

Adaptive mesh refinement for the Landau–Lifshitz–Gilbert equation

Jan Bohn, Willy Dörfler, Michael Feischl, Stefan Karch

CRC Preprint 2023/9, March 2023

KARLSRUHE INSTITUTE OF TECHNOLOGY

CRC 1173



Wave
phenomena

Participating universities



Universität Stuttgart

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN



Funded by

DFG

ADAPTIVE MESH REFINEMENT FOR THE LANDAU–LIFSHITZ–GILBERT EQUATION

JAN BOHN, WILLY DÖRFLER, MICHAEL FEISCHL, STEFAN KARCH

ABSTRACT. We propose a new adaptive algorithm for the approximation of the Landau–Lifshitz–Gilbert equation via a higher-order tangent plane scheme. We show that the adaptive approximation satisfies an energy inequality and demonstrate numerically, that the adaptive algorithm outperforms uniform approaches.

1. INTRODUCTION

The Landau–Lifshitz–Gilbert equation (LLG) serves as an important practical tool and as a valid model of micromagnetic phenomena occurring in, e.g., magnetic sensors, recording heads, and magneto-resistive storage device [20, 25, 27]. Cutting edge examples include *heat assisted magnetic recording* [29], where lasers heat the magnetic recording material locally in order to speed up the writing process. Other developments, such as *magneto-resistive memory* and *skyrmion based race-track memory* focus on new ways to store data magnetically, see e.g., [18]. All of these applications have in common that the magnetic materials exhibit a strong locality (in time and space) of magnetic effects [28]: So called domain walls reveal a change of direction of the magnetization on small length scales. Similarly, temporal behavior is characterized by rapid changes (switching processes) followed by long relatively stable periods. Those effects render non-adaptive simulations tremendously inefficient (see [28] for early experiments and [17] for current applications) and classical methods based on uniform meshes often fail to reach meaningful accuracy even on the largest computers.

Even uniform time-stepping for the LLG equation is challenging. While weak convergence of the approximations was known since at least 2008 (see, e.g., [5, 10]), strong a priori convergence of uniform time-stepping schemes that obey physical energy bounds has first been proved recently in [15] and then extended to higher-order in [2]. The latter two works build on the tangent plane idea first introduced in [5] in order to remove the nonlinear solver required in [10]. This is achieved by solving for the time derivative of the magnetization $\partial_t \mathbf{m}$ instead of the magnetization itself. The constraint $|\mathbf{m}| = 1$ translates to the orthogonality $\partial_t \mathbf{m} \cdot \mathbf{m} = 0$ almost everywhere. In the discretization, this can only hold nodewise [5] or in an average sense [2]. As an attractive side effect, the discrete approximations still obey the physical energy decay. This is in contrast to other higher-order methods for LLG in [26, 24, 14] (without strong error analysis) and [19, 6] which do not satisfy discrete energy bounds.

While the higher-order convergence depends on certain regularity assumptions on the exact solution (see [16] for existence proofs of smooth solutions under a smallness condition on the initial data), it was shown in [2] that even without regularity beyond H^1 one can still get weak convergence (a consequence of the energy decay) and thus is at least as good as traditional first order schemes.

The aim of this work is to exploit the higher-order time-stepping schemes developed in [2] in order to obtain a heuristic estimate of the temporal approximation error. This will allow us to construct an adaptive time-stepping scheme for LLG. Combined with gradient recovery estimators for the spatial error, we derive a fully adaptive integrator for LLG. While a convergence proof seems out of reach for the moment, we demonstrate in several experiments the effectiveness of the method.

In Section 1.2 we introduce the initial value problem in its strong form. In Section 2 we present the spatial discretisation that is based on the standard higher order conforming finite element

method and a Lagrangian setting to cope with the nonlinear constraint. For the time stepping, we use a collocation method based on the BDF- k method (for $k = 1, 2$) with a predictor based on extrapolation. In the uniform setting this method has proved to converge with optimal rates in [2]. In Section 3 we formulate the BDF method with variable time steps. Furthermore we present approximations to higher order derivatives with finite differences in order to estimate the truncation error of a given order. This allows to suggest new time step sizes and likewise also orders. In Theorem 1 we show that under regularity requirements, restricted time step changes and a certain size of the damping term, the time-adaptive method satisfies an energy bound. In Sections 3 and 4 we formulate our adaptive method and conclude in Section 6 with some numerical experiments.

