An Adaptive Step Implicit Midpoint Rule for the Time Integration of Newton's Linearisations of Non-Linear Problems with Applications in Micromagnetics

David Shepherd¹ · James Miles¹ · Matthias Heil² · Milan Mihajlović¹

Received: 20 June 2018 / Revised: 13 December 2018 / Accepted: 24 April 2019 / Published online: 21 May 2019
© The Author(s) 2019

Abstract

The implicit mid-point rule is a Runge–Kutta numerical integrator for the solution of initial value problems, which possesses important properties that are relevant in micromagnetic simulations based on the Landau–Lifshitz–Gilbert equation, because it conserves the magnetization length and accurately reproduces the energy balance (i.e. preserves the geometric properties of the solution). We present an adaptive step size version of the integrator by introducing a suitable local truncation error estimator in the context of a predictor-corrector scheme. We demonstrate on a number of relevant examples that the selected step sizes in the adaptive algorithm are comparable to the widely used adaptive second-order integrators, such as the backward differentiation formula (BDF2) and the trapezoidal rule. The proposed algorithm is suitable for a wider class of non-linear problems, which are linearised by Newton's method and require the preservation of geometric properties in the numerical solution.

Keywords Initial value problems · Runge–Kutta methods · Adaptive time integration · Predictor-corrector methods · Micromagnetics · Landau–Lifshitz–Gilbert equation

1 Background and Context

Initial value problems (IVP) arise in mathematical models of many important physical and engineering processes and phenomena. They either appear as stand-alone problems (as ordinary differential equations (ODEs), where the unknown functions depend only on a single variable, for example describing the motion in classical mechanics or chemical reactions), or in the context of applying the method of lines to partial differential equations (PDE), i.e. the problems where unknown functions have both spatial and time variation [1, p. 8]. The solutions of such problems frequently exhibit multiple spatio-temporal scales. In such cases implicit solvers allow the deployment of larger step sizes, while maintaining stability. Efficient integrators should also involve adaptivity [2, p. 73,74], which can be realised both

✉ Milan Mihajlović
milan@cs.man.ac.uk

¹ School of Computer Science, The University of Manchester, Manchester, UK

² School of Mathematics, The University of Manchester, Manchester, UK

with explicit and implicit methods. The design and development of these algorithms has been the topic of much previous research (see, for example [3, Sect. 3.16], [4–7]). Many publicly available codes for the solution of IVPs (for example CVODE [8], or the various solvers in Matlab [9]) involve some adaptivity strategy. Adaptive IVP solvers allow the selection of the right step sizes at each stage of the solution process to satisfy the required accuracy, thus removing the need for a “trial-and-error” approach. This also improves the computational efficiency and robustness of the overall algorithm.

In some problems the governing equations dictate that the solution vectors should have constant magnitude, and that the system’s energy should either be conserved or accurately reproduced. In such cases IVP solvers should preserve these properties at the discrete level, while computational efficiency is achieved by using the adaptive implicit methods. As an example, we consider micromagnetics [10]. Dynamical micromagnetic models are based on the Landau–Lifshitz–Gilbert (LLG) equation, a time-dependent differential equation modelling the magnetization evolution in a ferromagnetic body. The classical LLG equation models the magnetization at zero temperature, which implies that the magnetization vector length is constant at any point of the domain. The second important property of the LLG equation is the energy conservation in the non-damped case, or equivalently, under a constant applied field the energy decreases at a rate proportional to the damping factor. Solutions of the LLG equation in practical cases often exhibit multiple spatio-temporal scales favouring the use of adaptive implicit time integration schemes [11]. An integration scheme that preserves (or, in the damped case, accurately reproduces) the length/energy properties of the solution is referred to as a *geometric integrator*. Most commonly used implicit schemes (such as the second-order backward differentiation formula BDF2 or the trapezoidal rule (TR)) do not possess this property. By contrast, the implicit mid-point rule (IMR), combined with an appropriate spatial discretisation in the PDE cases, is a geometric integrator [12, 13]. The proposed adaptive version of IMR retains this favourable property. Some alternative numerical techniques aimed at preserving the magnetization length include periodic renormalisation [14], length constraints imposed through Lagrange multipliers [15], penalty formulations [16], and the self-correcting LLG equation used by Nmag [16]. However, all these formulations change the energy of the system.

The implicit mid-point rule (IMR) is a well-known second order implicit Runge–Kutta method [17, p. 204], [3, p. 262], that has been applied in fixed-step form to micromagnetic problems [12]. In this paper we present an adaptive version of the IMR based on the new local truncation error estimator in the context of a predictor–corrector scheme. We demonstrate the efficiency of the adaptive version of IMR by comparing its accuracy and computational cost with the fixed step version of the method [12], and with the adaptive versions of the BDF2 [3, p. 649] and the TR [3, p. 647] when applied to both ODE and PDE problems that arise in computational micromagnetics.

The paper is organised as follows. In Sect. 2 we introduce the LLG equation and describe a method for its discretisation that is used in this paper. Section 3 introduces the IMR and details some of its properties. In Sect. 4 we present the new local truncation error (LTE) estimator for the IMR and introduce the adaptive integrator. Finally, in Sect. 5 we evaluate the effectiveness of the adaptive IMR when applied to ODE and PDE problems associated with the LLG equation. We also compare the adaptive IMR to adaptive BDF2 and the standard version of TR in terms of the computational cost, accuracy and the preservation of geometric properties.

2 The Dynamical Micromagnetic Model

2.1 The Landau–Lifshitz–Gilbert Equation

Micromagnetics is a semi-continuum theory of ferromagnetic materials that covers the length-scales between the atomistic spin dynamics and the classical Maxwell's theory. Its aim is to describe the behaviour of a magnetic body, expressed by its magnetization vector $\mathbf{m}$, under various external conditions and internal material properties. This can be achieved either by calculating the minimum energy of a magnetic body [10], or by modelling the magnetization dynamics. In the latter case the time evolution of the magnetization is modelled by a differential equation. Given a space of k -continuous functions $C^k(\cdot)$, the evolution of the magnetization vector denoted by $\mathbf{m} \in [C^2(\Omega)]^3 \times C^1(T)$ in a magnetic body $\Omega \subset \mathbb{R}^d$, $d = 2, 3$ with a smooth boundary $\partial\Omega$ over a time interval T follows the Landau-Lifshitz (LL) equation, which, in non-dimensional form, reads

$$\frac{\partial \mathbf{m}}{\partial t} = -\frac{1}{1+\alpha^2} (\mathbf{m} \times \mathbf{h}) - \frac{\alpha}{1+\alpha^2} \mathbf{m} \times (\mathbf{m} \times \mathbf{h}) \quad \text{in } \Omega \subset \mathbb{R}^d, \quad (1)$$

$d = 2, 3$, subject to the Neumann-type boundary conditions (BCs) [10, p. 178]

$$\mathbf{m} \times \frac{\partial \mathbf{m}}{\partial \hat{\mathbf{n}}} = 0 \quad \text{on } \partial\Omega, \quad (2)$$

and the initial condition

$$\mathbf{m}(t=0) = \mathbf{m}_0 \quad \text{in } \bar{\Omega} = \Omega \cup \partial\Omega. \quad (3)$$

In (1) $\mathbf{h} \in [C(\Omega)]^3 \times C(T)$ is the effective magnetic field, and α is an empirical damping parameter. In (2) $\hat{\mathbf{n}}$ denotes the outward normal vector on $\partial\Omega$. The first term on the right-hand side of (1) determines the precession of $\mathbf{m}$, while the second term is responsible for damping. A mathematically equivalent form of (1), referred to as the Landau–Lifshitz–Gilbert (LLG) equation is frequently used, which in non-dimensional form reads

$$\frac{\partial \mathbf{m}}{\partial t} = -\mathbf{m} \times \left(\mathbf{h} - \alpha \frac{\partial \mathbf{m}}{\partial t} \right) \quad \text{in } \Omega, \quad (4)$$

The effective field $\mathbf{h}$ in (1) and (4) has several components [18, p. 306], namely

$$\mathbf{h} = \mathbf{h}_{\text{ap}} + \mathbf{h}_{\text{ex}} + \mathbf{h}_{\text{ms}} + \mathbf{h}_{\text{mc}}, \quad (5)$$

where $\mathbf{h}_{\text{ap}}$ is the applied (external) field, $\mathbf{h}_{\text{ex}} = \nabla^2 \mathbf{m}$ is the is a phenomenological exchange field that represents the effect of quantum mechanical exchange, which acts to align the magnetic moments of neighbouring atoms in ferromagnetic materials, $\mathbf{h}_{\text{ms}} = \nabla \phi$ is the magnetostatic (demagnetisation or stray) field expressed in terms of the magnetostatic potential ϕ , and $\mathbf{h}_{\text{mc}} = k_1 (\mathbf{m} \cdot \hat{\mathbf{e}}) \hat{\mathbf{e}}$ is the magnetocrystalline anisotropy field associated with the easy magnetization axis $\hat{\mathbf{e}}$ of a ferromagnetic material [18, p. 306]. The magnetostatic potential $\phi(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^3$ is related to the magnetization vector through the equation

$$\nabla^2 \phi = \nabla \cdot \mathbf{m} \quad \text{in } \Omega^* = \mathbb{R}^3 \setminus \Omega, \quad (6)$$

where Ω^* is the exterior of a ferromagnetic body Ω . Boundary conditions associated with (6) are of the form

$$\phi^{\text{int}} - \phi^{\text{ext}} = 0, \quad (7)$$

$$\frac{\partial \phi^{\text{int}}}{\partial \hat{\mathbf{n}}} - \frac{\partial \phi^{\text{ext}}}{\partial \hat{\mathbf{n}}} = \mathbf{m} \cdot \hat{\mathbf{n}}, \quad (8)$$

on $\partial\Omega$, where ϕ^{int} and ϕ^{ext} are the values of the potential just inside and outside of Ω , respectively, and

$$\phi(\mathbf{x}) \rightarrow 0 \quad \text{for } |\mathbf{x}| \rightarrow \infty. \quad (9)$$

The fact that (6) is posed over an infinite domain, together with the BC (9) at infinity cause difficulties with the discretisation of this part of the problem.

