

Article

A Novel Second-Order Explicit Integration Method for Nonlinear Ordinary Differential Equations in Dynamics

Gorka Urkullu ^{1,*} , Ibai Coria ¹, Igor Fernández de Bustos ²  and Haritz Uriarte ³ ¹ Department of Applied Mathematics, University of the Basque Country, Plaza Ingeniero Torres Quevedo, n. 1, 48013 Bilbao, Spain; ibai.coria@ehu.eus² Department of Mechanical Engineering, University of the Basque Country, Plaza Ingeniero Torres Quevedo, n. 1, 48013 Bilbao, Spain; igor.fernandezdebustos@ehu.es³ Department of Graphic Design and Engineering Projects, University of the Basque Country, Plaza Ingeniero Torres Quevedo, n. 1, 48013 Bilbao, Spain; haritz.uriarte@ehu.eus

* Correspondence: gorka.urkullu@ehu.eus

Abstract

This paper introduces a new explicit integration method for second-order ordinary differential equations (ODEs) commonly encountered in engineering applications. Traditionally, these problems are solved either by reformulating them as first-order systems to apply one-step methods such as Runge–Kutta schemes, or by using direct second-order approaches widely adopted in linear dynamics, including the generalized- α , central difference, and Newmark methods. The proposed method is derived from a Taylor series expansion truncated at the third derivative, resulting in a fully explicit algorithm that requires only one function evaluation per time step. Similar to Newmark’s formulation, it includes adjustable parameters that allow the user to balance accuracy and stability. For a specific parameter choice, the method exhibits convergence and stability properties comparable to those of the central difference scheme. An important advantage is that it remains explicit even when nonlinearities depend on first-derivative terms. The paper presents a theoretical analysis covering stability, local truncation error, spectral properties, numerical damping, and period elongation. The method is validated through four test cases from multibody dynamics, including linear and nonlinear problems. Results demonstrate that the Explicit Integration Grade 3 (EIG-3) method achieves accuracy comparable to existing explicit second-order integrators while significantly reducing computational cost, particularly in nonlinear applications.



Academic Editor: Michel Chipot

Received: 5 February 2026

Revised: 6 March 2026

Accepted: 16 March 2026

Published: 19 March 2026

Correction Statement: This article has been republished with a minor change. The change does not affect the scientific content of the article and further details are available within the backmatter of the website version of this article.

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: second-order differential equations; explicit integration; numerical stability; convergence analysis; multibody dynamics

MSC: 65L03; 65L05; 65L07; 65L20

1. Introduction

The numerical integration of Ordinary Differential Equations (ODEs) is a foundational pillar of computational mathematics, with its theoretical framework rigorously established in the classic monographs of Butcher [1], Lambert [2], and Hairer et al. [3,4]. These works establish the theoretical underpinnings for various classes of integration schemes, spanning from Runge–Kutta and linear multistep families to extrapolation and Taylor-based methods, which collectively define the standard for solving initial value problems across diverse scientific domains. In the fields of structural mechanics and multibody dynamics, these

ODEs, typically second-order systems, emerge from the spatial semi-discretization of continuous domains. While the finite element method (FEM) is the most common source of such systems [5–7], they also arise frequently when applying boundary element (BEM), spectral, or collocation methods to partial differential equations.

For stiff systems where low-frequency modes dominate the response, implicit integrators such as the Newmark, Hilber–Hughes–Taylor (HHT), and generalized- α methods are the preferred choice [8–11]. These schemes offer unconditional stability and controllable numerical dissipation, properties that have been extensively analyzed for constrained mechanical systems and penalty formulations [12–15]. Conversely, explicit methods are mandatory for scenarios involving high-frequency phenomena, such as wave propagation, impact dynamics, and complex nonlinear multibody systems [16–19]. In these high-dimensional applications, including railway engineering and vehicle trackside simulations [20,21], the iterative overhead required by implicit solvers to ensure convergence becomes computationally prohibitive [22].

Despite their maturity, existing explicit integrators face a severe algorithmic trade-off between computational cost and physical consistency. The standard central difference (CD) method, while robust for linear systems, becomes implicitly coupled when velocity-dependent nonlinearities, such as nonlinear damping or Coriolis forces, are present, requiring iterative solvers that destroy its strictly explicit advantage [23,24]. Furthermore, while second-order explicit Runge–Kutta (RK2) methods handle nonlinearities effectively, they require multiple function evaluations per step, doubling the cost in large-scale simulations [25,26]. Simultaneously, explicit linear multi-step (LMM) methods, such as the Adams–Bashforth family, suffer from shrinking absolute stability regions as their order of accuracy increases, often failing in high frequency applications where robust stability is critical [27,28].

To address these limitations, the present work proposes EIG-3, a novel explicit integration method grounded in Nordsieck’s polynomial approximations [29] and the framework of Taylor-based high-order expansions. EIG-3 is designed to achieve second-order accuracy through a strictly single-stage, single-function evaluation architecture. Unlike RK2, EIG-3 provides a self-starting structure that achieves the same convergence order with half the evaluations per time step. Furthermore, by utilizing local higher-order information at the current integration point rather than historical data, it overcomes the stability degradation inherent to LMM schemes. Crucially, EIG-3 remains strictly explicit even in the presence of velocity-dependent nonlinearities, avoiding the iterative overhead of the CD method. Moreover, EIG-3 features a parametric framework that allows the user to modulate the algorithm’s stability boundaries and spectral properties. By selecting specific parameter values, the method’s stability limits and convergence characteristics can be tuned to suit the requirements of the physical system, offering a level of flexibility similar to the Newmark family, without sacrificing its single evaluation efficiency.

The remainder of this paper is structured as follows: Section 2 presents the mathematical derivation of the EIG-3 formulas. Section 3 details the stability boundaries and converge and spectral analysis [30–33]. Section 4 evaluates the algorithm’s performance, energy conservation, and computational efficiency through numerical benchmarks in multibody dynamics, and Section 5 concludes the work.

2. Explicit Integrator Algorithm

2.1. Generating Function by Taylor Series Expansion

Second-order ordinary differential equations (ODEs) appear in numerous engineering fields, such as mechanical, thermodynamic, and electronic engineering. Despite their different forms, they can be expressed in the general form:

$$\mathbf{x}^{(2)}(t) = \mathbf{g}(\mathbf{x}^{(1)}(t), \mathbf{x}(t), t), \quad (1)$$

with the initial conditions:

$$\mathbf{x}(t_0) = \mathbf{x}_0 \text{ and } \mathbf{x}^{(1)}(t_0) = \mathbf{x}_0^{(1)},$$

where $\mathbf{g}(\mathbf{x}^{(1)}(t), \mathbf{x}(t), t)$ is a general function defining the system.

Typically, the independent variable is time, and differentiation is indicated by dots. However, in this work, we adopt Lagrange notation, where derivatives are represented as $\dot{\mathbf{x}}(t) = \mathbf{x}^{(1)}(t)$. The task consists of determining a function $\mathbf{x}(t)$ that meets the differential Equation (1) for all $t \in [0, t_f]$, $t_f > 0$ while fulfilling the initial conditions.

Following an explicit integration scheme, the Taylor series expansion of the unknown variables is performed around time t_n and truncated at the second derivative term:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{x}_n^{(1)} \Delta t + \mathbf{x}_n^{(2)} \frac{\Delta t^2}{2}, \quad (2)$$

$$\mathbf{x}_{n+1}^{(1)} = \mathbf{x}_n^{(1)} + \mathbf{x}_n^{(2)} \Delta t, \quad (3)$$

where $n \in \{0, 1, \dots, N-1\}$, N represents the number of time steps, $\Delta t = t_f/N$, and $\mathbf{x}_n$, $\mathbf{x}_n^{(1)}$ and $\mathbf{x}_n^{(2)}$ are the approximations to $\mathbf{x}(t_n)$, $\mathbf{x}^{(1)}(t_n)$ and $\mathbf{x}^{(2)}(t_n)$ respectively, in which $t_n = n\Delta t$. Considering Equations (2) and (3), $\mathbf{x}_n^{(2)}$ is needed to advance.

The unknowns of the system are $\mathbf{x}_{n+1}$, $\mathbf{x}_{n+1}^{(1)}$, and $\mathbf{x}_n^{(2)}$, that are three terms with only the two Equations (2) and (3). The third equation that completes the system will obviously be the equilibrium expression (1). Note that all terms belong to the interval $[t_{n-1}, t_{n+1}]$.

To control the convergence and stability properties of the method, the highest-order derivative is interpolated using a parameters α_i , leading to the modified equations:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{x}_n^{(1)} \Delta t + \left[\alpha_0 \mathbf{x}_n^{(2)} + (1 - \alpha_0) \mathbf{x}_{n-1}^{(2)} \right] \frac{\Delta t^2}{2}, \quad (4)$$

$$\mathbf{x}_{n+1}^{(1)} = \mathbf{x}_n^{(1)} + \left[\alpha_1 \mathbf{x}_n^{(2)} + (1 - \alpha_1) \mathbf{x}_{n-1}^{(2)} \right] \Delta t. \quad (5)$$

We clarify that these are the linear interpolation coefficients used to approximate the second derivative within the time interval $[t_{n-1}, t_n]$. Specifically, α_0 is used for the displacement ($x(t)$) update formula, while α_1 is used for the velocity ($\dot{x}(t)$) update formula. This distinction is crucial for controlling the convergence and stability properties of the EIG-3 method.