1.1. Notation. Let $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, be a bounded domain with polygonally boundary. By $L^2(\Omega)$ we denote the usual Lebesgue spaces with scalar product $(v, w)_\Omega = \int_\Omega vw$ and norm $\|v\|_{L^2(\Omega)} = (v, v)_\Omega^{1/2}$. Vector valued functions are in $L^2(\Omega)^d$ if they are componentwise in $L^2(\Omega)$ and the scalar product is defined as $(\mathbf{v}, \mathbf{w})_\Omega = \int_\Omega \mathbf{v} \cdot \mathbf{w}$. $H^1(\Omega)$ denotes the space of functions in $L^2(\Omega)$ with weak derivatives in $L^2(\Omega)$. A norm on $H^1(\Omega)$ is given by $(\|v\|_{L^2(\Omega)}^2 + \|\nabla v\|_{L^2(\Omega)}^2)^{1/2}$.

In the following we work with vector valued functions $\mathbf{m} = [m_1, m_2, m_3] : (0, T) \times \Omega \rightarrow \mathbb{R}^3$. For convenience we use the notation $\mathbf{m}(t) = \mathbf{m}(t, \cdot) : \Omega \rightarrow \mathbb{R}^3$. For $\nabla \mathbf{m}$ we let $\|\nabla \mathbf{m}\|_{L^2(\Omega)}^2 = \int_\Omega \nabla \mathbf{m} : \nabla \mathbf{m}$, where $\cdot :$ denotes the Frobenius product of two matrices. For vectors or tensors $|\cdot|$ denotes the euclidean norm or the Frobenius norm.

1.2. The Landau–Lifshitz–Gilbert-equation. The strong form of the LLG equations is to seek $\mathbf{m} : [0, T] \times \bar{\Omega}$ such that

$$(1.1) \quad \alpha \partial_t \mathbf{m} + \mathbf{m} \times \partial_t \mathbf{m} = \mathbf{P}(\mathbf{m}) \mathbf{H}_{\text{eff}}(\mathbf{m}) \quad \text{on } (0, T) \times \Omega,$$

$$(1.2) \quad \partial_n \mathbf{m} = \mathbf{0} \quad \text{on } (0, T) \times \partial \Omega,$$

$$(1.3) \quad \mathbf{m}(0) = \mathbf{m}^0 \quad \text{in } \Omega,$$

with a given vector field $\mathbf{m}^0 : \Omega \rightarrow \mathbb{R}^3$ of unit length, i. e., $|\mathbf{m}^0| = 1$. \mathbf{P} is the projection operator $\mathbf{P}(\mathbf{m}) = \mathbf{Id} - \frac{\mathbf{m} \otimes \mathbf{m}}{|\mathbf{m}|^2}$ for $\mathbf{m} \neq \mathbf{0}$. In this paper we let the effective field be given as

$$(1.4) \quad \mathbf{H}_{\text{eff}}(\mathbf{m}) = C_e \Delta \mathbf{m} + \mathbf{H}_{\text{ext}}.$$

Formally, we can derive from (1.1), by multiplying with \mathbf{m} , that $\partial_t |\mathbf{m}(t)|^2 = 0$, hence $|\mathbf{m}(t)| = |\mathbf{m}^0| = 1$. Multiplying with $\partial_t \mathbf{m}$ and integrating will lead to the energy inequality

$$(1.5) \quad \begin{aligned} \mathcal{E}(t) &= C_e \|\nabla \mathbf{m}(t)\|_{L^2(\Omega)}^2 + \frac{\alpha}{2} \int_0^t \|\partial_t \mathbf{m}(s)\|_{L^2(\Omega)}^2 \, ds \\ &\leq C_e \|\nabla \mathbf{m}(0)\|_{L^2(\Omega)}^2 + \frac{1}{2\alpha} \int_0^t \|\mathbf{H}_{\text{ext}}(s)\|_{L^2(\Omega)}^2 \, ds \end{aligned}$$

for all $t \in [0, T]$.