2.2 Finite Element Discretisation

The weak formulation of (4) in a residual form is to find $\mathbf{m} \in [H^1(\Omega)]^3$ such that $\mathbf{r}(\mathbf{m}, \mathbf{v}) = \mathbf{0}$ for all functions $\mathbf{v} \in [H^1(\Omega)]^3$, i.e.

$$\mathbf{r}(\mathbf{m}, \mathbf{v}) = \int_{\Omega} \left[\frac{\partial \mathbf{m}}{\partial t} + (\mathbf{m} \times \mathbf{h}) - \alpha \left(\mathbf{m} \times \frac{\partial \mathbf{m}}{\partial t} \right) \right] \mathbf{v} \, d\Omega = 0, \quad (10)$$

where $H^1(\Omega)$ is the standard Sobolev space. The discrete problem is obtained by restricting this space to a finite element space S_h associated with the subdivision of Ω into a set of quadrilateral elements and adopting piecewise bi/trilinear polynomial approximation $\mathbf{m}_h$ for the solution $\mathbf{m}$ over the elements, i.e. $S_h = \text{span}\{\psi_k\}_{k=1}^N$, $N = N_i + N_b$ (N_i is the number of nodes in the interior of Ω and N_b the number of nodes on $\partial\Omega$). Then we have

$$\mathbf{m} \approx \mathbf{m}_h = \sum_{k=0}^N \mathbf{m}_k \psi_k, \quad (11)$$

with $\mathbf{m}_k = [m_{k,x} \ m_{k,y} \ m_{k,z}]^T$ representing the values of $\mathbf{m}$ at node k . This process results in a semi-discrete problem, which is fully discretised approximating the time derivatives in (10). We use the IMR method in this context (see Sect. 3). The linearisation of the non-linear discrete problem at each time step is done using the Newton method, leading to the solution of a sequence of linear algebraic systems of the form $J_\ell \delta \bar{\mathbf{m}}_\ell = -\bar{\mathbf{r}}_\ell$, $\bar{\mathbf{m}}_{\ell+1} = \bar{\mathbf{m}}_\ell + \delta \bar{\mathbf{m}}_\ell$, $\ell = 0, 1, \dots$. The Jacobian matrix with elements $J_{mk} = \frac{\partial r_m}{\partial m_k}$ can be written in block form

$$J = \begin{bmatrix} c\tilde{M} & -K_z & K_y \\ K_z & c\tilde{M} & -K_x \\ -K_y & K_x & c\tilde{M} \end{bmatrix}, \quad (12)$$

where $\tilde{M}$ is a matrix similar to a mass matrix [19, p. 55], while the blocks K_* are sums of mass and Laplacian-like matrices (for detailed expressions see [13, pp. 97–102]).

The Jacobian matrices are assembled from the elemental contributions with their coefficients calculated two ways: either using standard Gaussian quadratures [19, p. 30], or nodal (Newton–Cotes) quadratures [20]. The former approach is standard in FE computations and gives optimal order of accuracy per amount of computational work. However, it does not lead to the conservation of the magnetization length.¹ The nodal quadrature (also known in micromagnetic literature as the “reduced integration” [21]) for an integral over an element Ω_e is defined by

$$\int_{\Omega_e} \mathbf{f}(\mathbf{x}) d\mathbf{x} \approx \sum_{\ell \in V(\Omega_e)} \beta_\ell \mathbf{f}(\mathbf{x}_\ell), \quad \beta_\ell = \int_{\Omega_e} \varphi_\ell^{(e)} d\mathbf{x}, \quad (13)$$

where $\varphi_\ell^{(e)}$ is the local basis function associated with the local node ℓ and $V(\Omega_e)$ is the set of nodes of an element Ω_e . The nodal quadratures use element’s nodes as the knots, thus

¹ In this case the magnetization is conserved in an “element-wise”, rather than “point-wise” fashion.

allowing the point-wise magnetization length conservation when the weak formulation of the problem is used as the basis for spatial discretisation [20]. In order for this procedure to be effective, we need to set a tight tolerance ϵ_N in the stopping criterion of Newton's method (e.g. $\epsilon_N < 10^{-10}$).

Notice that J in (12) is a block skew-symmetric matrix. Linear systems with the coefficient matrix J are solved either directly, or by preconditioned Krylov solvers. In our experiments reported in Sects. 5.2 and 5.3 GMRES preconditioned by ILU(1) method is used [22].

2.3 Hybrid Finite-Boundary Element Method for the Exchange Field

Spatial discretisation of (6) with BCS (7)–(9) is problematic, as the domain Ω^* is infinite. The application of FEM in this context would require so-called “infinite elements” [23]. An alternative is to truncate the domain Ω^* or to use asymptotic BCs [24]. The asymptotic accuracy of such approaches is usually low. An alternative approach is to apply the hybrid finite/boundary element method, in which the external domain is replaced by a dipole layer placed on $\partial\Omega$, which simulates the correct BC (9) [25]. Standard FEM is used to calculate the potentials in Ω .

We use the linearity of the problem (6)–(9) to introduce two new potentials $u(\mathbf{x})$ and $v(\mathbf{x})$, such that $\phi = u + v$, and u is defined to be non-zero only in Ω . Therefore, $u(\mathbf{x})$ satisfies

$$\nabla^2 u = \nabla \cdot \mathbf{m} \quad \text{in } \Omega, \quad (14)$$

$$\frac{\partial u^{\text{int}}}{\partial \hat{\mathbf{n}}} = \mathbf{m} \cdot \hat{\mathbf{n}} \quad \text{on } \partial\Omega, \quad (15)$$

$$u = 0 \quad \text{in } \Omega^*. \quad (16)$$

Thus, u is calculated in the domain Ω using the FEM and the double layer charge distribution on $\partial\Omega$ simulates the potential v in Ω^* and is solved by BEM. After setting the conditions for v [13, p. 114–116], we obtain the following problem:

$$\nabla^2 \phi = \nabla \cdot \mathbf{m}, \quad \nabla^2 u = \nabla \cdot \mathbf{m} \quad \text{in } \Omega, \quad (17)$$

subject to the BCs

$$\phi(\mathbf{x}) = \mathcal{G}[u(\mathbf{x})], \quad (18)$$

$$\frac{\partial u}{\partial \hat{\mathbf{n}}} = \mathbf{m} \cdot \hat{\mathbf{n}} \quad \text{on } \partial\Omega, \quad (19)$$

where $\mathcal{G}[\cdot]$ is defined as

$$\mathcal{G}[u] = - \int_{\partial\Omega} u(\mathbf{x}) \cdot \frac{\partial \mathbb{G}(\mathbf{x}, \mathbf{y})}{\partial \hat{\mathbf{n}}(\mathbf{y})} d\mathbf{y} + \gamma(\mathbf{x})u(\mathbf{x}). \quad (20)$$

In (20) $\mathbb{G}(\mathbf{x}, \mathbf{y})$ denotes the Green's function of the Laplacian, and $\gamma(\mathbf{x})$ is the fractional (space) angle, where for any smooth part of $\partial\Omega$ we have $\gamma = \frac{1}{2}$. After discretisation, Eq. (18) becomes a matrix-vector product $\bar{\phi} = G\bar{u}$, where the elements of the BEM matrix G are obtained as

$$G_{ij} = - \frac{1}{2^{d-1}\pi} \int_{\partial\Omega_j} \psi_j(\mathbf{y}) \frac{\hat{\mathbf{n}}(\mathbf{y})(\mathbf{y} - \mathbf{x}_i)}{|\mathbf{y} - \mathbf{x}_i|^d} d\mathbf{y} + \gamma(\mathbf{x}_i)\delta_{ji}, \quad j, i = 1, \dots, N_b. \quad (21)$$

In (21) ψ_j is the basis function, $d = 2, 3$ is the spatial dimension, and δ_{ji} is the Kronecker delta.

The accuracy of the FEM/BEM method is better than that of the asymptotic BCs method [26]. The main drawback of the hybrid method is that it produces a dense coefficient matrix of size N_b , leading to both the computational and storage cost of $O(N_b^2)$ (in the discrete version of (18) the BEM matrix needs to be multiplied by a vector to impose the BCs). This cost can be prohibitive when modelling micromagnetic problems posed over thin domains, where $N_b = O(N)$. This problem was circumvented by the approximation of the BEM matrix by low rank hierarchical matrices, as implemented in the library `HLiB` [27,28], reducing both the storage and computational cost from $O(N_b^2)$ to $O(N_b \log N_b)$.

3 The Implicit Mid-Point Rule

Consider a system of N coupled IVPs

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}(t)), \quad \mathbf{y}(0) = \mathbf{y}_0 \quad (22)$$

for $t \in T = [0, t_{\max}]$, where $\mathbf{y}(t) \in [C^1(T)]^N$ is a vector function, $\mathbf{y}_0$ is the initial condition, and $\mathbf{f}$ is a locally Lipschitz-continuous function. We denote an approximation to $\mathbf{y}(t_n)$ by $\mathbf{y}_n$ and write $\Delta_n = t_{n+1} - t_n$ for the size of n -th time step, and introduce three commonly used second-order implicit methods for numerical integration of (22).