2.2. Algorithm Implementation

The equations above are reorganized to isolate, $\mathbf{x}_n^{(2)}$, ensuring numerical stability and minimizing round-off errors. The equilibrium equation is rewritten as:

$$\mathbf{x}_n^{(2)} = \mathbf{g}_i(\mathbf{x}_n, \mathbf{x}_n^{(1)}, \mathbf{x}_{n-1}^{(2)}, t_n). \quad (6)$$

The computational procedure consists of the following steps:

1. Initialization: At each time step n , the values $\mathbf{x}_n$, $\mathbf{x}_n^{(1)}$ and $\mathbf{x}_{n-1}^{(2)}$ are known.
2. Compute $\mathbf{x}_n^{(2)}$: Use the equilibrium Equation (6) to determine it.
3. Update $\mathbf{x}_{n+1}$ and $\mathbf{x}_{n+1}^{(1)}$: Compute them using Equations (4) and (5).

For the first time step, where $\mathbf{x}_{n-1}^{(2)}$ is unknown, it can be computed directly from the equilibrium equation:

$$\mathbf{x}_0^{(2)} = g(\mathbf{x}_0, \mathbf{x}_0^{(1)}, t_0). \quad (7)$$

Thus, the initialization process follows these steps:

1. Compute $\mathbf{x}_0^{(2)}$ using the equilibrium Equation (7).
2. Use the Taylor series expansion to estimate $\mathbf{x}_1$ and $\mathbf{x}_1^{(1)}$, Equations (2) and (3).

For computational efficiency and numerical robustness, polynomials in Equations (2) and (4) are evaluated using Horner's method, which reduces the number of arithmetic operations and minimizes floating-point errors.

3. Analysis of the New Explicit Algorithm

The characteristics of the proposed explicit algorithm can be analyzed by considering the free vibration equation for a single degree of freedom (SDOF) undamped system:

$$x^{(2)}(t) = -\omega^2 x(t) + f(t), \quad (8)$$

and $x(t_0) = d$, $x^{(1)}(t_0) = v$, where ω is the undamped natural frequency of the system. To examine the stability and accuracy properties, it is sufficient to consider an arbitrary function $f(t) = 0$ for all $t \in [t_0, t_f]$.

3.1. Stability

The amplification matrix describes how the unknown variables (including derivatives) evolve over time. It is a fundamental tool for assessing the stability of numerical integration algorithms [34]. For the proposed explicit algorithms applied to Equation (8), the numerical scheme can be expressed as:

$$\mathbf{z}_{n+1} = \mathbf{A}\mathbf{z}_n. \quad (9)$$

To evaluate the stability, the advancement Formulas (4) and (5) and the equilibrium Equation (8) are reformulated. The objective is to group the unknown state variables at the next time step into a vector $\mathbf{z}_{n+1}$ and the known values into $\mathbf{z}_n$. Given that

$$\mathbf{z}_{n+1} = \begin{Bmatrix} x_{n+1} \\ \Delta t x_{n+1}^{(1)} \\ \Delta t^2 x_{n+1}^{(2)} \end{Bmatrix} \text{ and } \mathbf{z}_n = \begin{Bmatrix} x_n \\ \Delta t x_n^{(1)} \\ \Delta t^2 x_{n-1}^{(2)} \end{Bmatrix},$$

the following coupled linear system is obtained by rearranging these equations:

$$\mathbf{U}\mathbf{z}_{n+1} = \mathbf{P}\mathbf{z}_n.$$

The amplification matrix $\mathbf{A}$ is then derived as $\mathbf{A} = \mathbf{U}^{-1}\mathbf{P}$, which allows for the recursive calculation with Equation (9). The explicit components of matrices $\mathbf{U}$ and $\mathbf{P}$ are given by:

$$\mathbf{U} = \begin{pmatrix} 1 & 0 & -\frac{\alpha_0}{2} \\ 0 & 1 & -\alpha_1 \\ 0 & 0 & 1 \end{pmatrix},$$

$$\mathbf{P} = \begin{pmatrix} 1 & 1 & \frac{1-\alpha_0}{2} \\ 0 & 1 & 1-\alpha_1 \\ -\Omega^2 & 0 & 0 \end{pmatrix},$$

where $\Omega = \omega\Delta t$. It can be easily verified that matrix $\mathbf{U}$ is always invertible. Thus, the amplification matrix is:

$$\mathbf{A} = \begin{pmatrix} 1 - \frac{\alpha_0\Omega^2}{2} & 1 & \frac{1-\alpha_0}{2} \\ -\alpha_1\Omega^2 & 1 & 1-\alpha_1 \\ -\Omega^2 & 0 & 0 \end{pmatrix}. \quad (10)$$

The properties of an explicit algorithm can be inferred from the spectral characteristics of this matrix [11,35].

3.1.1. Stability Condition

A numerical method is stable if the spectral radius of the amplification matrix satisfies:

$$\rho(\mathbf{A}) = \max|\lambda(\mathbf{A})| < 1.$$

If $\mathbf{A}$ remains stable for every $\Omega \in [0, \Omega_c]$, with Ω_c a positive constant, the scheme is considered conditionally stable. Conversely, if $\mathbf{A}$ is stable for any $\Omega \in [0, \infty)$, it is termed unconditionally stable.

The amplification matrix also gives information about the numerical damping introduced by the integration method. To assess this, we define two key measures:

- Relative period error e_T :

$$e_T = \frac{\bar{T} - T}{T},$$

where the exact period is $T = 2\pi/\omega$ and the numerically obtained period is $\bar{T} = 2\pi/\bar{\omega}$.

- Algorithmic damping ratio $\bar{\xi}$, which measures the numerical dissipation.

The spectral properties of the method are characterized by the eigenvalues of the amplification matrix $\mathbf{A}$. These eigenvalues can be expressed in terms of the numerical frequency $\bar{\omega}$ and the algorithmic damping ratio $\bar{\xi}$ as follows:

$$\lambda_i = \exp[-\bar{\xi}\bar{\omega}\Delta t \pm i\bar{\omega}\Delta t] = \exp[-\bar{\xi}\bar{\Omega} \pm i\bar{\Omega}]. \quad (11)$$

3.1.2. Characteristic Polynomial

The characteristic polynomial of the amplification matrix is computed using the Faddeev–LeVerrier’s method [30,31], which provides:

$$p_{\mathbf{A}}(\lambda) = \sum_{j=0}^3 c_j \lambda^j.$$

The polynomial coefficients are computed recursively as follows:

$$\begin{aligned} \mathbf{B}_j &= \mathbf{A}\mathbf{B}_{j-1} + c_{3-j+1}\mathbf{I}, \\ c_{3-j} &= \frac{-1}{j} \text{tr}(\mathbf{A}\mathbf{B}_j) \quad j = 1, 2, 3, \end{aligned}$$

with $\mathbf{B}_0 = \mathbf{0}$ and $c_3 = 1$.

Thus, the characteristic polynomial has the following form:

$$p_{\mathbf{A}}(\lambda) = c_3\lambda^3 + c_2\lambda^2 + c_1\lambda + c_0, \quad (12)$$

where

$$\begin{aligned} c_2 &= -2 + \frac{\alpha_0 \Omega^2}{2}, \\ c_1 &= 1 + \left(\frac{1}{2} - \alpha_0 + \alpha_1\right) \Omega^2, \\ c_0 &= (1 + \alpha_0 - 2\alpha_1) \frac{\Omega^2}{2}. \end{aligned} \quad (13)$$

The conditions that keep the spectral radius below one can be obtained by substituting $\lambda = e^{i\theta}$ into (12), where $\theta \in [0, 2\pi]$. Separating the real and the imaginary parts of the resulting equations, one obtains:

$$\cos \theta (\cos^2 \theta + c_2 \cos \theta + c_1) + c_0 = 0,$$

$$\sin \theta (\sin^2 \theta + c_2 \sin \theta + c_1) = 0.$$

For the limiting case $\theta = 0$, corresponding to $\lambda = 1$, the resulting equation is:

$$c_3 + c_2 + c_1 + c_0 = \Omega^2 = 0.$$

In this limiting case the eigenvalues of $\mathbf{A}$ are $\lambda_1 = \lambda_2 = 1$ and $\lambda_3 = 0$.