2. DISCRETISATION OF THE LANDAU–LIFSHITZ–GILBERT-EQUATION

2.1. Weak formulation. The derivation of a weak form can be simplified if one uses test functions $\varphi \in C^\infty(\Omega)^3$ with $\mathbf{m}(t) \cdot \varphi = 0$ at time $t > 0$, since it will allow us later to replace $\mathbf{P}(\mathbf{m})$ in (1.1) by \mathbf{Id} . Thus we define the solution dependent constraint space by

$$\mathbb{T}(\mathbf{m}(t)) = \{\varphi \in H^1(\Omega)^3 \mid \mathbf{m}(t) \cdot \varphi = 0\}.$$

For ease of notation we will omit the argument t in the sequel and arrive formally at the following weak equation for \mathbf{m}

$$\alpha(\partial_t \mathbf{m}, \varphi)_\Omega + (\mathbf{m} \times \partial_t \mathbf{m}, \varphi)_\Omega + C_e(\nabla \mathbf{m}, \nabla \varphi)_\Omega = (\mathbf{H}_{\text{ext}}, \varphi)_\Omega \quad \text{for all } \varphi \in \mathbb{T}(\mathbf{m}).$$

Existence of weak solutions $\mathbf{m} \in H^1(\Omega)^3$ is well-known (see, e.g., [27]) and follows with a standard Galerkin–Fadeo approximation and compactness arguments. Uniqueness of weak

solutions is open in general and there are non-uniqueness results for the related harmonic map heat flow [12]. Note that in case one of the weak solutions is also a strong solution, we have uniqueness [13]. In terms of regularity, it is suspected that even smooth initial conditions can lead to blow up of $\nabla \mathbf{m}$ in $L^\infty(\Omega)^{3,3}$ in finite time, however, no proof is known. For LLG on smooth bounded domains with Neumann boundary conditions, smooth initial data close to constants lead to arbitrarily smooth solutions [16].

2.2. Spatial discretisation. Since Ω is assumed to be polygonally bounded, we let \mathcal{K}_h be a regular triangulation such that $\bar{\Omega} = \bigcup\{K \in \mathcal{K}_h\}$. On \mathcal{K}_h we consider the finite element space $W_h \subset H^1(\Omega)$ of constant degree $p \geq 1$. For vector fields we then use $\mathbf{V}_h = W_h^3 \subset H^1(\Omega)^3$. It looks quite natural to correspondingly define the discrete constraint space by

$$\mathbb{T}_h^s(\mathbf{m}_h) = \{\boldsymbol{\varphi}_h \in \mathbf{V}_h \mid \mathbf{m}_h \cdot \boldsymbol{\varphi}_h = 0 \text{ at every node}\}$$

for some $\mathbf{m}_h \in H^1(\Omega)^3$. Since this choice leads to problems with higher order convergence [2, Rem. 2.2] the method for which this could be achieved in [2] is based on the weaker constraint space

$$\mathbb{T}_h(\mathbf{m}_h) = \mathbb{T}_h^w(\mathbf{m}_h) = \{\boldsymbol{\varphi}_h \in \mathbf{V}_h \mid (\mathbf{m}_h \cdot \boldsymbol{\varphi}_h, \psi_h)_\Omega = 0 \text{ for all } \psi_h \in W_h\}.$$

It is then straightforward to define the semi-discrete solution $\mathbf{m}_h : [0, T] \rightarrow \mathbf{V}_h$ by

$$\begin{aligned} \alpha(\partial_t \mathbf{m}_h, \boldsymbol{\varphi}_h)_\Omega + (\mathbf{m}_h \times \partial_t \mathbf{m}_h, \boldsymbol{\varphi}_h)_\Omega + C_e(\nabla \mathbf{m}_h, \nabla \boldsymbol{\varphi}_h)_\Omega &= (\mathbf{H}_{\text{ext}}, \boldsymbol{\varphi}_h)_\Omega \\ &\text{for all } \boldsymbol{\varphi}_h \in \mathbb{T}_h(\mathbf{m}_h). \end{aligned}$$