The IMR is defined by [17, p. 204]

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \Delta_n \mathbf{f}\left(\frac{t_{n+1} + t_n}{2}, \frac{\mathbf{y}_n + \mathbf{y}_{n+1}}{2}\right). \quad (23)$$

With the notation $t_{n+1/2} = \frac{t_{n+1} + t_n}{2} = t_n + \frac{\Delta_n}{2}$, $\mathbf{y}_{n+1/2} = \frac{\mathbf{y}_{n+1} + \mathbf{y}_n}{2}$ for the mid-point approximations of t and $\mathbf{y}$, (23) becomes

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \Delta_n \mathbf{f}(t_{n+1/2}, \mathbf{y}_{n+1/2}). \quad (24)$$

The trapezoidal rule (TR) [3, p. 260] is given by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{\Delta_n}{2} (\mathbf{f}(t_{n+1}, \mathbf{y}_{n+1}) + \mathbf{f}(t_n, \mathbf{y}_n)), \quad (25)$$

and the second order backward difference formula (BDF2) [3, p. 715] by

$$\mathbf{y}_{n+1} - \frac{(1 + w_n)^2}{1 + 2w_n} \mathbf{y}_n + \frac{w_n^2}{1 + 2w_n} \mathbf{y}_{n-1} = \Delta_{n+1} \frac{1 + w_n}{1 + 2w_n} \mathbf{f}(t_{n+1}, \mathbf{y}_{n+1}), \quad (26)$$

where $w_n = \frac{\Delta_{n+1}}{\Delta_n}$, $\Delta_{n+1} = t_{n+1} - t_n$, $\Delta_n = t_n - t_{n-1}$.

3.1 Local Truncation Errors and the Order of Convergence

The local truncation error (LTE) of an integration scheme is the error due to a single integration step [1, p. 40]:

$$\mathbf{T}_n = \mathbf{y}(t_{n+1}) - \hat{\mathbf{y}}_{n+1}, \quad (27)$$

where $\hat{\mathbf{y}}_{n+1}$ is an approximation to $\mathbf{y}(t_{n+1})$ obtained by the integration method with all the history values $\mathbf{y}_n, \mathbf{y}_{n-1}, \dots$ being exact (i.e. $\mathbf{y}_{n-i} = \mathbf{y}(t_{n-i})$). The LTE expression for the IMR is more complex than those of TR or BDF2 due to the approximation $\mathbf{y} \approx \frac{\mathbf{y}_n + \mathbf{y}_{n+1}}{2}$. To derive it, we apply Taylor expansions of $\mathbf{y}(t_{n+1})$ and $\mathbf{y}(t_n)$ about the midpoint $t_{n+1/2}$:

$$\mathbf{y}(t_{n+1}) = \mathbf{y}(t_{n+1/2}) + \frac{\Delta_n}{2} \mathbf{y}'(t_{n+1/2}) + \frac{\Delta_n^2}{8} \mathbf{y}''(t_{n+1/2}) + \frac{\Delta_n^3}{48} \mathbf{y}'''(t_{n+1/2}) + O(\Delta_n^4), \quad (28)$$

and

$$\mathbf{y}(t_n) = \mathbf{y}(t_{n+1/2}) - \frac{\Delta_n}{2} \mathbf{y}'(t_{n+1/2}) + \frac{\Delta_n^2}{8} \mathbf{y}''(t_{n+1/2}) - \frac{\Delta_n^3}{48} \mathbf{y}'''(t_{n+1/2}) + O(\Delta_n^4), \quad (29)$$

Substituting (28) and (29) into

$$\mathbf{T}_n^{\text{IMR}} = \mathbf{y}(t_{n+1}) - \mathbf{y}(t_n) - \Delta_n \mathbf{f} \left(t_{n+1/2}, \frac{\mathbf{y}(t_n) + \mathbf{y}_{n+1}^{\text{IMR}}}{2} \right) \quad (30)$$

gives

$$\mathbf{T}_n^{\text{IMR}} = \underbrace{\frac{\Delta_n^3}{24} \mathbf{y}'''(t_{n+1/2})}_I + \Delta_n \underbrace{\left[\mathbf{y}'(t_{n+1/2}) - \mathbf{f} \left(t_{n+1/2}, \frac{\mathbf{y}(t_n) + \mathbf{y}_{n+1}^{\text{IMR}}}{2} \right) \right]}_{\text{II}} + O(\Delta_n^4). \quad (31)$$

The LTE expression (31) has two terms: terms similar to (I) feature in LTEs of many other second-order integrators [cf. (32) and (33)], while the term (II) arises from the use of the approximation to $\mathbf{y}(t_{n+1/2})$ in the evaluation of $\mathbf{f}$ (i.e. it reflects the Runge–Kutta nature of the IMR). The IMR is asymptotically second-order accurate (i.e. the global error behaves as $O(\Delta_n^2)$ when $\Delta_n \rightarrow 0$) [29, p. 152]. For completeness, we present the LTE expressions for the TR [3, p. 261] and the BDF2 methods [3, p. 715], respectively:

$$\mathbf{T}_n^{\text{TR}} = -\frac{\Delta_n^3 \mathbf{y}'''_n}{12} + O(\Delta_n^4), \quad (32)$$

and

$$\mathbf{T}_n^{\text{BDF2}} = \frac{(\Delta_n + \Delta_{n-1})^2}{\Delta_n(2\Delta_n + \Delta_{n-1})} \frac{\Delta_n^3 \mathbf{y}'''_n}{6} + O(\Delta_n^4). \quad (33)$$

3.2 Properties of the IMR

The IMR has two important properties that makes it a method of choice for the time integration of semi-discretised PDEs:

- It is A-stable [17, p. 43,251], i.e. it can be applied to stiff problems.
- It requires only one evaluation of the function $\mathbf{f}$ per step (a one-step method, unlike many other Runge–Kutta methods).

These properties of the IMR method are shared with TR and BDF2. However, the IMR does not introduce spurious damping of oscillatory solutions (TR does possess this property, although it is lost in the modified versions of it [4,30], introduced to avoid the locking effect in the step size selection caused by the accumulation of the round-off errors). All backward differentiation formulae are damping [3, p. 265]. When applied to the time integration of the LLG equation, the IMR has geometric properties, related to the conservation of the magnetization length and reproduces a correct energy behaviour of the system. It is also worth noticing that when applied to the integration of linear problems, IMR and standard TR will produce exactly the same numerical solutions, i.e. the two methods are equivalent (by virtue of $\mathbf{f}_{n+1/2} = \frac{1}{2}(\mathbf{f}_{n+1} + \mathbf{f}_n)$ for linear functions $\mathbf{f}$ [3, p. 650]).

4 Adaptive Implicit Mid-Point Rule

Implicit time integration of ODEs or semi-discretised PDEs is a computationally intensive task. Numerically effective integrators employ adaptivity, which enables an integrator to take steps of the size governed, within the prescribed tolerance, by the physics of the problem, rather than artificial (method-specific) constraints. Adaptive step selection algorithms rely on computable estimates of the LTE. In this section we discuss the difficulties in applying the commonly used LTE estimation schemes to IMR and introduce the new approach.

4.1 Local Truncation Error Estimation

Typical LTE estimators used in implicit IVP integrators (such as TR or BDF2) are based on the solution estimate $\mathbf{y}_{n+1}^E \sim \mathbf{y}(t_{n+1})$ computed by an explicit method with the same asymptotic order of accuracy, using the history values computed by an implicit method. This is referred to as *the predictor step*. Algebraic rearrangements of the analytical expressions for the LTEs of an explicit and an implicit method lead to a computable estimate of the implicit LTE with a desired order of accuracy. This approach is known as *Milne's device* [3, p. 707–716]. It is not straightforward to apply this approach to estimating the LTE of the IMR, due to the presence of the term II in (31). The difficulty of constructing a Milne device in this case lies in finding a suitable predictor which LTE involves the term II from (31). In the case of linear or linearised problems (such as the Simo-Armero linearisation of the Navier–Stokes equations [31]), the IMR is numerically equivalent to TR, which implies that the use of the standard AB2 predictor will produce the same behaviour (in terms of the number and the size of the selected steps) for both AB2-TR and AB2-IMR integrators. This is no longer the case when $\mathbf{f}$ is non-linear and we focus on this scenario in the remainder of this paper.

Adaptivity in explicit Runge–Kutta methods is achieved by finding a pair of methods of different orders and to obtain a LTE estimate by comparing the two solutions [17, p. 165]. The effectiveness of this approach relies on the existence of pairs of methods that share the most of their function evaluation points – the so-called embedded methods (the examples include the Dormand–Prince pair of order 4/5 deployed in Matlab's function `ode45` [9]). To deploy this approach in the context of the IMR would require a third order implicit method that involve a function evaluation in addition to $\mathbf{f}(t_{n+1/2}, \mathbf{y}_{n+1/2})$. However, the function evaluation $\mathbf{f}(t_{n+1/2}, \mathbf{y}_{n+1/2})$ at the “mid-point” of the interval $[t_n, t_{n+1}]$ leads to the cancellation of the second-order terms in (28)–(29), and an additional function evaluation at a point in $[t_n, t_{n+1}]$ would break this symmetry. To restore it, we would require two additional function evaluations.