Similarly, for the other limiting case $\theta = \pi$, corresponding to $\lambda = -1$, one reaches:

$$-c_3 + c_2 - c_1 + c_0 = -4 + (2\alpha_0 - 2\alpha_1)\Omega^2 = 0.$$

In this limiting case, the condition is:

$$\Delta t \leq \frac{1}{\omega} \sqrt{\frac{4}{2\alpha_0 - 2\alpha_1}}. \quad (14)$$

To better illustrate the stability criteria derived in (14), Figure 1 presents the stable region in the $\alpha_0 - \alpha_1$ plane for $\omega = 10$ and $\omega = 1$. It can be observed that the maximum allowable time step Δt decreases as the frequency ω increases, but this effect can be “dumped” with an intelligent election of the parameters, restricting the stability zone to the shaded area defined by the inequality.

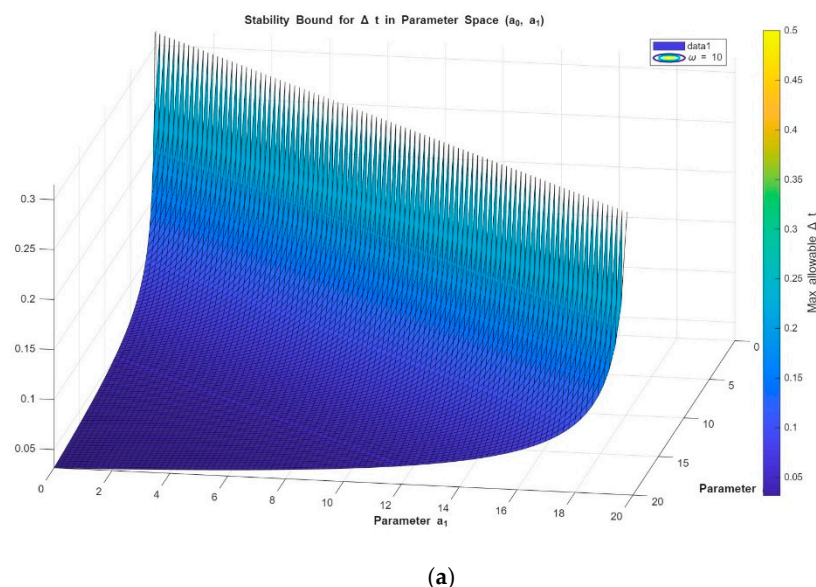


Figure 1. Cont.

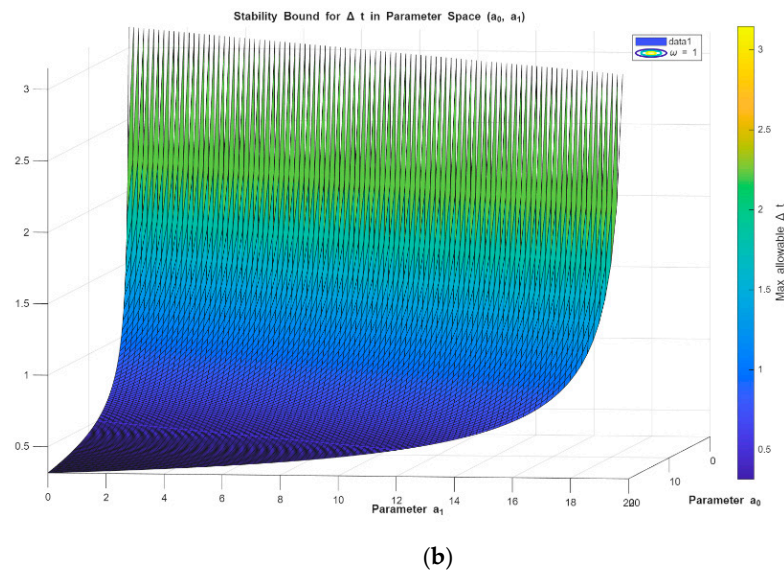


Figure 1. Stable region in the $\alpha_0 - \alpha_1$ plane for (a) $\omega = 10$ and (b) $\omega = 1$.

3.2. Consistency and Accuracy

The conventional approach to determining the local order of convergence using the amplification matrix is not the most straightforward in this case [10,11,32,35]. Instead, a more practical method is to approximate the exact solution by expanding the variables in a Taylor series truncated at the third derivative term. Under this framework, the proposed algorithm can be categorized based on the highest-order derivative included in its formulation, which is the third derivative. Consequently, the method introduced in this work is referred to as the explicit integration method of grade 3 (EIG-3).

Assuming that the truncation error in the second derivative term is negligible, $x_n^{(2)} - \bar{x}_n^{(2)} = 0$, we can express the approximation as:

$$x_n^{(2)} - \bar{x}_n^{(2)} = \left[x_{n-1}^{(2)} + x_{n-1}^{(3)} \Delta t \right] - \left[\alpha_2 \bar{x}_n^{(2)} + (1 - \alpha_2) x_{n-1}^{(2)} \right],$$

where terms with an overbar represent approximate values. Specifically, α_2 is the parameter defining the linear interpolation for the second-derivative approximation, consistent with the formulation previously established in Equations (4) and (5). Given that $x_{n-1}^{(2)} = x_n^{(2)} - x_n^{(3)} \Delta t$ and $x_{n-1}^{(3)} = x_n^{(3)}$, the local error $\mathbf{l}_n^{(2)}$ for the second derivative term is given by:

$$\mathbf{l}_n^{(2)} = x_n^{(2)} - \bar{x}_n^{(2)} = \left(1 - \frac{1}{\alpha_2} \right) x_n^{(3)} \Delta t = 0.$$

Using these expressions, the local errors for displacement $\mathbf{l}_n$ and velocity $\mathbf{l}_n^{(1)}$ are:

$$\mathbf{l}_n^{(1)} = \left(\frac{3}{2} - \alpha_1 \right) x_n^{(3)} \Delta t^2 + O(\Delta t^3), \quad (15)$$

$$\mathbf{l}_n = \left(\frac{2}{3} - \frac{\alpha_0}{2} \right) x_n^{(3)} \Delta t^3 + O(\Delta t^4).$$

As observed, the local error in velocity is at least second order, ensuring the consistency of the method. A numerical integration method is only valuable if it achieves at least second-order accuracy. As demonstrated in the convergence analysis, this condition is met when $\alpha_1 = 3/2$, as reflected in Equation (15).

3.3. Comparison of Algorithms

A set of parameters is sought to satisfy three key properties:

1. Maximizing stability: ensuring the algorithm has the largest possible stability region.
2. Minimizing numerical damping: reducing artificial energy dissipation.
3. Optimizing convergence: selecting parameters that maximize the order of accuracy.

The parameter set is defined by the vector $\alpha = [\alpha_0, \alpha_1]$.

Fixing $\alpha_1 = 3/2$, the condition that ensures a real eigenvalue of modulus 1 is:

$$\Delta t \leq \frac{1}{\omega} \sqrt{\frac{4}{2\alpha_0 - 3}}. \quad (16)$$

This equation provides a stability boundary, assuming no other eigenvalues have a modulus greater than 1. For specific values of α_0 , simple expressions can be derived to determine the maximum stable step size Δt . For instance:

- When $\alpha_0 = 3$, stability is given for $\Delta t \leq 1.154/\omega$.
- When $\alpha_0 = 5$, stability holds for $\Delta t \leq 0.756/\omega$.
- The most favorable case is when $\alpha_0 = 2$, yielding stability for $\Delta t \leq 2/\omega$.

This last configuration offers an extended stability range, making it an optimal choice in terms of stability.

3.3.1. Spectral and Accuracy Comparisons

The numerical stability and performance of the EIG-3 method are evaluated through its spectral characteristics. Figure 2 illustrates the spectral radius (ρ), Figure 3 shows the algorithmic damping ratio (ξ), and Figure 4 presents the relative period error ($\frac{\bar{T}-T}{T}$).

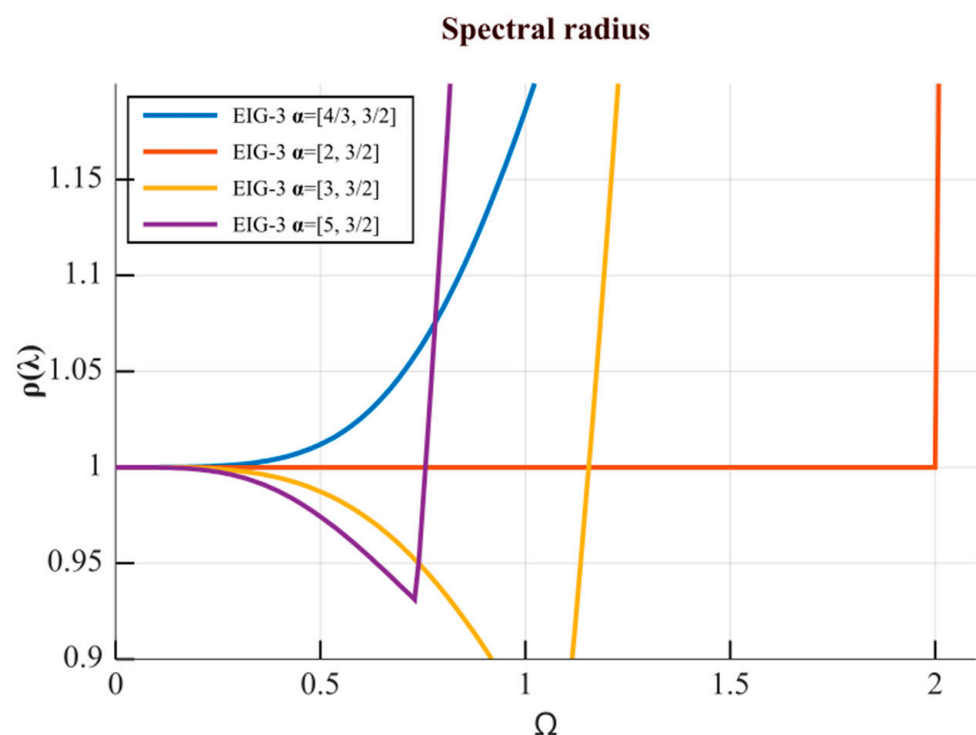


Figure 2. Comparison of spectral radius of different parameters for EIG-3 configurations.