2.3. Temporal discretisation. We complete the discretisation by defining a time-stepping method. For this we let $0 = t_0 < \dots < t_{N_T} = T$ be a decomposition of the interval $[0, T]$. The finite element space at a time $t_n \in [0, T]$ may change due to regular refinement (and derefinement) of the preceding one and is denoted by \mathbf{V}_h^n . We want to set up an equation for the approximation $\mathbf{m}^n \approx \mathbf{m}(t_n)$ at time t_n . Further we introduce the unknown $\mathbf{v}^n \approx \partial_t \mathbf{m}(t_n)$ and assume that we can impose a relation between \mathbf{v}_h^n and \mathbf{m}_h^{n-j} for a set of indices $j \geq 0$. For example, if we assume a linear time-dependence of \mathbf{m}_h , we can state $\mathbf{m}_h^n = \mathbf{m}_h^{n-1} + \tau_n \mathbf{v}_h^n$. Thus the resulting equation

$$\alpha(\mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + (\mathbf{m}_h^n \times \mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + C_e(\nabla \mathbf{m}_h^n, \nabla \boldsymbol{\varphi}_h)_\Omega = (\mathbf{H}_{\text{ext}}^n, \boldsymbol{\varphi}_h)_\Omega \quad \text{for all } \boldsymbol{\varphi}_h \in \mathbb{T}_h(\mathbf{m}_h^n)$$

can be seen as a nonlinear equation for \mathbf{v}_h^n from which \mathbf{m}_h^n could then easily be obtained. However, we have to note that the test space also depends on \mathbf{v}_h^n . This approach is called *tangent plane scheme* [23].

The following linearisation has turned out to be useful: construct an approximation (or *predictor*) $\widehat{\mathbf{m}}_h^n$ to \mathbf{m}_h^n and then define \mathbf{m}_h^n by

$$\alpha(\mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + (\widehat{\mathbf{m}}_h^n \times \mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + C_e(\nabla \widehat{\mathbf{m}}_h^n, \nabla \boldsymbol{\varphi}_h)_\Omega = (\mathbf{H}_{\text{ext}}^n, \boldsymbol{\varphi}_h)_\Omega \quad \text{for all } \boldsymbol{\varphi}_h \in \mathbb{T}_h(\widehat{\mathbf{m}}_h^n)$$

which is now a linear problem for \mathbf{v}_h^n . There is still the problem to cope with the constraint, for example, one might construct a local basis [2, Sect. 2.2]. Here we will add the constraint as a separate weak equation. Thus we introduce the Lagrange variable $\lambda_h^n \in W_h$ and state the saddle point problem

$$\begin{aligned} (2.1) \quad \alpha(\mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + (\widehat{\mathbf{m}}_h^n \times \mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + C_e(\nabla \widehat{\mathbf{m}}_h^n, \nabla \boldsymbol{\varphi}_h)_\Omega \\ + (\widehat{\mathbf{m}}_h^n \cdot \boldsymbol{\varphi}_h, \lambda_h^n)_\Omega &= (\mathbf{H}_{\text{ext}}^n, \boldsymbol{\varphi}_h)_\Omega \quad \text{for all } \boldsymbol{\varphi}_h \in \mathbf{V}_h^n, \\ (\widehat{\mathbf{m}}_h^n \cdot \mathbf{v}_h^n, \psi_h)_\Omega &= 0 \quad \text{for all } \psi_h \in W_h^n. \end{aligned}$$

Note that the analysis of [2] is based on this formulation. It is straightforward to show that the discrete formulation is well-posed and leads to a unique discrete solution \mathbf{v}_h^n .

2.4. BDF time stepping. For the time stepping we now use especially the BDF (*backward differencing*) scheme of order $k \in \mathbb{N}$. It provides a general relation

$$(2.2) \quad \xi_k \mathbf{m}_h^n = \tilde{\Phi}_k(\mathbf{m}_h^{n-1}, \dots, \mathbf{m}_h^{n-k}) + \tau_n \mathbf{v}_h^n$$

for $n \geq k$ and an affine linear mapping $\tilde{\Phi}_k : (\mathbb{R}^3)^k \rightarrow \mathbb{R}^3$. The case $k = 1$ has been given as an example in the previous Section 2.3, more details will be provided in Section 3. We insert this relation into (2.1) and get the following linear equation for \mathbf{v}_h^n

$$(2.3) \quad \begin{aligned} \alpha(\mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + (\widehat{\mathbf{m}}_h^n \times \mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + \beta_k \tau_n (\nabla \mathbf{v}_h^n, \nabla \boldsymbol{\varphi}_h)_\Omega \\ + (\widehat{\mathbf{m}}_h^n \cdot \boldsymbol{\varphi}_h, \lambda_h^n)_\Omega = \langle \mathbf{f}^n, \boldsymbol{\varphi}_h \rangle_\Omega \quad \text{for all } \boldsymbol{\varphi}_h \in \mathbf{V}_h^n, \\ (\widehat{\mathbf{m}}_h^n \cdot \mathbf{v}_h^n, \psi_h)_\Omega = 0 \quad \text{for all } \psi_h \in W_h^n, \end{aligned}$$