In order to bypass these problems, we consider explicit third order methods as predictors in the non-linear case. Suitable candidates are AB3 [1, p. 127], RK3 [1, p. 73], and the eBDF3 method [17, p. 378]. The first two of these methods are computationally more expensive and/or require the storage of more history values than the AB2 method. This is, however, not the case with the eBDF3 method. In order to be a competitive predictor, it should lead either to the overall reduction in the number of time steps in non-linear cases for a set LTE tolerance, or a better accuracy for a fixed number of steps performed compared to the AB2 predictor. In particular, the AB2 method requires the storage of the function values (the solution derivatives) at two history points. In the non-linear case this amounts to either storing the Jacobians at these points or assembling them on the fly. We notice that the function evaluations at multiple points also introduce additional implementation complications in

ODE/PDE cases where the time derivative is given implicitly (such as in the LLG equation). The situation is even worse for AB3 and RK3 methods that require the computation or storage of the function values at three history points. Neither of these predictors is self-starting, which is a drawback for a self-starting IMR method. The eBDF3 method is not self-starting either, but requires the storage of three history *solution* values, rather than its derivatives, and only one function evaluation (at Δ_n). This makes it a competitive alternative to the AB2 method and the other two third order methods. The computational cost of applying the eBDF3 predictor consists of one sparse matrix-vector multiplication (where we assume that a dense matrix block obtained in the BEM discretisation of the magnetostatic field is sparsified prior to its use) and, in PDE cases, the solution of one trivial linear system with a diagonal mass matrix (assuming that reduced quadratures are used). The eBDF3 predictor is therefore judiciously chosen among the other alternatives (AB2, AB3, RK3) due to its low computational and/or storage overhead.

The eBDF3 method is unstable for fixed step sizes [17, p. 378], however this is not an issue when used as a predictor, as we only perform a single step of it using IMR-computed history values. Thus, the eBDF3 method can be viewed as an extrapolation of $\mathbf{y}$ at t_{n+1} using the history values.

The LTE estimate of a generic explicit third-order method at the time step $\Delta_n = t_{n+1} - t_n$ is given by:

$$\mathbf{T}_n^{\text{em3}} = \mathbf{y}(t_{n+1}) - \mathbf{y}_{n+1}^{\text{em3}} = C_{\text{em3}} \Delta_n^4 + O(\Delta_n^5), \quad (34)$$

and the LTE of the IMR can be written as

$$\mathbf{T}_n^{\text{IMR}} = \mathbf{y}(t_{n+1}) - \mathbf{y}_{n+1}^{\text{IMR}} = C_{\text{IMR}} \Delta_n^3 + O(\Delta_n^4). \quad (35)$$

Expressing $\mathbf{y}(t_{n+1})$ from (34) and substituting to (35) gives

$$\mathbf{y}_{n+1}^{\text{em3}} - \mathbf{y}_{n+1}^{\text{IMR}} = C_{\text{IMR}} \Delta_n^3 - C_{\text{em3}} \Delta_n^4 + O(\Delta_n^5) = C_{\text{IMR}} \Delta_n^3 + O(\Delta_n^4) \approx \mathbf{T}_n^{\text{IMR}}, \quad (36)$$

i.e. we have an estimate:

$$\mathbf{T}_n^{\text{IMR}} = \mathbf{y}_{n+1}^{\text{em3}} - \mathbf{y}_{n+1}^{\text{IMR}} + O(\Delta_n^4). \quad (37)$$

4.2 The Variable Step eBDF3 Method

Both explicit and implicit BDF methods are derived from a divided difference representation of an interpolation polynomial $\mathbf{p}(t)$ through $\mathbf{y}_{n+1}$ and a number of history values. Setting $\mathbf{p}'(t_{n+1}) = \mathbf{f}(t_{n+1}, \mathbf{y}_{n+1})$ leads to implicit BDF methods, while $\mathbf{p}(t_n) = \mathbf{f}(t_n, \mathbf{y}_n)$ gives the family of the explicit BDF methods [17, p. 364]. The Lagrange interpolation polynomial of degree k [17, p. 400] can be written in terms of the backward Newton divided differences $\mathbf{y}[t_{n+1}, \dots, t_{n+1-j}]$, $j = 1, \dots, k$ as

$$\mathbf{p}_k(t) = \mathbf{y}_{n+1} + \sum_{j=1}^k \mathbf{y}[t_{n+1}, \dots, t_{n+1-j}] \prod_{i=0}^{j-1} (t - t_{n+1-i}), \quad (38)$$

Differentiating (38) yields

$$\mathbf{p}'_k(t) = \sum_{j=1}^k \mathbf{y}[t_{n+1}, \dots, t_{n+1-j}] \sum_{\ell=0}^{j-1} \prod_{\substack{i=0 \\ i \neq \ell}}^{j-1} (t - t_{n+1-i}). \quad (39)$$

Equation (39) can be simplified by noticing that the product on the right-hand side evaluated at $t = t_n$ (for $i = 1$) is zero. This gives

$$\begin{aligned} \mathbf{f}(t_n, \mathbf{y}_n) &= \sum_{j=1}^k \mathbf{y}[t_{n+1}, \dots, t_{n+1-j}] \prod_{\substack{i=0 \\ i \neq 1}}^{j-1} (t_n - t_{n+1-i}) \\ &= \mathbf{y}[t_{n+1}, t_n] + \Delta_n \sum_{j=2}^k \mathbf{y}[t_{n+1}, \dots, t_{n+1-j}] \prod_{i=2}^{j-1} (t_n - t_{n+1-j}). \end{aligned} \quad (40)$$

For $k = 1$ and $k = 2$ (40) reduces to the well-known forward Euler and the leapfrog (explicit mid-point rule) methods, respectively [3, p. 715]. For larger values of k the complexity of (40) increases significantly. The eBDF3 method is obtained for $k = 3$:

$$\begin{aligned} \mathbf{f}(t_n, \mathbf{y}_n) &= \mathbf{y}[t_{n+1}, t_n] - \Delta_n \mathbf{y}[t_{n+1}, t_n, t_{n-1}] - \Delta_n \mathbf{y}[t_{n+1}, t_n, t_{n-1}, t_{n-2}], \\ &= \frac{\mathbf{y}_{n+1} - \mathbf{y}_n}{\Delta_n} - \frac{\Delta_n}{\Delta_n + \Delta_{n-1}} \left(\frac{\mathbf{y}_{n+1} - \mathbf{y}_n}{\Delta_n} - \frac{\mathbf{y}_n - \mathbf{y}_{n-1}}{\Delta_{n-1}} \right) \\ &\quad - \frac{\Delta_n}{\Delta_n + \Delta_{n-1} + \Delta_{n-2}} \left(\frac{\frac{\mathbf{y}_{n+1} - \mathbf{y}_n}{\Delta_n} - \frac{\mathbf{y}_n - \mathbf{y}_{n-1}}{\Delta_{n-1}}}{\Delta_n + \Delta_{n-1}} - \frac{\frac{\mathbf{y}_n - \mathbf{y}_{n-1}}{\Delta_{n-1}} - \frac{\mathbf{y}_{n-1} - \mathbf{y}_{n-2}}{\Delta_{n-2}}}{\Delta_{n-1} + \Delta_{n-2}} \right). \end{aligned} \quad (41)$$

The rearrangement of (41) for $\mathbf{y}_{n+1}$ was done using the symbolic library SymPy [32,33] resulting in

$$\mathbf{y}_{n+1} = b\mathbf{f}(t_n, \mathbf{y}_n) + c_n \mathbf{y}_n + c_{n-1} \mathbf{y}_{n-1} + c_{n-2} \mathbf{y}_{n-2}, \quad (42)$$

where

$$\begin{aligned} b &= \frac{\Delta_{n+1}}{\Delta_n(\Delta_n + \Delta_{n-1})} (\Delta_{n+1} + \Delta_n)(\Delta_{n+1} + \Delta_n + \Delta_{n-1}), \\ c_n &= -\frac{(2\Delta_{n+1}\Delta_n + \Delta_{n+1}\Delta_{n-1} - \Delta_n^2 - \Delta_n\Delta_{n-1})}{\Delta_n^2(\Delta_n + \Delta_{n-1})^2} (\Delta_{n+1} + \Delta_n)(\Delta_{n+1} + \Delta_n + \Delta_{n-1}), \\ c_{n-1} &= \frac{\Delta_{n+1}^2}{\Delta_n^2\Delta_{n-1}} (\Delta_{n+1} + \Delta_n + \Delta_{n-1}), \\ c_{n-2} &= -\frac{\Delta_{n+1}^2(\Delta_{n+1} + \Delta_n)}{\Delta_{n-1}(\Delta_n + \Delta_{n-1})^2}. \end{aligned} \quad (43)$$

For constant step size ($\Delta_n = \Delta_{n-1} = \Delta_{n+1} = \Delta$) (43) reduces to the scheme given in [17, p. 364] with $b = 3\Delta$, $c_n = -\frac{3}{2}$, $c_{n-1} = 3$, $c_{n-2} = -\frac{1}{2}$. The computational cost of (43) can be high for linear scalar IVPs. However, if non-linear IVP systems are solved in the context of the method of lines, the cost of computing (43) is small compared to the (repeated) solution of a linear system required by an implicit corrector.

The LTE estimate for IMR (37) is used to predict the next step size via the standard heuristics that involves the user-prescribed tolerance ε_T [3, p. 268]:

$$\Delta_{n+1} = \Delta_n \left(\frac{\varepsilon_T}{\|\mathbf{T}_n\|} \right)^{1/3} = \Delta_n \left(\frac{\varepsilon_T}{\|\mathbf{y}_{n+1}^{\text{em3}} - \mathbf{y}_{n+1}^{\text{IMR}}\|} \right)^{1/3}, \quad (44)$$

where $\|\cdot\|$ is a suitable norm of the LTE (e.g. a mass matrix norm, see [3, p. 708]). In the example from Sect. 5.1 we do not introduce any further restrictions in the step selection

while in the cases from Sects. 5.2 and 5.3 we make the criterion (44) more robust by adding the following two heuristical rules, which are in line with the recommendations from [19, p. 432]: i) if Δ_{n+1}/Δ_n is smaller than 0.7, we adopt $\Delta_n = \Delta_n/2$ and repeat the calculation for the step Δ_n (the step cancellation), and ii) we limit the step size growth factor Δ_{n+1}/Δ_n by 4 to improve numerical stability.