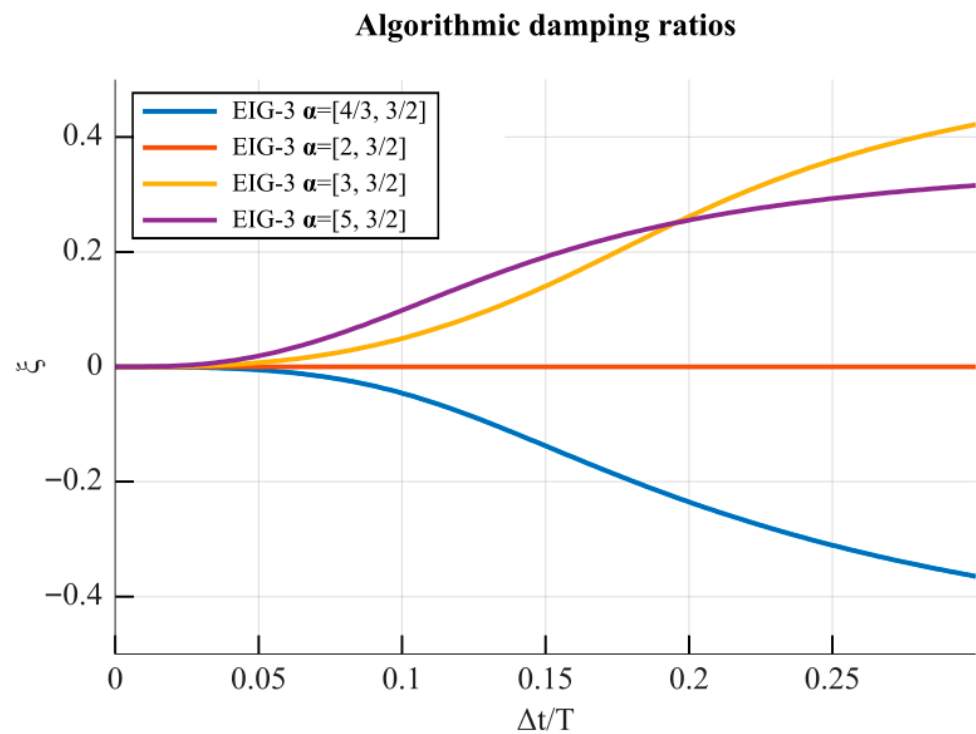


Figure 3. Comparison of algorithm damping ratio for different EIG-3 configurations.

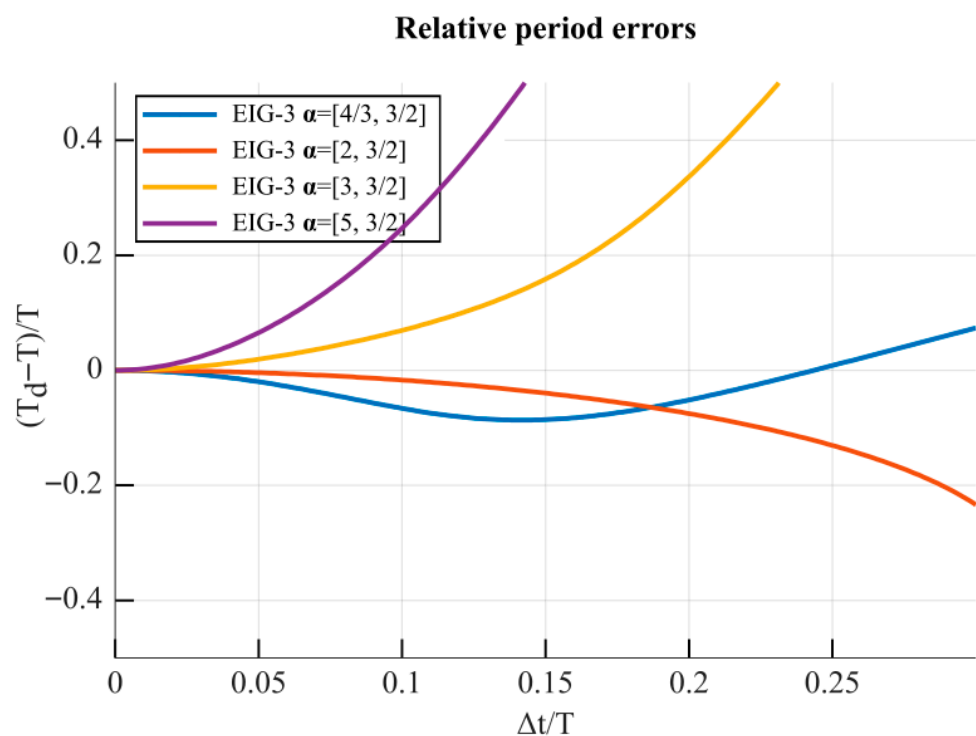


Figure 4. Comparison of the relative period error for different EIG-3 configurations.

These illustrations provide visual confirmation of the stability conditions derived in Section 3.1. Specifically, Figure 2 demonstrates that for the chosen parameter sets, the spectral radius remains below or equal to unity within the stable range of Ω , preventing the artificial growth of the solution. For $\alpha = [2, 3/2]$, the spectral radius is exactly equal to unity throughout the stable range, indicating a non-dissipative behavior. In this case,

the term c_0 associated with the third eigenvalue in the characteristic polynomial is zero, meaning the amplification matrix $\mathbf{A}$ has only two nontrivial eigenvalues:

$$\lambda_{1,2} = -\frac{(\Omega^2 - 2)}{2} \pm \sqrt{\left(\frac{\Omega^2 - 2}{2}\right)^2 - 1}. \quad (17)$$

As observed in Figure 3, this specific case results in zero algorithmic damping ($\zeta = 0$), which is ideal for preserving energy in undamped systems. If $\Omega^2 < 4$, the eigenvalues remain complex conjugates, ensuring stability. This agrees with Equation (16), confirming the stability boundary $\Omega \in [0, 2]$ for $\alpha = [2, 3/2]$.

Furthermore, Figure 4 clarifies the dispersion characteristics; while $\alpha = [2, 3/2]$ exhibits a slight period shrinkage, other configurations allow for controllable period elongation, providing the user with flexibility to tune the method according to the problem's requirements. For other parameter choices, such as $\alpha = [4/3, 3/2]$, the figures reveal that stability is only achieved in a limiting case, without a continuous stable range of $\Omega \in [0, \Omega_c]$.

3.3.2. Comparison with Other Explicit Methods

The performance of the EIG-3 method is benchmarked against several classical explicit integrators, including one-step methods (RK2, RK3), multistep methods (AB2, AB3), and the central difference (CD) method.

Figure 5 compares the spectral radii of these integrators. It is evident that higher-order methods like RK3 and AB3 possess significantly smaller stability regions. In contrast, EIG-3 and CD demonstrate superior stability boundaries, extending up to $\Omega \in [0, 2]$, whereas RK3 is limited to $\Omega \in [0, 1.732]$. This makes EIG-3 highly efficient for problems requiring larger time steps.