with $\beta_k = C_e/\xi_k$ and the new right hand side

$$(2.4) \quad \langle \mathbf{f}^n, \boldsymbol{\varphi}_h \rangle_\Omega := (\mathbf{H}_{\text{ext}}^n, \boldsymbol{\varphi}_h)_\Omega - \beta_k (\nabla \tilde{\Phi}_k(\mathbf{m}_h^{n-1}, \dots, \mathbf{m}_h^{n-k}), \nabla \boldsymbol{\varphi}_h)_\Omega.$$

\mathbf{m}_h^n is then defined by (2.2). Note that a discrete solution \mathbf{v}_h^n exists since the form is coercive on $\mathbb{T}_h(\widehat{\mathbf{m}}_h^n)$.

As an example for a predictor $\widehat{\mathbf{m}}_h^n$ one can take extrapolation of order $k - 1$, i.e., seek the unique polynomial $\mathbf{q} \in \mathbb{P}_{k-1}$ with $\mathbf{q}(t_{n-1-j}) = \mathbf{m}_h^{n-1-j}$ for $j = 0, \dots, k - 1$, and set $\widehat{\mathbf{m}}_h^n = \mathbf{q}(t_n)/|\mathbf{q}(t_n)|$. This problem has been analyzed in [2] and it was possible to prove, for uniform step size τ , unconditional convergence in time up to order two [2, Thm. 3.1] and convergence up to order five in h with some restrictions on α [2, Thm. 3.4].

2.5. The uniform discretisation. We consider a fixed (almost) uniform mesh of meshsize h with a finite element method of polynomial order p and in time BDF- k for $k = 1, \dots, 5$. We will make regularity requirements for weak solutions as in [2, (3.2)]

$$(2.5) \quad \begin{aligned} \mathbf{m} \in C^{k+1}([0, T], L^\infty(\Omega)^3) \cap C^1([0, T], W^{p+1, \infty}(\Omega)^3), \\ \Delta \mathbf{m} + \mathbf{H}_{\text{ext}} \in C^0([0, T], W^{p+1, \infty}(\Omega)^3). \end{aligned}$$

Then it has been proved in [2] that for sufficiently small τ and h the discrete solutions exist and it holds

$$(2.6) \quad \|\nabla(\mathbf{m}(t_n) - \mathbf{m}_h^n)\|_\Omega \leq C(\tau^k + h^p)$$

for $t_n = n\tau \leq T$. In case $k \in \{1, 2\}$ one needs the additional condition $\tau \leq C\sqrt{h}$ [2, Thm. 3.1], while in case $k \in \{3, 4, 5\}$ one requires $p \geq 2$, $\tau \leq Ch$, and $\alpha \geq \alpha_k > 0$ for some numbers α_k [2, Thm. 3.4].

3. ADAPTIVITY IN TIME

3.1. The variable step BDF method. To describe the BDF method of order k for $k \in \mathbb{N}$ we find it convenient to work on the grid

$$[0, t_1, \dots, t_k] = [0, \tau_1, \tau_1 + \tau_2, \tau_1 + \dots + \tau_k]$$

for positive step sizes τ_j , $j = 1, \dots, k$. This can afterwards be translated to a mesh of the form t_{n-k}, \dots, t_n (for $n \geq k$) by index shift. We will use abbreviations

$$\tau_{12} = \tau_1 + \tau_2, \quad \tau_{23} = \tau_2 + \tau_3, \quad \tau_{122} = \tau_1 + 2\tau_2, \quad \dots$$

and let $\tau = \max\{\tau_1, \dots, \tau_k\}$. The definition of a variable step BDF method to solve the ordinary differential equation (ODE) $\mathbf{y}' = \mathbf{F}(t, \mathbf{y})$, $\mathbf{y} : [0, T] \rightarrow \mathbb{R}^N$, $\mathbf{F} : [0, T] \times \mathbb{R}^N \rightarrow \mathbb{R}^N$, is as follows: Given $\{\mathbf{y}^j\}_{j=0}^{k-1}$, determine \mathbf{y}^k (approximating $\mathbf{y}(t_k)$) such that there is a polynomial $\mathbf{q} \in \mathbb{P}_k^m$ with

$$\mathbf{q}(t_j) = \mathbf{y}^j, \quad j = 0, \dots, k-1, \quad \mathbf{q}'(t_k) = \mathbf{F}(t_k, \mathbf{y}^k).$$