We notice that some public IVP solvers, such as CVODE [8, Sect. 2.1] keep the step size constant as long as feasible. This allows the use of the inexact Newton's method, which does not require the assembly of the Jacobians at every iteration, thus reducing the computational cost. However, if the associated linear systems are solved by a preconditioned Krylov method, the gains in convergence speed obtained using exact Newton's method can outweigh the savings obtained by not assembling the Jacobian matrices [34, p. 128]. The adaptive high-order explicit method from [7] also selects fairly constant time steps over long time intervals.

5 Numerical Examples

In this section we present numerical experiments that demonstrate the effectiveness, accuracy, and geometric properties of the proposed adaptive IMR scheme. The first case study is an ODE system modelling a magnetization reversal in a small spherical (both isotropic and anisotropic) ferromagnetic particle, while the two further cases are the PDE problems which involve spatial variation of the magnetization vector and are discretised using finite element method and the method of lines is applied to the semi-discretised system of IVPs [35]. The implementation of the ODE example in Sect. 5.1 is done in Matlab, while the PDE examples are implemented in `omph-lib`, an object-oriented multi-physics finite element library [36].

5.1 The Magnetic Reversal of a Spherical Particle

This is a widely used model-problem from micromagnetics which is represented by an ODE system. We consider the magnetization of a small spherical particle made of either an isotropic or anisotropic ferromagnetic material under a spatially uniform applied field $\mathbf{h}_{\text{ap}}$. The evolution of the magnetization vector $\mathbf{m} \in C^1(T)$ follows the LL Eq. (1) with $\mathbf{h} = \mathbf{h}_{\text{ap}} + k_1(\mathbf{m} \cdot \mathbf{e})\mathbf{e}$ is the effective field. In case of an isotropic material ($k_1 = 0$) $\mathbf{h}$ has only the applied field component [18, p. 306]. For a sufficiently small sphere the magnetization is spatially uniform, i.e. $\mathbf{m} = \mathbf{m}(t)$ and the problem (1) is a system of three ODEs for the unknown Cartesian components of $\mathbf{m}$. The problem has an analytical solution [37], which describes the process of the magnetization switching and can be expressed in spherical polar coordinates (θ, ϕ) as $\theta = \cos^{-1}(m_z/1)$, $\phi = \tan^{-1}(m_y/m_x)$, where θ is the angle between $\mathbf{m}$ and $\mathbf{h}_{\text{ap}}$. The energy of the system is in this case given by $E = -\mathbf{m} \cdot \mathbf{h}_{\text{ap}}$. In the ideal non-damped case ($\alpha = 0$) the energy remains conserved throughout the magnetization reversal (i.e. the process of the realignment of $\mathbf{m}$ from the initial configuration $\mathbf{m}_0$ to the external field $\mathbf{h}_{\text{ap}}$) – the property that needs to be replicated by the time integrator. For anisotropic materials ($k_1 > 0$) we study the behaviour of the time integrators for the realistic values of the phenomenological anisotropy parameter $k_1 \in [0, 4]$.

a) Isotropic case ($k_1 = 0$). We consider the damping value $\alpha = 0.01$. In this case the problem (1) is only mildly non-linear. Thus, we would expect that the AB2-IMR method performs well, and its step size selection to be comparable to that obtained with a third-order predictor. We choose $\mathbf{h}_{\text{ap}} = (0, 0, -1.1)^t$, which is a sufficient magnetic field strength

Fig. 1 Time evolution of the Cartesian components of the magnetization vector $\mathbf{m}$ in the spherical isotropic particle reversal problem with $\alpha = 0.01$

Table 1 The number of the time steps N_t and the minimum of magnetization length $|\mathbf{m}|_{\min}$ for the spherical particle reversal problem with $\alpha = 0.01$ and $k_1 = 0$ integrated over the time interval $T = [0, 1000]$ using GL-BDF2, AB2-TR, AB2-IMR and eBDF3-IMR schemes

ϵ_T	Scheme	GL-BDF2	AB2-TR	IMR	
				AB2	eBDF3
10^{-4}	N_t	9693	5724	5724	8311
	$ \mathbf{m} _{\min}$	0.980221	0.997881	1.000000	
10^{-5}	N_t	20,787	12,321	12,321	17,798
	$ \mathbf{m} _{\min}$	0.998045	0.999544	1.000000	
10^{-6}	N_t	44,765	26,538	26,538	38,289
	$ \mathbf{m} _{\min}$	0.999754	0.999906	1.000000	

to reverse the initial magnetization $\mathbf{m}_0 = \hat{\mathbf{m}}_0 / \|\hat{\mathbf{m}}_0\|$, where $\hat{\mathbf{m}}_0 = (0.01, 0, 1)^t$. In all our experiments it took 1–2 iterations for the exact Newton method to converge to a prescribed tolerance $\epsilon_N = 10^{-14}$. The system (1) is integrated over the interval $T = [0, 1000]$, which is sufficient to reach a steady state after the reversal. The time evolution of the Cartesian components of the magnetization vector $\mathbf{m} = (m_x, m_y, m_z)^t$ is presented in Fig. 1.

We first examine the efficiency of the step size selection procedure and the magnetization length conservation properties of different integration schemes as a function of the LTE tolerance ϵ_T . The results are summarised in Table 1. From these figures we see that the adaptive step IMR method conserves $|\mathbf{m}|$ to the level of Newton’s tolerance ϵ_N , while this is not the case for the BDF2 and TR methods. TR is more accurate in this context than BDF2, which can also be seen from Fig. 2, which depicts the time evolution of the magnetization length for the three time integrators. Comparing the number of time steps required to complete the integration, it follows that AB2-TR and AB2-IMR have the same behaviour (which is expected in the case of linear and mildly non-linear problems). The combination eBDF3-IMR leads to approximately 40% higher number of time steps than the AB2 predictor for a fixed value of ϵ_T . The time step evolution for various integrators is presented in Fig. 3.

Fig. 2 Time evolution of the magnetization length $|\mathbf{m}|$ for BDF2, TR and IMR methods for the isotropic spherical particle reversal problem with $\alpha = 0.01$ and $\epsilon_T = 10^{-4}$

Fig. 3 The time step size evolution for BDF2, TR, and IMR, for the spherical particle reversal problem with $\alpha = 0.01$, $k_1 = 0$ and $\epsilon_T = 10^{-5}$

Next we test the accuracy of the proposed time integrators. To this end, we record the time t_1 for which $m_z(t_1) = 0$ (cf. Fig. 1), i.e. the time t_1 for which $\mathbf{m} \cdot \mathbf{k} = 0$. The reference solution for this problem is calculated by both adaptive TR and IMR with a tight tolerance ($\epsilon_T = 10^{-10}$) as $t_s = 481.72$. In Table 2 we present the accuracy results when all the tested methods take the same number of steps to perform the simulation over the interval $T = [0, 490]$. This is achieved by adjusting the LTE tolerance ϵ_T for each method separately. The rationale behind this experiment is to evaluate the quality of the integration points distribution for each of the predictors considered.

From these results we see that the combination eBDF3-IMR is as accurate as AB2-TR. This is due to the character of the problem, which is only weakly non-linear, implying a near

Table 2 The switching times t_1 for which $m_z(t_1) = 0$ with fixed number of time steps N_t for the spherical particle reversal problem with $\alpha = 0.01$ and $k_1 = 0$

Scheme	GL-BDF2	AB2-TR	IMR	
			AB2	eBDF3
ϵ_T	$3.808 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$3.032 \cdot 10^{-5}$
N_t	6231	6231	6231	6231
t_1	> 490	483.46	483.48	483.46
ϵ_T	$4.659 \cdot 10^{-6}$	$1 \cdot 10^{-6}$	$1 \cdot 10^{-6}$	$3.010 \cdot 10^{-6}$
N_t	13,474	13,474	13,474	13,474
t_1	485.90	482.09	482.09	482.09
ϵ_T	$4.780 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$3.003 \cdot 10^{-7}$
N_t	29,053	29,053	29,053	29,053
t_1	482.31	481.80	481.80	481.80

equivalence of the TR and IMR schemes. Indeed, for the tight LTE tolerances ϵ_T the TR and IMR schemes are virtually indistinguishable.

b) Anisotropic case ($k_1 \neq 0$). This case involves material anisotropy, expressed by an additional field component (the anisotropic field). We test the efficiency and accuracy of the proposed time integrators when the non-linearity in the problem, controlled by the parameter k_1 , increases. In particular, there is no equivalence between the TR and the IMR method in these cases, and we expect that the third order predictor for IMR leads to a better accuracy when the number of time steps is fixed, when compared to AB2-TR and AB2-IMR methods. As in the previous example, we choose $\mathbf{h}_{\text{ap}} = (0, 0, -1.1)^t$, $\mathbf{m}_0 = \hat{\mathbf{m}}_0 / \|\hat{\mathbf{m}}_0\|$, where $\hat{\mathbf{m}}_0 = (0.01, 0, 1)^t$, and select the direction of the material anisotropy as $\hat{\mathbf{e}} = \mathbf{e} / \|\mathbf{e}\|$ where $\mathbf{e} = (1, -0.3, 0)^t$. The remaining problem and algorithm parameters are the same as in the isotropic case. The size of the time interval required to complete the reversal varies for different values of k_1 and is selected to be sufficiently long for the system to reach a steady state. For any non-zero value of k_1 the steady-state position of the magnetization vector after the reversal is no longer aligned with $\mathbf{h}_{\text{ap}}$. The time evolution of the magnetization vector components with $k_1 = 4$ is presented in Fig. 4.