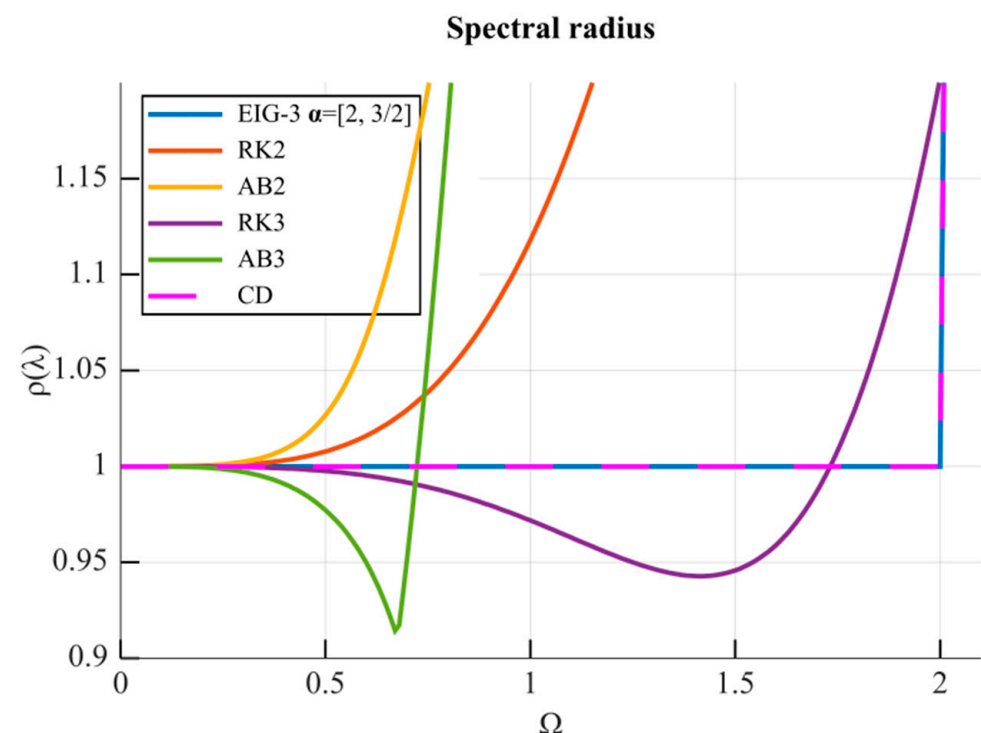


Figure 5. Comparison of spectral radius for different explicit integrator methods.

As shown in Figure 6, EIG-3 (with $\alpha = [2, 3/2]$) and CD are the only methods that introduce zero artificial damping across their entire stability region. Other methods, partic-

ularly RK2, AB2 and AB3, introduce numerical dissipation that can prematurely attenuate the high-frequency response of the system.

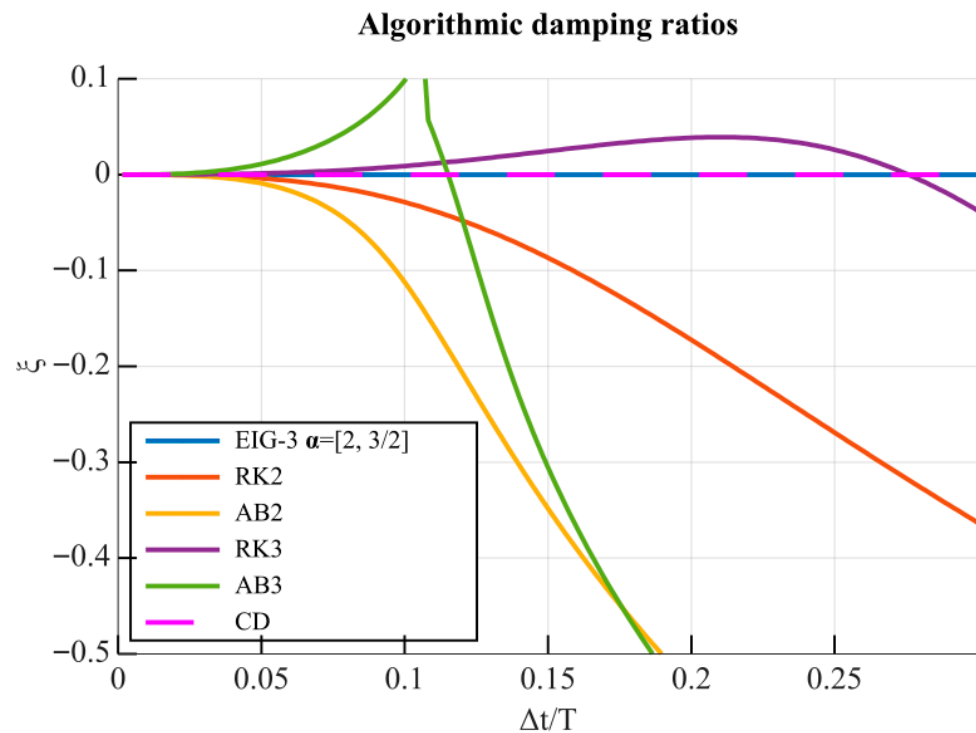


Figure 6. Comparison of algorithm damping ratio for different explicit integrator methods.

Figure 7 illustrates the relative period errors. EIG-3 and CD exhibit identical dispersion characteristics, with a slight period shrinkage as $\Delta t/T$ increases. While other methods like AB3 show period elongation, they do so at the cost of much stricter stability limits and higher dissipation.

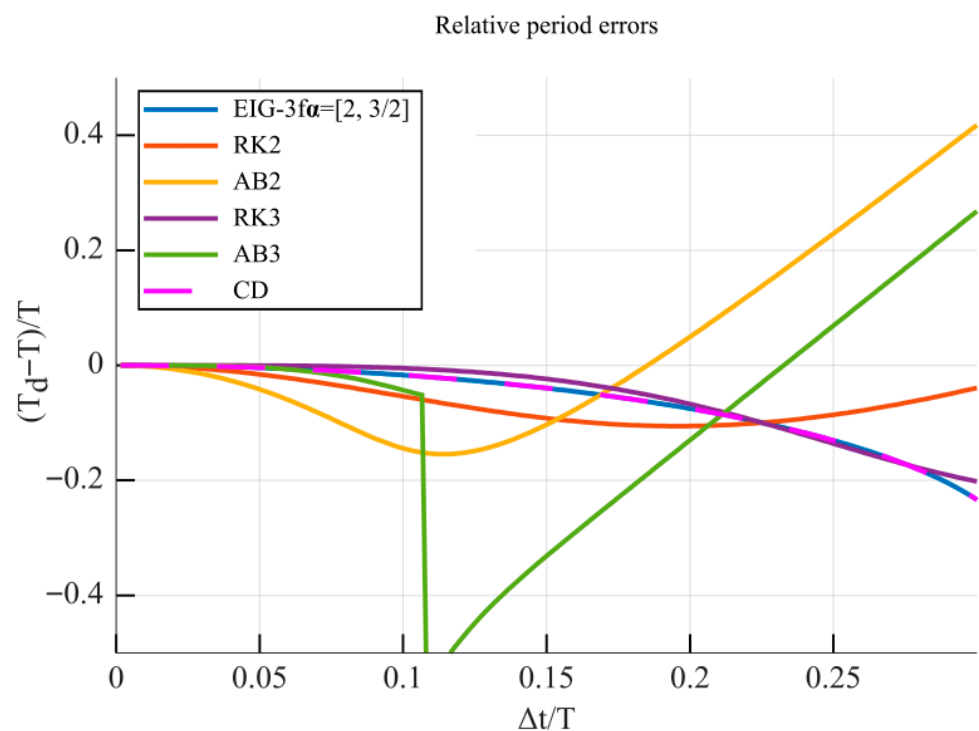


Figure 7. Comparison of relative period errors for different explicit integrator methods.

3.3.3. Key Advantages of EIG-3 over Central Difference (CD)

As shown in Figure 5 through Figure 7, the CD method and EIG-3 exhibit identical spectral radii, damping behavior, and dispersion characteristics when $\alpha = [2, 3/2]$. This equivalence stems from their similar amplification matrices, meaning they share the same vector space representation for linear systems. However, EIG-3 offers three crucial advantages over CD in practical engineering applications:

- If the ODE contains a nonlinear term dependent on the first derivative, $f(x, \dot{x}, t)$, the standard CD scheme requires an iterative solution to preserve second-order accuracy. Because the central difference approximation for $\dot{x}_n$ involves the unknown x_{n+1} , the value cannot be computed directly. EIG-3 remains strictly explicit for any second-order ODE, avoiding iterative procedures and significantly improving computational efficiency in nonlinear problems.
- While CD can be forced to be explicit by using a backward difference for the first derivative, this reduces the global convergence rate of that variable to $O(\Delta t)$. EIG-3 maintains second-order convergence for both the state variable and its derivative while remaining fully explicit.
- Standard CD is a rigid, non-dissipative scheme. EIG-3 can be “tuned” via its parameters to introduce controlled numerical dissipation. This is vital for stabilizing simulations or filtering high-frequency noise in complex dynamical systems where a purely non-dissipative approach like CD may lead to numerical instability.

4. Numerical Examples

In this chapter, four numerical examples of second-order differential equations are presented to validate the properties of the proposed explicit integration method (EIG-3). All the test cases involve mechanical system models. The results are compared with those obtained using the Runge–Kutta (RK2), Adams–Bashforth (AB2), and central difference (CD) methods. The examples include:

1. A single degree of freedom (DoF) system.
2. An elastic hardening spring system.
3. An elastic softening spring system.
4. A spring pendulum system.

In the case of the spring pendulum problem, where the central difference method encounters nonlinearity, Newton’s second-order method is applied for solution refinement. The accuracy of the simulations is evaluated using the analytical solution for the single DoF case and by checking the conservation of mechanical energy for the other examples. Computational efficiency is assessed by averaging the runtime over ten simulations.