This leads to a nonlinear equation of the form

$$(3.1) \quad \Phi_k(\mathbf{y}) = \xi_k \mathbf{y} - \tilde{\Phi}_k(\mathbf{y}^{k-1}, \dots, \mathbf{y}^0) - \tau_k \mathbf{F}(t_k, \mathbf{y}) = \mathbf{0},$$

with $\xi_k > 0$, to determine \mathbf{y}^k . For $k = 1, 2, 3$ we obtain the explicit formulae

$$(3.2) \quad \Phi_1(\mathbf{y}) = \mathbf{y} - \mathbf{y}^0 - \tau_1 \mathbf{F}(t_1, \mathbf{y}),$$

$$(3.3) \quad \Phi_2(\mathbf{y}) = \frac{\tau_{122}}{\tau_{12}} \mathbf{y} - \frac{\tau_{12}}{\tau_1} \mathbf{y}^1 + \frac{\tau_2^2}{\tau_1 \tau_{12}} \mathbf{y}^0 - \tau_2 \mathbf{F}(t_2, \mathbf{y}).$$

If we compare this to (2.2), we can get the values for ξ_k and expressions for $\tilde{\Phi}_k(\mathbf{y}^{k-1}, \dots, \mathbf{y}^0)$. These formulas agree with those in the case of constant τ [22, Ch. V]. Recall that for $k = 1$ we get the implicit Euler method.

We will, in the provided notation and for later use, formulate consistent approximations for second and third order derivatives

$$(3.4) \quad \partial_\tau^2 \mathbf{y}(t_2) = 2 \left(\frac{1}{\tau_{12} \tau_2} \mathbf{y}_2 - \frac{1}{\tau_1 \tau_2} \mathbf{y}_1 + \frac{1}{\tau_1 \tau_{12}} \mathbf{y}_0 \right),$$

$$(3.5) \quad \partial_\tau^3 \mathbf{y}(t_3) = 6 \left(\frac{1}{\tau_{123} \tau_{23} \tau_3} \mathbf{y}_3 - \frac{1}{\tau_{12} \tau_2 \tau_3} \mathbf{y}_2 + \frac{1}{\tau_1 \tau_2 \tau_{23}} \mathbf{y}_1 - \frac{1}{\tau_1 \tau_{12} \tau_{123}} \mathbf{y}_0 \right).$$

3.2. Time step selection. In the beginning we will always start with the first order method, i.e., the implicit Euler method (Φ_1). Since the local truncation error LTE at a time t is known to be $1/2 |\mathbf{y}''(t)| \tau^2$, we suggest as a next time step

$$(3.6) \quad \tau_n^{[1]} := \left(\frac{2 \text{TOL}}{|\mathbf{y}''(t_n)|} \right)^{1/2},$$

with the idea to get $\text{TOL} \geq \text{LTE} \approx |\mathbf{y}''(t_n)| \tau_n^2$. In order to evaluate this we use

$$(3.7) \quad \mathbf{y}''(t_n) = \partial_t \mathbf{F}(t_n, \mathbf{y}(t_n)) + \partial_{\mathbf{y}} \mathbf{F}(t_n, \mathbf{y}(t_n)) \mathbf{F}(t_n, \mathbf{y}(t_n)),$$

but alternatively, we may use the finite difference approximation $\partial_\tau^2 \mathbf{y}(t_n)$ (3.4) for $n \geq 2$.

The next order method is given by Φ_2 and the truncation error in the uniform case is known to be $1/3 |\mathbf{y}'''(t)| \tau^3$. In the case of variable step sizes we use Taylor expansion of $\Phi_2(\mathbf{y}(t))$ in t_n and get the local truncation error $1/6 \tau_{n-1,n} \tau_n^2 |\mathbf{y}'''(t_n)|$ and from this the idea to let

$$(3.8) \quad \tau_n^{[2]} := \left(\frac{6 \text{TOL}}{\tau_{n-1,n} |\mathbf{y}'''(t_n)|} \right)^{1/2}.$$

At this point we will rely on the finite difference approximation $\partial_\tau^3 \mathbf{y}(t_n)$ (3.5) for $n \geq 3$.