In Table 3 we present the numbers of time steps N required to complete the integration for the LTE tolerance $\epsilon_T = 10^{-5}$ and the bounds for the magnetization length $|\mathbf{m}|$ for different values of k_1 and different integrators. From the table we see that the increase in k_1 leads to the poorer magnetization length preservation (although the used level of tolerance ϵ_T gives fairly good values for all integrators, looser values of ϵ_T will cause problems for BDF2 and TR, with notable loss of accuracy in $|\mathbf{m}|$, see Fig. 5 for the case $\epsilon_T = 10^{-4}$). As k_1 is increased, the combination AB2-IMR requires fewer steps than AB2-TR.

The time step size history for three different combinations of time integrators is presented in Fig. 6. Comparing to the isotropic case, where a smooth variation in step sizes is observed in all time integrators, we see that the step size selection exhibits a considerably more dynamic behaviour, which is the consequence of a more complex dynamics in the problem.

Next, we report the accuracy of the first zero crossing in the z component of the magnetization for $k_1 = 4$ (i.e. the smallest time t_1 for which $m_z(t_1) = 0$). We do this with respect to two different criteria: fixed number of time steps and fixed execution time for all tested integrators. The first criterion reflects the quality of a predictor, i.e. how the distribution of a prescribed number of the integration points that it produces affects the solution accuracy. The second criterion tests whether the computational overhead associated with adaptive step

Fig. 4 Time evolution of the Cartesian components of the magnetization vector $\mathbf{m}$ in the spherical particle reversal problem with $\alpha = 0.01$ and $k_1 = 4$

Table 3 The number of the time steps N_t and the magnetization length interval $[|\mathbf{m}|_{\min}, |\mathbf{m}|_{\max}]$ for (1) for the anisotropic spherical particle reversal problem with $\alpha = 0.01$ as a function of k_1 with $\epsilon_T = 10^{-5}$. The integration is done over the interval $T = [0, t_{k_1}]$

k_1 t_{k_1}	Scheme	GL-BDF2	AB2-TR	IMR	
				AB2	eBDF3
0	N_t	20,787	12,321	12,323	17,798
1000	$ \mathbf{m} _{\min}$	0.998045	0.999544	1.000000	
	$ \mathbf{m} _{\max}$	1.000000	1.000000		
0.4	N_t	21,071	12,486	12,464	17,915
1100	$ \mathbf{m} _{\min}$	0.998288	0.999360	1.000000	
	$ \mathbf{m} _{\max}$	1.000000	1.000083		
1	N_t	19,135	11,310	11,209	15,768
1250	$ \mathbf{m} _{\min}$	0.998386	0.999360	1.000000	
	$ \mathbf{m} _{\max}$	1.000001	1.000069		
2.5	N_t	19,501	11,665	11,530	15,926
1250	$ \mathbf{m} _{\min}$	0.997159	0.997707	1.000000	
	$ \mathbf{m} _{\max}$	1.000001	1.000001		
4	N_t	18,396	11,092	10,970	15,204
1250	$ \mathbf{m} _{\min}$	0.997139	0.997560	1.000000	
	$ \mathbf{m} _{\max}$	1.000002	1.000001

Fig. 5 Time evolution of the magnetization length $|m|$ for BDF2, TR and IMR methods for the anisotropic spherical particle reversal problem with $\alpha = 0.01$, $k_1 = 4$ and $\epsilon_T = 10^{-4}$

Fig. 6 The time step size evolution for GL-BDF2, AB2-TR and eBDF3-IMR methods for the anisotropic spherical particle reversal problem with $\alpha = 0.01$, $k_1 = 4$ and $\epsilon_T = 10^{-5}$

selection leads to a more accurate method than computationally cheaper fixed step counterpart within a set amount of wall clock time.

For the first criterion we consider three different levels of precision, determined by a fixed number of steps N_1 required to cover the interval $[0, 150]$ for all methods. These levels are $N_1 = 4142$, 8967 , and $19,336$, obtained by setting ϵ_T in AB2-TR method to 10^{-5} , 10^{-6} , and 10^{-7} , respectively, and adjusting it in other integrators to match this number of steps. The reference solution $t_1^{\text{ref}} = 145.038$ is obtained by the eBDF3-IMR and AB2-TR methods with $\epsilon_T = 10^{-10}$. The results are presented in Table 4. From the results we observe that

Table 4 The computed minimum values of t_1 for which $m_z(t_1) = 0$ for the anisotropic spherical particle reversal problem with $\alpha = 0.01$, $k_1 = 4$ as a function of the number of time steps N_1

Scheme	GL-BDF2	AB2-TR	IMR	
			AB2	eBDF3
ϵ_T	$4.083 \cdot 10^{-5}$	$1 \cdot 10^{-5}$	$9.660 \cdot 10^{-6}$	$2.607 \cdot 10^{-5}$
N_1	4142	4142	4142	4142
t_1	> 150	145.160	145.144	145.136
ϵ_T	$4.690 \cdot 10^{-6}$	$1 \cdot 10^{-6}$	$9.640 \cdot 10^{-7}$	$2.558 \cdot 10^{-6}$
N_1	8967	8967	8967	8967
t_1	145.100	145.063	145.060	145.059
ϵ_T	$4.782 \cdot 10^{-7}$	$1 \cdot 10^{-7}$	$9.642 \cdot 10^{-8}$	$2.547 \cdot 10^{-7}$
N_1	19,336	19,336	19,336	19,336
t_1	145.051	145.044	145.043	145.042

Table 5 The computed minimum values of t_1 for which $m_z(t_1) = 0$ for the anisotropic spherical particle reversal problem with $\alpha = 0.01$, $k_1 = 4$ for fixed step and adaptive IMR method under the condition of the same execution time

Method	eBDF3-IMR		Fixed step IMR	
	N_t	t_1	N_t	t_1
10^{-4}	7426	145.297	7459	146.896
10^{-5}	15,652	145.089	16,951	145.112
10^{-6}	33,439	145.049	35,238	145.063

the eBDF3-IMR combination is the most accurate, although for very tight tolerances ϵ_T the differences to AB2-TR and AB2-IMR methods become less pronounced.

For the second criterion (fixed execution time) we compare only the adaptive and the fixed step versions of the IMR integrator. We consider again three different levels of precision for adaptive IMR and adjust the number of time steps for the fixed step variant so that both methods have the same execution time to perform the integration over the time interval $[0, 500]$. In Table 5 we report the values of t_1 at which the first zero crossing $m_z(t_1)$ occurs. From these figures it can be concluded that fixed step IMR completes more steps in a set amount of time (due to an increased computational cost per step in the adaptive method), but still has lower accuracy, justifying the additional costs of the adaptive version. We remark that the number of steps reported in Table 5 suggest that the computational cost of an adaptive step is only 5–10% larger than that of a fixed step. This is expected in the case of non-linear problems, with the cost of computing a predictor consists of a matrix-vector multiplication, while the cost of the corrector is dominated by multiple linear system solves that are costly.

Finally, in Table 6 we report the energy $E = -\mathbf{m} \cdot (\mathbf{h}_{\text{ap}} + \mathbf{h}_{\text{mc}})$ at the time $t = 600$ of the simulation with $k_1 = 4$ as the function of the LTE tolerance ϵ_T . This time is sufficient for the magnetization vector to reach the steady-state after the reversal. We observe from the results that for the larger values of ϵ_T there is a notable error in the computed energy (more than 5% for the BDF2 method with $\epsilon_T = 10^{-4}$). By contrast, the computed energy remains constant with the IMR method, irrespectively of the LTE tolerance.

Table 6 The energy $E = -\mathbf{m} \cdot (\mathbf{h}_{\text{ap}} + \mathbf{h}_{\text{mc}})$ for the anisotropic spherical particle reversal problem at $t = 600$ with $\alpha = 0.01$ and $k_1 = 4$, as the function of the LTE tolerance ϵ_T

ϵ_T	GL-BDF2	AB2-TR	eBDF3-IMR
10^{-4}	− 3.7685	− 3.8038	− 4.0000
10^{-5}	− 3.9771	− 3.9817	− 4.0000
10^{-6}	− 3.9976	− 3.9981	− 4.0000

5.2 The PDE Example with an Analytical Solution

This model problem is defined in the domain $\Omega \subset \mathbb{R}^2$, and is modelled by the non-dimensional form of the LLG equation (4). In this example we consider only the exchange coupling, hence $\mathbf{h} = \nabla^2 \mathbf{m}$. We report on the behaviour of the adaptive IMR when applied to a problem whose analytical solution has a non-trivial spatial variation. A more detailed description can be found in [38,39]. We consider an infinite two-dimensional domain, which is reduced to a finite counterpart by applying periodic BCs. For chosen constants $c \in \mathbb{R}$ and $\mathbf{k} \in \mathbb{R}^3$, define

$$\begin{aligned} \hat{t} &= \frac{t}{1 + \alpha^2}, \quad b(t) = |\mathbf{k}|^2 \alpha \hat{t}, \quad d(t) = \sqrt{\sin^2(c) + \cos^2(c) e^{2b(t)}}, \\ g(t) &= \frac{1}{\alpha} \log \left[\frac{d(t) + \cos(c) e^{b(t)}}{1 + \cos(c)} \right]. \end{aligned} \quad (45)$$

Then the components of the magnetization are given by

$$\begin{aligned} m_x(t) &= \frac{1}{d(t)} \sin(c) \cos(\mathbf{k} \cdot \mathbf{x} + g(t)), \quad m_y(t) = \frac{1}{d(t)} \sin(c) \sin(\mathbf{k} \cdot \mathbf{x} + g(t)), \\ m_z(t) &= \frac{1}{d(t)} \cos(c) e^{b(t)}. \end{aligned} \quad (46)$$

For large intervals of the values of c and $\mathbf{k}$ (46) represents a damped periodic precession tending to a steady state $\mathbf{m} = [0 \ 0 \ \pm 1]^t$ as $t \rightarrow \infty$, reducing to a simple wave in space and time for $\alpha \rightarrow 0$.