All simulations were implemented in C/C++ and executed on an Intel Xeon E5345 at 2.33 GHz (Clovertown architecture) running Windows 10. The source code was compiled using the -O3 optimization flag. The algorithm was programmed with a single thread (just one core was used, without hyperthreading).

To evaluate the performance of the EIG-3 method, two complementary metrics are employed. In Section 4.1, where an analytical solution is available, the relative L^2 error norm is used to measure absolute numerical precision. For the subsequent nonlinear and high-dimensional cases (Sections 4.2–4.4), the relative mechanical energy drift is periodized. The ability of an integrator to maintain the system’s total energy is a fundamental indicator of its physical consistency and long-term numerical stability.

4.1. Single Degree of Freedom Model

The single degree of freedom (single DoF) model is widely used in structural dynamics to study vibration modes in discretized systems. This model consists of a spring, a point mass, and a damping element, representing different energy transfer mechanisms. The system is governed by the following equation:

$$x^{(2)} + 2\zeta\omega x^{(1)} + \omega^2 x = f, \quad (18)$$

where $\omega = \sqrt{k/m}$ is the natural frequency of the vibration mode, ζ is the damping rate and f is the external force. In this example, $x(0) = 1$, $x^{(1)}(0) = 0$, $\zeta = 0$, and $f = 0$. Furthermore, the natural frequency of the system will be $\omega = 4$ rad/s.

A schematic representation of this model is shown in Figure 8.

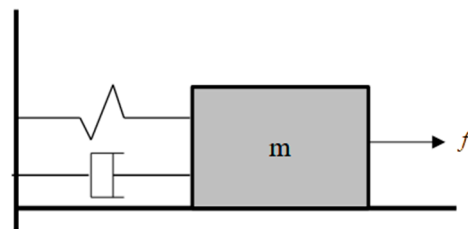


Figure 8. Single degree of freedom problem.

According to Equation (16), the maximum time step ensuring stability for the EIG-3 method with $\alpha = [2, 3/2]$ is:

$$\Delta t_{\max} = 2/\omega_{\max} = 2/4 = 0.5 \text{ s.}$$

To illustrate stability behavior, Figure 9 presents the solutions obtained using the EIG-3 method for four different time steps. The results confirm that the stability threshold is $\Delta t = 0.5$ s. Solutions with smaller time steps remain stable, while those exceeding this threshold become unstable.

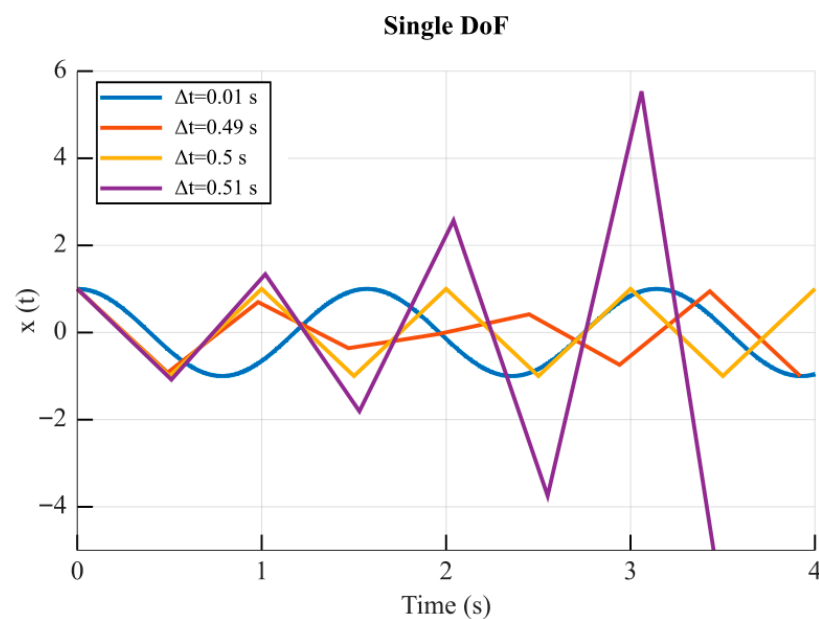


Figure 9. Single degree of freedom problem with EIG3 with $\Delta t = 0.01$, $\Delta t = 0.49$, $\Delta t = 0.5$ and $\Delta t = 0.51$ s.

The analytical solution of Equation (18) for the given conditions is:

$$x(t) = \cos(4t). \quad (19)$$

Comparison of Methods

The problem was solved using the EIG-3, CD, AB2, and RK2 methods for different time steps, with a simulation duration of 1000 s. The accuracy was assessed using the analytical solution. The results are summarized in Table 1, where “Time” refers to the computational time in seconds, and L^2 represents the relative error norm to the analytical solution.

Table 1. Results for single DoF with EIG-3, DC, AB2 and RK2.

Time Step		Methods						
Δt	EIG-3		CD		AB2		RK2	
	Time	L ²	Time	L ²	Time	L ²	Time	L ²
1×10^{-1}	0.0	1.99	0.0	1.99	0.0	7.32×10^{38}	0.0	6.9×10^{23}
1×10^{-2}	3.1×10^{-3}	0.26	3.1×10^{-3}	0.26	7.8×10^{-3}	2.0	0.015	1.03
1×10^{-3}	4.1×10^{-2}	2.6×10^{-3}	4×10^{-2}	2.6×10^{-3}	6.9×10^{-2}	2.6×10^{-2}	0.109	1.07×10^{-2}
1×10^{-4}	0.45	2.7×10^{-5}	0.39	2.7×10^{-5}	0.74	2.6×10^{-4}	1.15	1.1×10^{-4}

The results confirm that EIG-3 and CD produce identical accuracy and computational cost, which is expected given their similar stability characteristics. On the other hand, the AB2 and RK2 methods exhibit smaller stability regions. For example, when using $\Delta t = 0.1$ both methods become unstable when the algorithm becomes unstable. Although AB2 and RK2 are both second-order accurate methods, they perform less accurately than EIG-3 and CD at the same time step. The RK2 method incurs higher computational costs since it requires two function evaluations per time step.

4.2. Elastic Hardening Spring

This example is based on the work of Xie [33]. The equation governing the motion of a hardening elastic spring is given by:

$$x^{(2)} + 100(1 + 10x^2)x = 0,$$

with the initial conditions $x(0) = 1.5$ and $x^{(1)}(0) = 0$.

The period of oscillation for this system is $T = 0.15153$ s, and the simulation runs for $100T$ seconds. Since this is a conservative system, accuracy is evaluated based on the drift in mechanical energy (energy drift), calculated as:

$$E = 0.5(x^{(1)})^2 + 50x^2 + 500x^4.$$

Table 2 presents the accuracy and computational cost for the different integration schemes across various time step sizes.

Table 2. Results for elastic hardening spring with EIG-3, DC, AB2 and RK2.

Time Step		Methods							
Δt	EIG-3		CD		AB2		RK2		
	Time	Energy Drift	Time	Energy Drift	Time	Energy Drift	Time	Energy Drift	
1×10^{-3}	0.0	2.6	0.0	1.03	0.0	101.0	0.0	49.3	
1×10^{-4}	3.1×10^{-3}	2.6×10^{-2}	1.5×10^{-2}	1.03×10^{-2}	1.5×10^{-2}	9.3×10^{-2}	1.6×10^{-2}	5.7×10^{-2}	
1×10^{-5}	7.2×10^{-2}	2.6×10^{-4}	6.3×10^{-2}	1.03×10^{-4}	0.12	5×10^{-4}	0.19	1.4×10^{-4}	
1×10^{-6}	0.76	2.6×10^{-6}	0.61	7.7×10^{-6}	1.18	5.1×10^{-6}	1.9	1.05×10^{-6}	

The results indicate that all four methods provide similar accuracy when a sufficiently small time step is used. However, for $\Delta t = 1 \times 10^{-3}$ the AB2 and RK2 methods exhibit significantly larger errors compared to EIG-3 and CD. This is due to the fact that AB2 and RK2 become unstable at this step size. When the time step is reduced and all methods operate within their stability regions, their accuracy aligns with their expected order of convergence.

In terms of computational cost, EIG-3 and CD perform similarly, although CD is slightly more efficient. Since the given ODE is linear with respect to velocity, CD remains fully explicit in this case. AB2, despite only requiring one function evaluation per step, is slightly more expensive computationally. This effect is more pronounced for RK2, which requires two function evaluations per step, leading to a noticeable increase in cost, even though the function being evaluated is relatively simple.

