossRef]
4. Spong, M.W.; Vidyasagar, M. *Robot Dynamics and Control*; John Wiley & Sons: Hoboken, NJ, USA, 1989.
5. Nijmeijer, H.; van der Schaft, A. *Nonlinear Dynamical Control Systems*; Springer: Berlin/Heidelberg, Germany, 1990.
6. Marino, R.; Tomei, P. *Nonlinear Control Design*; Prentice Hall: London, UK, 1995.
7. Sastry, S.; Bodson, M. *Adaptive Control: Stability, Convergence, and Robustness*; Prentice Hall: Hoboken, NJ, USA, 1990.
8. Absil, P.A.; Sepulchre, R. Continuous dynamical systems that realize discrete optimization on the hypercube. *Syst. Control Lett.* **2004**, *52*, 297–304. [CrossRef]
9. Schropp, J. Using dynamical systems methods to solve minimization problems. *Appl. Numer. Math.* **1995**, *18*, 321–335. [CrossRef]
10. Hairer, E.; Nørsett, S.; Wanner, G. *Solving Ordinary Differential Equations I. Nonstiff Problems*; Springer: Berlin/Heidelberg, Germany, 1987.
11. Arnold, V.I. *Mathematical Methods of Classical Mechanics*, 2nd ed.; Number 60 in Graduate Texts in Mathematics; Springer: New York, NY, USA, 1997.
12. Hairer, E.; Lubich, C.; Wanner, G. *Geometric Numerical Integration*; Springer: Berlin/Heidelberg, Germany, 2002.

13. Stuart, A.; Humphries, A. *Dynamical Systems and Numerical Analysis*; Cambridge University Press: Cambridge, UK, 1996.
14. Quispel, G.R.W.; Turner, G.S. Discrete gradient methods for solving ODEs numerically while preserving a first integral. *J. Phys. A Math. Gen.* **1996**, *29*, L341–L349. [CrossRef]
15. Schropp, J. Conserving first integrals under discretization with variable step size integration procedures. *J. Comput. Appl. Math.* **2000**, *115*, 503–517. [CrossRef]
16. McLachlan, R.; Quispel, G.; Robidoux, N. Unified Approach to Hamiltonian Systems, Poisson Systems, Gradient Systems, and Systems with Lyapunov Functions or First Integrals. *Phys. Rev. Lett.* **1998**, *81*, 2399–2403. [CrossRef]
17. Sanz-Serna, J.M. Symplectic integrators for Hamiltonian problems: An overview. *Acta Numer.* **1992**, *1*, 243. [CrossRef]
18. McLachlan, R.; Quispel, R.; Robidoux, N. Geometric integration using discrete gradients. *Philos. Trans. R. Soc. Lond. Ser. A* **1999**, *357*, 1021–1045. [CrossRef]
19. Calvo, M.; Laburta, M.P.; Montijano, J.I.; Rández, L. Projection methods preserving Lyapunov functions. *BIT Numer. Math.* **2010**, *50*, 223–241. [CrossRef]
20. Hairer, E.; Lubich, C. Energy-diminishing integration of gradient systems. *IMA J. Numer. Anal.* **2014**, *34*, 452–461. [CrossRef]
21. Iserles, A. *A First Course in the Numerical Analysis of Differential Equations*, 2nd ed.; Cambridge Texts in Applied Mathematics; Cambridge University Press: Cambridge, UK; New York, NY, USA, 2009.
22. Eidnes, S. Order theory for discrete gradient methods. *BIT Numer. Math.* **2022**, *62*, 1207–1255 . [CrossRef]
23. Atencia, M.; Hernández, Y.; Joya, G.; Sandoval, F. Numerical Implementation of Gradient Algorithms. In *Advances in Computational Intelligence*; Rojas, I., Joya, G., Cabestany, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; Volume 7903, pp. 355–364.
24. Hernández-Solano, Y.; Atencia, M.; Joya, G.; Sandoval, F. A discrete gradient method to enhance the numerical behaviour of Hopfield networks. *Neurocomputing* **2015**, *164*, 45–55. [CrossRef]
25. Khalil, H.K. *Nonlinear Systems*; Prentice Hall: Hoboken, NJ, USA, 2002.
26. Hairer, E.; Iserles, A.; Sanz-Serna, J.M. Equilibria of Runge-Kutta methods. *Numer. Math.* **1990**, *58*, 243–254. [CrossRef]
27. Bárta, T.; Chill, R.; Fašangová, E. Every ordinary differential equation with a strict Lyapunov function is a gradient system. *Monatshefte Für Math.* **2011**, *166*, 57–72. [CrossRef]
28. Itoh, T.; Abe, K. Hamiltonian-conserving discrete canonical equations based on variational difference quotients. *J. Comput. Phys.* **1988**, *76*, 85–102. [CrossRef]
29. Ramos, J.; García-López, C. Piecewise-linearized methods for initial-value problems. *Appl. Math. Comput.* **1997**, *82*, 273–302. [CrossRef]
30. Mickens, R.E. *Advances in the Applications of Nonstandard Finite Difference Schemes*; World Scientific Publishing Company: Singapore, 2005.
31. Lang, J.; Verwer, J. ROS3P—An Accurate Third-Order Rosenbrock Solver Designed for Parabolic Problems. *BIT Numer. Math.* **2001**, *41*, 731–738. [CrossRef]
32. Al'shin, A.B.; Al'shina, E.A.; Limonov, A.G. Two-stage complex Rosenbrock schemes for stiff systems. *Comput. Math. Math. Phys.* **2009**, *49*, 261–278. [CrossRef]
33. Alonso-Mallo, I.; Cano, B. Efficient Time Integration of Nonlinear Partial Differential Equations by Means of Rosenbrock Methods. *Mathematics* **2021**, *9*, 1970. [CrossRef]
34. Grimm, V.; Quispel, G. Geometric Integration Methods that Preserve Lyapunov Functions. *BIT Numer. Math.* **2005**, *45*, 709–723. [CrossRef]
35. Calvo, M.; Hernández-Abreu, D.; Montijano, J.; Rández, L. On the preservation of invariants by explicit Runge-Kutta methods. *SIAM J. Sci. Comput.* **2006**, *28*, 868–885. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.