The domain $\Omega = [0, 1]^2 \subset \mathbb{R}^2$ is used in our simulations with periodic BCs. Spatial discretisation is performed by finite elements (uniform quadrilateral elements with the resolution 80×80 resulting in 1681 nodes). Our spatial discretisation is similar to that used in [20]. The difference between the two approaches is that instead of a provably convergent, but restrictive linearisation procedure (the convergence is proved under a time step size restriction $\Delta = O(h^2)$), we use the standard Newton method with a tolerance linked to the desired level of error in the magnetization length and/or energy, leaving to the adaptive eBDF3-IMR procedure to select the appropriate time step sizes. The initial condition is set according to (46) for $t = 0$ and $\mathbf{h}_{\text{ap}} = \mathbf{0}$. The solution parameters are $\mathbf{k} = [2\pi, 2\pi]^t$, $c = 0.1\pi$ and $\mathbf{h}_{\text{ap}} = 0.01$. For the energy conservation experiments we set $\alpha = 0$.

The snapshot of the solution at $t = 0.1$ is given in Fig. 7. In Fig. 8 we depict the error in the nodal magnetization

$$\mathcal{E}_{|\mathbf{m}|}(t_n) = \max_{j \in V} \left| |\mathbf{m}_{j,n}| - 1 \right|, \quad (47)$$

where $\mathbf{m}_{j,n}$ are the FE approximations at the nodes j of the magnetization vector at $t = t_n$ and V is the set of all nodes in the mesh. We consider four different configurations: BDF2, TR and IMR with Gaussian integration, and IMR with nodal (reduced) integration. When IMR with nodal quadrature is used the error (47) remains small and controlled by the adopted

Fig. 7 Snapshot of the solution of the PDE model problem with a periodic wave solution at $t = 0.1$. Different colours represent the values of the magnetization component m_x . The component m_z is constant throughout the domain ($m_z = 0.958$) (Color figure online)

Fig. 8 Evolution of the magnetization length error $\mathcal{E}_{|\mathbf{m}|}(t)$ for different time integrators and quadrature schemes applied to the model PDE problem

Newton's tolerance ε_N . When Gaussian quadratures are used, the error (47) grows rapidly up to 10^{-4} at which level it remains fairly constant, irrespective of the time integration scheme.

We next examine the energy conservation properties for different schemes for the non-damped case $\alpha = 0$. The only energy component in this example is the exchange energy defined by [13, Ch. 2]

$$e_{\text{ex}} = \frac{1}{2} \int_{\Omega} (\nabla \mathbf{m})^2 d\Omega. \quad (48)$$

The integral in (48) can be evaluated exactly by both the Gaussian and the nodal quadratures when piecewise linear elements are used, since $\nabla \mathbf{m}$ is a constant in each element. The results are summarised in Fig. 9. We observe that BDF2 does not conserve energy, but both TR and IMR with Gaussian quadratures perform well for this particular problem. The best results are

Fig. 9 Evolution of energy change $e_{\text{ex}}(t)$ for different time integrators and quadrature schemes for the model PDE problem with $\alpha = 0$

obtained, however, when IMR and nodal quadratures in FE discretisation are used. Accurate results for the length and energy conservation with Gaussian quadratures are not observed for any of the time integrators in more general cases (see, for example, [13, Sect. 8.2] for the test-problem with a non-uniform external field).

5.3 The NIST μMAG Standard Problem #4

The μMAG standard problem #4 [40] is a widely used benchmark for testing various micro-magnetic methods and models. It involves modelling the reversal of the magnetization in a thin film of a permalloy material under an applied field. The problem is modelled by a PDE system (4)–(9), where we consider an isotropic case ($k_1 = 0$). The damping parameter is $\alpha = 0.02$ and we consider two cases of applied field: $\mu_0 \mathbf{h}_{\text{ap},1} = [-24.6 \ 4.3 \ 0.0]^t \text{ mT}$ and $\mu_0 \mathbf{h}_{\text{ap},2} = [-35.5 \ -6.3 \ 0.0]^t \text{ mT}$, where $\mu_0 = 4\pi \cdot 10^{-7} \text{ H} \cdot \text{m}^{-1}$. The non-dimensionalisation of the LLG equation results in the unit length of 5.69 nm and the unit time of 5.65 ps. Hence, the non-dimensional domain is of size $87.94 \times 21.98 \times 0.53$.

The initial condition $\mathbf{m}_0$ is the S-state obtained by slowly relaxing the magnetization from the state obtained by a saturating field in the $[1 \ 1 \ 1]^t$ direction [40]. The S-state is obtained by applying the field

$$\mathbf{h}_{\text{ap}} = \begin{cases} 10 \left(1 - \frac{t}{100}\right) & t < -200 \\ 0 & t \geq -200 \end{cases} \quad (49)$$

over the non-dimensional time interval $[-300, 0]$ to the initial state $\mathbf{m} = [1 \ 1 \ 1]^t$ with $\alpha = 1$. This choice allows the system to reach the initial condition quickly.

The domain Ω is discretised by a structured uniform mesh of hexahedral elements with a single layer of elements in the z -direction. Then we apply the hybrid FEM/BEM method described in Sect. 2.3. The parameters used in the eBDF3-IMR method are $\Delta_0 = 10^{-4}$ and $\epsilon_T = 10^{-5}$. Newton's method tolerance is $\epsilon_N = 10^{-11}$. We comment briefly on the computational cost of the FEM/BEM method. At each time step the eBDF3 predictor requires

one sparse matrix-vector multiplication (recall that we use hierarchical approximation of the BEM matrix) at a cost $O(N + N_b \log N_b)$ and a trivial solution of a linear system with a mass coefficient matrix with a cost $O(N)$ (when reduced integration is used, the mass matrix is diagonal by construction [13, p. 104]). The Newton iteration in the corrector takes 2–3 steps to converge [13, Fig. 8.24]. The cost of assembling the Jacobian is $O(N)$, as the BEM matrix does not need to be reassembled. The Krylov solver is preconditioned by a general algebraic, rather than a problem-specific, preconditioner, leading to an inevitable growth in iteration counts as N is increased. For our simulations GMRES converged in 10–25 iterations. Therefore, we pessimistically estimate the computational cost per time step of the eBDF3-IMR method as $150C_{mv}$, where C_{mv} is the cost of a sparse matrix-vector multiplication.

We compare the accuracy of the solutions obtained by the adaptive IMR scheme to the results from [12] obtained by the IMR scheme with fixed step sizes and finite difference method applied to spatial discretisation (magnetostatic effects are handled by the FFT and monolithically coupled to the LLG model). In Fig. 10 we present the mean magnetization and the time step sizes Δ_n generated by the proposed adaptive scheme for the field $\mathbf{h}_{ap,1}$, while Fig. 11 plots the same results for $\mathbf{h}_{ap,2}$.

The results obtained by adaptive IMR (in red) are superimposed to those in green plotted using the raw data from [12]. The two solutions agree reasonably well, except in the interval [100, 200] for $\mathbf{h}_{ap,2}$. It is worth noticing that this is the interval where all the solutions submitted to μ MAG repository [40] exhibit different behaviour. The upshot is that the proposed adaptive algorithm correctly identifies this difficult part of the time interval and responds by cutting the

Fig. 10 The evolution of the Cartesian components of the mean magnetization and the time step size Δ_n for the μ MAG problem #4 for $\mathbf{h}_{ap,1}$ obtained with adaptive IMR and FEM (red) and the solution plotted using the raw data from [12] (green). One non-dimensional time unit is equal to 5.65 ps (Color figure online)

Fig. 11 The evolution of the Cartesian components of the mean magnetization and the time step size Δ_n for the μ MAG problem #4 for $\mathbf{h}_{\text{ap},2}$ obtained with adaptive IMR and FEM (red) and the solution plotted using the raw data from [12] (green). One non-dimensional time unit is equal to 5.65 ps (Color figure online)

stepsize to a value significantly below the fixed steps adopted in [12]. After the oscillations in $\mathbf{m}$ are damped out for $t > 200$, the step size increases steadily. In Fig. 12 we present the mean value of m_y in the critical part of the time interval calculated with two different spatial resolutions (the number of grid points in x direction is equal to the number assigned to the quadrature rule in the figure legend) and two different types of numerical quadratures.

When compared to adaptive BDF2 and TR methods, IMR selects significantly smaller steps during the initial relaxation phase (for $t < 0$). This effect may be attributed to order reduction observed in IMR when applied to very stiff problems. For $t > 0$ time steps selected by all three methods are commensurate. We notice that recently introduced adaptive high-order explicit midpoint method [7] keeps the step size fairly constant both in the highlighted time interval [100, 200] and during the entire simulation for the case $\mathbf{h}_{\text{ap},1}$. The time step of 200 fs reported in [7] is between 3 and 20 times smaller than the step sizes taken by the eBDF3-IMR method for $t > 0$ (cf. Fig. 10).

The comparison of the computational cost per time step between the explicit method from [7] and the coupled FEM/BEM method is not straightforward. Results from Table 1 in [7] indicate approximately between 25 and 40 total (effective and stray) field evaluations per time step. If we assume that the cost of one field computation is proportional to C_{mv} , we can conclude that the overall computational times of both methods are broadly comparable in this particular case.²

² The cost of an implicit step is between 4 and 6 times higher than the cost of an explicit step, but the implicit step sizes are between 3 and 20 times larger than that of the explicit integrator.