4.3. Elastic Softening Spring

The elastic softening spring problem, also discussed in [33], is governed by the following differential equation:

$$x^{(2)} + 100 \tanh(x) = 0,$$

with the initial conditions $x(0) = 4$ and $x^{(1)}(0) = 0$. The period of this nonlinear oscillator is $T = 1.14$ s. The simulation is run for a total time of $100T$. As this system is conservative, the accuracy of the numerical methods is evaluated based on the drift in mechanical energy, which is computed as:

$$E = 0.5 \left(x^{(1)} \right)^2 + 100 \ln[\cosh(x)].$$

Due to the longer period of oscillation compared to the previous (hardening) example, the integration step sizes can be relatively larger. As shown in Table 3, all methods are stable for $\Delta t = 1 \times 10^{-3}$, but only EIG-3 and CD remain stable for $\Delta t = 1 \times 10^{-2}$ while AB2 and RK2 become unstable. These results are consistent with the stability boundaries illustrated earlier in Figures 5–7.

Table 3. Results for elastic softening spring with EIG-3, DC, AB2 and RK2.

Time Step		Methods							
Δt	EIG-3		CD		AB2		RK2		
	Time	Energy Drift	Time	Energy Drift	Time	Energy Drift	Time	Energy Drift	
1×10^{-2}	0.0	2.7	0.0	0.59	0.0	45.2	1.5×10^{-2}	20.4	
1×10^{-3}	1.7×10^{-2}	2.7×10^{-2}	1.6×10^{-2}	5.9×10^{-3}	1.9×10^{-2}	6.4×10^{-2}	3.8×10^{-2}	2×10^{-2}	
1×10^{-4}	0.16	2.7×10^{-4}	0.25	5.9×10^{-5}	0.23	2.8×10^{-4}	0.41	4.7×10^{-5}	
1×10^{-5}	1.57	2.7×10^{-6}	1.5	8.4×10^{-7}	2.29	2.4×10^{-6}	4.1	4.7×10^{-7}	

Although all methods display second-order convergence based on the evolution of the error, EIG-3 and CD consistently yield more accurate results for larger time steps. Interestingly, EIG-3 and CD also outperform AB2 and RK2 in precision at smaller time steps, even though the latter two methods are also nominally second order.

Regarding computational cost, this example demonstrates a higher function evaluation overhead compared to the elastic hardening case. This is partly due to the evaluation of the hyperbolic tangent function, which is computationally more expensive. Nevertheless, the system remains linear with respect to velocity, meaning CD retains its explicit nature and exhibits similar efficiency to EIG-3.

4.4. Spring Pendulum

The spring pendulum problem is a two degrees of freedom system composed of a massless rod, a linear spring, and a point mass, as illustrated in Figure 10. The system moves under the influence of gravity, with acceleration due to gravity taken as $g = 9.81$ m/s².

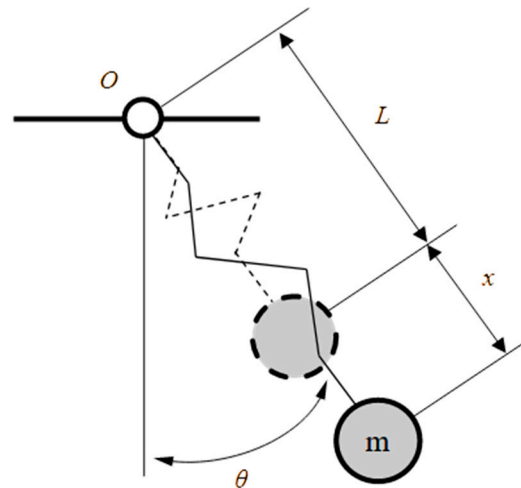


Figure 10. Spring pendulum problem.

Given a mass 1 kg, a rod length of 0.5 m and a spring stiffness 98.1 N/m, the governing equations of motion are:

$$x^{(2)} - (0.5 + x)(\theta^{(1)})^2 - 9.81 \cos \theta + 98.1x = 0,$$

$$\theta^{(2)} + \frac{(2x^{(1)}\theta^{(1)} + 9.81 \sin \theta)}{0.5 + x} = 0,$$

with initial conditions: $x(0) = 0.25$, $\theta(0) = \pi/2$ and $x^{(1)}(0) = \theta^{(1)}(0) = 0$. This problem, detailed in [28], is conservative, and energy is conserved over time. The system's total mechanical energy is calculated as:

$$E = 0.5(x^{(1)})^2 + 0.5(0.5 + x)^2(\theta^{(1)})^2 - 9.81(0.5 + x) \cos x + 0.5 \cdot 98.1x^2.$$

The simulation is run for 100 s. The performance of each method is summarized in Table 4, showing both accuracy (via energy drift) and computational time for different time steps.

Table 4. Results for spring pendulum with EIG-3, DC, AB2 and RK2.

Time Step		Methods							
Δt		EIG-3		CD		AB2		RK2	
	Time	Energy Drift	Time	Energy Drift	Time	Energy Drift	Time	Energy Drift	
1×10^{-2}	0.0	1.67	1.7×10^{-2}	3.9×10^{-2}	1.5×10^{-3}	3.4	1.5×10^{-3}	1.27	
1×10^{-3}	1.6×10^{-2}	2.5×10^{-3}	8.8×10^{-2}	3.9×10^{-4}	1.5×10^{-2}	4.2×10^{-3}	2.8×10^{-2}	1.1×10^{-3}	
1×10^{-4}	0.16	1.6×10^{-5}	0.89	3.9×10^{-6}	0.19	2.4×10^{-5}	0.29	2×10^{-6}	
1×10^{-5}	1.6	1.5×10^{-7}	8.9	8×10^{-8}	1.8	2.2×10^{-7}	2.9	1.1×10^{-8}	

The results show that RK2 and CD achieve greater accuracy than EIG-3 and AB2, particularly at smaller time steps. For EIG-3, the slightly lower precision could stem from the initial error propagation. Nevertheless, all methods show a consistent convergence rate compatible with their theoretical order.

In terms of computational cost, EIG-3 proves to be more efficient than CD in this example. Despite CD often being classified as an explicit method, here it behaves as implicit due to the nonlinear velocity terms. Consequently, an iterative solution is required.

To handle this, Newton's method was applied with the previous time step's values as an initial guess. Each iteration involved solving a linear system using LDU factorization with Rook's pivoting strategy, following [36]. This introduces a significantly higher

computational cost compared to EIG-3, which remains strictly explicit and does not require solving any system of equations.

A further comparison of EIG-3 with finer time steps is shown in Table 5.

Table 5. Results for spring pendulum with EIG-3 for different time steps.

Method	Time Step				
EIG-3	8×10^{-6}	6×10^{-6}	4×10^{-6}	2×10^{-6}	1×10^{-6}
Time	1.9	2.54	3.8	7.6	15.2
Energy drift	9.7×10^{-8}	5.4×10^{-8}	2.4×10^{-8}	6.1×10^{-9}	1.5×10^{-9}

As seen, a time step of 2×10^{-6} s achieves better accuracy than CD at a higher step (1×10^{-5}) but with less computational cost. This demonstrates the advantage of EIG-3 in handling nonlinear systems explicitly and efficiently.

5. Conclusions and Future Works

A new explicit integrator for second-order ordinary differential equations has been developed in this work. The proposed algorithm is based on a Taylor series expansion and allows for flexibility through the adjustment of a set of parameters. This parametric framework enables the custom tuning of stability and accuracy, offering a versatility typically associated with implicit structural integrators but within a strictly explicit context. Among the various configurations, a particular set of parameters has been identified that yields properties, both in terms of stability and convergence order, comparable to those of the central difference (CD) method. Notably, the new method only requires a single function evaluation per time step, making it simple to implement and computationally efficient.

To validate the theoretical foundations of the method, four numerical examples drawn from multibody dynamics have been solved. The performance of the new integrator (EIG-3) has been compared with three standard second-order explicit methods: Runge–Kutta 2 (RK2), Adams–Bashforth 2 (AB2), and central differences. In problems where the system is linear with respect to velocity, EIG-3 and CD exhibit similar computational costs, outperforming RK2 in efficiency, which is expected given RK2's requirement for two function evaluations per step. Furthermore, the single evaluation of EIG-3 facilitates variable time-stepping, as it avoids the overhead of intermediate stages in RK methods. In addition, EIG-3 demonstrated a more robust stability range compared to the AB2 multistep method, validating the advantage of using local higher-order derivatives instead of distant previous points.

Although all methods analyzed are formally second-order, differences in precision were observed, with CD and RK2 showing slightly better accuracy in some linear cases. However, in problems involving nonlinear terms in velocity, CD requires the use of iterative solvers, such as Newton's method, to maintain accuracy, which increases the computational cost. EIG-3, by contrast, remains strictly explicit even in the presence of such nonlinearities, thereby preserving its low computational overhead. In these scenarios, EIG-3 demonstrates superior performance compared to CD. When compared to RK2, EIG-3 typically completes the simulation in about half the time, depending on the complexity of the evaluated function.

Overall, EIG-3 emerges as a competitive and novel alternative to existing explicit integration methods for second-order ODEs. It offers a stability region comparable to that of the CD method while avoiding the need for iterative solvers in nonlinear problems. Additionally, within its stability limits, EIG-3 introduces no numerical damping, preserving the fidelity of the solution.

The Taylor-based derivation presented here serves as a modular foundation for a new family of integrators. Future work will focus on the development of higher-order versions of this integrator by incorporating higher-order derivative terms from the Taylor expansion. Further research will explore alternative parameter combinations to enhance both stability and convergence. Another promising direction is the extension of this approach to the solution of differential-algebraic equations (DAEs).

Author Contributions: Conceptualization, G.U., I.F.d.B., I.C. and H.U.; methodology, G.U., I.F.d.B., I.C. and H.U.; software, G.U. and I.F.d.B.; validation, G.U., I.F.d.B., I.C. and H.U.; formal analysis, G.U., I.F.d.B., I.C. and H.U.; investigation, G.U. and I.F.d.B.; resources, G.U.; data curation, I.F.d.B.; writing—original draft preparation, G.U., I.F.d.B., I.C. and H.U.; writing—review and editing, G.U. and I.F.d.B.; visualization, I.C. and H.U.; supervision, G.U. and I.F.d.B.; project administration, G.U.; funding acquisition, G.U., I.F.d.B., I.C. and H.U. All authors have read and agreed to the published version of the manuscript.

Funding: The authors thank the Basque Government for its financial support for this research work through the funding of the Research Group (IT1542-22 and IT1814-26). The authors are also grateful for the support of the project PID2021-124677NB-I00 funded by the Ministry of Science and Innovation through the state research agency MCIN/AEI/10.13039/501100011033 and by the “ERDF: A way of making Europe” program.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

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

References

- Butcher, J.C. *Numerical Methods for Ordinary Differential Equations*; John Wiley & Sons: Hoboken, NJ, USA, 2016.
- Lambert, J.D. *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 1991.
- Hairer, E.; Nørsett, S.P.; Wanner, G. *Solving Ordinary Differential Equations I*; Springer: Berlin/Heidelberg, Germany, 1993. [[CrossRef](#)]
- Hairer, E.; Wanner, G. *Solving Ordinary Differential Equations II*; Springer: Berlin/Heidelberg, Germany, 1991. [[CrossRef](#)]
- Bathe, K.-J. *Finite Element Procedures*; Klaus-Jürgen Bathe: Watertown, MA, USA, 2006.
- Zienkiewicz, O.C. *El Método de los Elementos Finitos*; Reverté: Barcelona, Spain, 1981.
- Hughes, T.J.R. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*; Courier Corporation: North Chelmsford, MA, USA, 2012.
- Gavrea, B.; Negrut, D.; Potra, F.A. The Newmark integration method for simulation of multibody systems: Analytical considerations. In *Proceedings of the ASME Design Engineering Division*; American Society of Mechanical Engineers: New York, NY, USA, 2005; pp. 1079–1092. [[CrossRef](#)]
- Negrut, D.; Rampalli, R.; Ottarsson, G.; Sajdak, A. On an implementation of the Hilber-Hughes-Taylor method in the context of index 3 differential-algebraic equations of multibody dynamics (DETC2005-85096). *J. Comput. Nonlinear Dyn.* **2007**, *2*, 73–85. [[CrossRef](#)]
- Hilber, H.M.; Hughes, T.J.R.; Taylor, R.L. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthq. Eng. Struct. Dyn.* **1977**, *5*, 283–292. [[CrossRef](#)]
- Chung, J.; Hulbert, G. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- α method. *J. Appl. Mech.* **1993**, *60*, 371–375. [[CrossRef](#)]
- de Bustos, I.F.; Uriarte, H.; Urkullu, G.; García-Marina, V. A non-damped stabilization algorithm for multibody dynamics. *Meccanica* **2022**, *57*, 371–399. [[CrossRef](#)]
- Cuadrado, J.; Dopico, D.; Naya, M.A.; Gonzalez, M. Penalty, semi-recursive and hybrid methods for MBS real-time dynamics in the context of structural integrators. *Multibody Syst. Dyn.* **2004**, *12*, 117–132. [[CrossRef](#)]
- González, F.; Kövecses, J. Use of penalty formulations in dynamic simulation and analysis of redundantly constrained multibody systems. *Multibody Syst. Dyn.* **2013**, *29*, 57–76. [[CrossRef](#)]
- Arnold, M.; Brüls, O. Convergence of the generalized- α scheme for constrained mechanical systems. *Multibody Syst. Dyn.* **2007**, *18*, 185–202. [[CrossRef](#)]

16. Belytschko, T.; Liu, W.K.; Moran, B.; Elkhodary, K. *Nonlinear Finite Elements for Continua and Structures*; John Wiley & Sons: Hoboken, NJ, USA, 2014.
17. Pfeiffer, F.; Glocker, C. *Multibody Dynamics with Unilateral Contacts*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2000.
18. Hallquist, J.O. *LS-DYNA Theory Manual*; Livermore Software Technology Corporation: Livermore, CA, USA, 2006; Volume 3, pp. 25–31.
19. Shabana, A.A. *Computational Dynamics*; John Wiley & Sons: Hoboken, NJ, USA, 2009.
20. Kasimu, A.; Zhou, W.; Yan, H.; Wang, Y.; Shen, C.; Zhang, Q. Evaluation of the vehicle-cargo and vehicle-trackside clearance of long and big railway freight vehicles. *Alex. Eng. J.* **2025**, *119*, 22–34. [[CrossRef](#)]
21. Ying, P.; Tang, H.; Chen, L.; Ren, Y.; Kumar, A. Dynamic modeling and vibration characteristics of multibody system in axial piston pump. *Alex. Eng. J.* **2023**, *62*, 523–540. [[CrossRef](#)]
22. Lee, K.C.; Alias, M.A.; Senu, N.; Ahmadian, A.; Lee, K.C.; Alias, M.A.; Senu, N.; Ahmadian, A. On efficient frequency-dependent parameters of explicit two-derivative improved Runge-Kutta-Nystrom method with application to two-body problem. *Alex. Eng. J.* **2023**, *72*, 605–620. [[CrossRef](#)]
23. Kim, W.; Lee, J.H. An improved explicit time integration method for linear and nonlinear structural dynamics. *Comput. Struct.* **2018**, *206*, 42–53. [[CrossRef](#)]
24. Urkullu, G.; de Bustos, I.F.; García-Marina, V.; Uriarte, H. Direct integration of the equations of multibody dynamics using central differences and linearization. *Mech. Mach. Theory* **2019**, *133*, 432–458. [[CrossRef](#)]
25. Soares, D., Jr. A novel family of explicit time marching techniques for structural dynamics and wave propagation models. *Comput. Methods Appl. Mech. Eng.* **2016**, *311*, 838–855. [[CrossRef](#)]
26. Kim, W. Higher-order explicit time integration methods for numerical analyses of structural dynamics. *Lat. Am. J. Solids Struct.* **2019**, *16*, e201. [[CrossRef](#)]
27. Kim, W. A simple explicit single step time integration algorithm for structural dynamics. *Int. J. Numer. Methods Eng.* **2019**, *119*, 383–403. [[CrossRef](#)]
28. Chung, J.; Lee, J.M. A new family of explicit time integration methods for linear and non-linear structural dynamics. *Int. J. Numer. Methods Eng.* **1994**, *37*, 3961–3976.
29. Nordsieck, A. On Numerical Integration of Ordinary Differential Equations. *Math. Comput.* **1962**, *16*, 22–49. Available online: <https://www.ams.org/mcom/1962-16-077/S0025-5718-1962-0136519-5/> (accessed on 21 May 2024). [[CrossRef](#)]
30. Helmborg, G.; Wagner, P.; Veltkamp, G. On Faddeev-Leverrier’s method for the computation of the characteristic polynomial of a matrix and of eigenvectors. *Linear Algebra Appl.* **1993**, *185*, 219–233.
31. Bär, C. The faddeev-leverrier algorithm and the Pfaffian. *Linear Algebra Appl.* **2021**, *630*, 39–55.
32. Erlicher, S.; Bonaventura, L.; Bursi, O.S. The analysis of the generalized- α method for non-linear dynamic problems. *Comput. Mech.* **2002**, *28*, 83–104. [[CrossRef](#)]
33. Xie, Y.M. An assessment of time integration schemes for non-linear dynamic equations. *J. Sound Vib.* **1996**, *192*, 321–331. [[CrossRef](#)]
34. Mohammadzadeh, S.; Ghassemieh, M.; Park, Y. Structure-dependent improved Wilson- θ method with higher order of accuracy and controllable amplitude decay. *Appl. Math. Model.* **2017**, *52*, 417–436. [[CrossRef](#)]
35. Hilber, H.M.; Hughes, T.J.R. Collocation, dissipation and [overshoot] for time integration schemes in structural dynamics. *Earthq. Eng. Struct. Dyn.* **1978**, *6*, 99–117. [[CrossRef](#)]
36. de Bustos, I.F.; García-Marina, V.; Urkullu, G.; Abasolo, M. An efficient LDU algorithm for the minimal least squares solution of linear systems. *J. Comput. Appl. Math.* **2018**, *344*, 346–355. [[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.