Fig. 12 The mean value of m_y for the μ MAG problem #4 with $\mathbf{h}_{\text{app},2}$ for adaptive IMR with different spatial resolutions and different numerical quadratures. One non-dimensional time unit is equal to 5.65 ps

The maximum pointwise error (47) in the magnetization length observed during the simulation was in both cases $4 \cdot 10^{-10}$, in line with the adopted Newton's tolerance.

6 Conclusions

In this paper we have introduced an adaptive IMR method. Using the example domain of micromagnetic problems we have shown that the method conserves the magnetisation length and accurately reproduces the energy balance (i.e. it preserves the geometric properties of the solution even for loose LTE tolerances). The method uses a computationally efficient and reliable method for the LTE estimation in the IMR and is based on the use of a third-order explicit method as the predictor. In our experiments the selected step sizes of the adaptive IMR were commensurate to the other commonly used second-order methods, while the geometric integration properties were preserved. We also remark that geometric integrator properties of IMR are relevant for other problems (for example, if the Navier–Stokes problem is solved, the kinetic energy $\mathbf{u}'\mathbf{u}$ will be correctly reproduced independently of the step size, where $\mathbf{u}$ is the fluid velocity [3, p. 651], provided that an appropriate spatial discretisation is used, see [41]). This makes the proposed adaptive integrator suitable choice for a much wider class of problems than studied in this paper.

The codes used to produce the results in Sect. 5.1 are publicly available at <https://github.com/davidshepherd7/Landau-Lifshitz-Gilbert-ODE-model>, and for the PDE examples from Sects. 5.2 and 5.3 at <https://github.com/davidshepherd7/oomp-lib-micromagnetics>.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Ascher, U., Petzold, L.: *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, Philadelphia (1998)
2. Ascher, U., Mattheij, R., Russell, R.: *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*. SIAM, Philadelphia (1995)
3. Gresho, P.M., Sani, R.L.: *Incompressible Flow and the Finite Element Method: Advection-Diffusion and Isothermal Laminar Flow*, 1st edn. Wiley, Chichester (1998)
4. Gresho, P.M., Griffiths, D.F., Silvester, D.J.: Adaptive time-stepping for incompressible flow part I: scalar advection-diffusion. *SIAM J. Sci. Comput.* **30**(4), 2018–2054 (2008)
5. Kay, D., Gresho, P., Griffiths, D., Silvester, D.: Adaptive time stepping for incompressible flow; part II: Navier–Stokes equations. *SIAM J. Sci. Comput.* **32**, 111–128 (2010). <https://doi.org/10.1137/070688018>
6. Elman, H., Mihajlović, M., Silvester, D.: Fast iterative solvers for buoyancy driven flow problems. *J. Comput. Phys.* **230**, 3900–3914 (2011). <https://doi.org/10.1016/j.jcp.2011.02.014>
7. Exl, L., Mauser, N., Schrefl, T., Suess, D.: The extrapolated explicit midpoint scheme for variable order and step size controlled integration of the Landau–Lifshitz–Gilbert equation. *J. Comput. Phys.* **346**, 14–24 (2017). <https://doi.org/10.1016/j.jcp.2017.06.005>
8. Hindmarsh, A.C., Serban, R.: *User Documentation for cvode v2.7.0* (2012)
9. MathWorks, ode45 (2014). <http://www.mathworks.co.uk/help/matlab/ref/ode45.html>
10. Aharoni, A.: *Introduction to the Theory of Ferromagnetism*, 2nd edn. Oxford University Press, Oxford (1996)
11. Shepherd, D., Miles, J., Heil, M., Mihajlović, M.: Discretisation induced stiffness in micromagnetic simulations. *IEEE Trans. Magn.* **50**, 7201304 (2014). <https://doi.org/10.1109/TMAG.2014.2325494>
12. d’Aquino, M., Serpico, C., Miano, G.: Geometrical integration of Landau–Lifshitz–Gilbert equation based on the mid-point rule. *J. Comput. Phys.* **209**(2), 730–753 (2005). <https://doi.org/10.1016/j.jcp.2005.04.001>
13. Shepherd, D.: *Numerical methods for dynamical micromagnetics*, Ph.D. thesis, The University of Manchester (2015)
14. Fidler, J., Schrefl, T.: Micromagnetic modelling—the current state of the art. *J. Phys. D Appl. Phys.* **33**(15), R135–R156 (2000). <https://doi.org/10.1088/0022-3727/33/15/201>
15. Szabolics, H., Buda-Prejbeanu, L., Toussaint, J., Fruchart, O.: A constrained finite element formulation for the Landau–Lifshitz–Gilbert equations. *Comput. Mater. Sci.* **44**, 253–258 (2008). <https://doi.org/10.1016/j.commatsci.2008.03.019>
16. Fischbacher, T., Fangohr, H.: Continuum multi-physics modeling with scripting languages: the Nsim simulation compiler prototype for classical field theory, arXiv (2009) 1–50. [arXiv:0907.1587v1](https://arxiv.org/abs/0907.1587v1)
17. Hairer, E., Morsett, S.P., Wanner, G.: *Solving Ordinary Differential Equations I: Nonstiff Problems*, 2nd edn. Springer, Berlin (1991)
18. Coey, J.: *Magnetism and Magnetic Materials*, 1st edn. Cambridge University Press, Cambridge (2010)
19. Elman, H., Silvester, D., Wathen, A.: *Finite Elements and Fast Iterative Solvers*, 2nd edn. Oxford University Press, Oxford (2014)
20. Bartels, S., Prohl, A.: Convergence of an implicit finite element method for the Landau–Lifshitz–Gilbert equation. *SIAM J. Numer. Anal.* **44**(4), 1405 (2006). <https://doi.org/10.1137/050631070>
21. Cimrák, I.: A survey on the numerics and computations for the Landau–Lifshitz equation of micromagnetism. *Arch. Comput. Methods Eng.* **15**(3), 277–309 (2008). <https://doi.org/10.1007/s11831-008-9021-2>
22. Suess, D., Tsiantos, V., Schrefl, T., Fidler, J., Scholtz, W., Forster, H., Dittrich, R., Miles, J.: Time resolved micromagnetics using a preconditioned time integration method. *J. Magn. Magn. Mater.* **248**(2), 298–311 (2002). [https://doi.org/10.1016/S0304-8853\(02\)00341-4](https://doi.org/10.1016/S0304-8853(02)00341-4)
23. Ying, L.: *Infinite Element Methods*. Springer, Berlin (1995)
24. Yang, B.: *Numerical studies of dynamical micromagnetics*, Ph.D. thesis, University of California (1997)
25. Fredkin, D., Koehler, T.: Hybrid method for computing demagnetizing fields. *IEEE Trans. Magn.* **26**(2), 415–417 (1990)
26. Bottauscio, O., Chiampì, M., Manzini, A.: A finite element procedure for dynamic micromagnetic computations. *IEEE Trans. Magn.* **44**(11), 3149–3152 (2008). <https://doi.org/10.1109/TMAG.2008.20016>
27. Knittel, A., Franchin, M., Bordignon, G., Fischbacher, T., Bending, S., Fangohr, H.: Compression of boundary element matrix in micromagnetic simulations. *J. Appl. Phys.* **105**(7), 07D542 (2009)
28. Borm, S., Grasedyck, L.: *HLib*. <http://www.hlib.org/>
29. Atkinson, K.E., Han, W., Stewart, D.E.: *Numerical Solution of Ordinary Differential Equations*, 1st edn. Wiley, Hoboken (2009)

30. Lee, J., Nam, J., Pasquali, M.: A new stabilization of adaptive step trapezoid rule based on finite difference interrupts. *SIAM J. Sci. Comput.* **37**, A725–A746 (2015). <https://doi.org/10.1137/140966915>
31. Simo, J., Armero, F.: Unconditional stability and long-term behaviour of transient algorithms for the incompressible Navier–Stokes and Euler equations. *Comput. Methods Appl. Mech. Eng.* **111**, 111–154 (1994). [https://doi.org/10.1016/0045-7825\(94\)90042-6](https://doi.org/10.1016/0045-7825(94)90042-6)
32. SymPy: Python library for symbolic mathematics. <http://www.sympy.org/>
33. Shepherd, D.: sym-bdf3.py. <https://github.com/davidshepherd7/Landau-Lifshitz-Gilbert-ODE-model/blob/master/algebra/sym-bdf3.py>
34. Iserles, A.: *A First Course in the Numerical Analysis of Differential Equations*, 2nd edn. Cambridge University Press, Cambridge (2009)
35. Gilbert, T.: A phenomenological theory of damping in ferromagnetic materials. *IEEE Trans. Magn.* **40**(6), 3443–3449 (2004). <https://doi.org/10.1109/TMAG.2004.836740>
36. Heil, M., Hazel, A.L.: oomph-lib—An object-oriented multi-physics finite-element library. In: *Fluid-Structure Interaction, Lecture Notes in Computational Science and Engineering*, vol. 53, pp. 19–49. Springer, Berlin (2006)
37. Mallinson, J.: Damped gyromagnetic switching. *IEEE Trans. Magn.* **36**(4), 1976–1981 (2000)
38. Jeong, D., Kim, J.: An accurate and robust numerical method for micromagnetics simulations. *Curr. Appl. Phys.* **14**(3), 476–483 (2014). <https://doi.org/10.1016/j.cap.2013.12.028>
39. Fuwa, A., Ishiwata, T., Tsutsumi, M.: Finite difference scheme for the Landau–Lifshitz equation. *Proc. Czech-Jpn. Semin. Appl. Math.* **29**, 107–113 (2006)
40. McMichael, B.: μ MAG – Micromagnetic modelling activity group. <http://www.ctcms.nist.gov/rdm/mumag.org.html>
41. Cliffe, K.: On conservative finite element formulations of the inviscid Boussinesq equations. *Int. J. Numer. Methods Fluids* **1**(2), 117–127 (1981)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.