Every time step choice will be subject to a restriction compared to the previous step in the form

$$\begin{aligned} \tau_n^* &= \max \{ \tau_n^{[k]} \mid k_{\min} \leq k \leq k_{\max} \} = \tau_n^{[k_n^*]}, \\ \tau_{n+1} &= \max \left\{ \tau_{\min}, \frac{1}{2} \tau_n, \min \{ \tau_n^*, \sqrt{2} \tau_n, \tau_{\max} \} \right\}, \end{aligned}$$

where $k_{\min} \leq k_{\max} \in \mathbb{N}$ determine the set of admissible orders and k_n^* notes which order provides the maximal step size. τ_{\min} and τ_{\max} are prescribed lower and upper bounds for the time step size. Note that values $|\mathbf{y}''(t_n)|, |\mathbf{y}'''(t_n)|$ close or equal to 0 are of no harm since we have set a value τ_{\max} . If we also allow order selection, we will choose $k_{n+1} = \max \{ k_{\min}, k_n - 1, \min \{ k_n^*, k_n + 1, k_{\max} \} \}$. In case the nonlinear equations are solved with Newton's method, we might prevent the number of steps to be beyond, say, 4 and thus prohibit enlargement of τ_n in case this number is reached.

3.3. Application to LLG. We iteratively solve (2.3), starting from a normalized initial condition \mathbf{m}_h^0 that approximates \mathbf{m}^0 . The BDF-2 method with variable stepsize is then defined by Sections 2, 3.1, and 3.2 in the case $k = 2$, taking $L^2(\Omega)$ -norms for the difference quotients in time. Then we use extrapolation of order $k - 1$ to define the prediction $\widehat{\mathbf{m}}_h^n$ [2, (2.1)]. An important statement about the uniform BDF-2 method for LLG is the validity of the energy inequality [2, Prop. 3.1]. We can derive a corresponding result for BDF-2 with variable stepsize under the condition that the coarsening factor is limited, and α has a positive lower bound.

Theorem 1 (Energy inequality for orders $k = 1, 2$). *Consider the discretization (2.3) of the LLG equation (1.1) for $k \in \{1, 2\}$ with finite elements of polynomial degree $p \geq 1$. Let, for $k = 2$, the time steps fulfill $\tau_n/\tau_{n-1} \leq \kappa_0 \leq \sqrt{2} + 1$ and $\alpha \geq \alpha_2 > 0$ for some α_2 that depends on κ_0 . Then, the numerical solution satisfies the following discrete energy inequality: for $n > k$ it holds*

$$\gamma_k^- \|\nabla \mathbf{m}_h^n\|_{L^2(\Omega)}^2 + \frac{1}{2} \alpha \sum_{j=k}^n \tau_j \|\dot{\mathbf{m}}_h^j\|_{L^2(\Omega)}^2 \leq \gamma_k^+ \sum_{i=0}^{k-1} \|\nabla \mathbf{m}_h^i\|_{L^2(\Omega)}^2 + \frac{1}{2\alpha} \sum_{j=k}^n \tau_j \|\mathbf{H}_{\text{ext}}(t_j)\|_{L^2(\Omega)}^2$$

for some positive constants γ_k^-, γ_k^+ when the solution fulfills the regularity requirements (2.5).

Proof. We first note that the BDF-formula provides coefficients δ_j , $j = 0, \dots, k$, such that

$$\tau_n \dot{\mathbf{m}}_h^n = \sum_{j=0}^k \delta_j \mathbf{m}_h^{n-j},$$

see Section 3.1. As in [2] we choose $\eta > 0$, write down the weak equation for $\mathbf{m}_h^n - \eta \mathbf{m}_h^{n-1}$ and test it with \mathbf{m}_h^n . The main point is a lower bound for $(\nabla(\mathbf{m}_h^n - \eta \mathbf{m}_h^{n-1}), \nabla \dot{\mathbf{m}}_h^n)_\Omega$. If we let

$$\delta(s) := \sum_{i=0}^k \delta_i s^i,$$

and if we can show that $\text{Re}(\delta(s)/(1-\eta s)) > 0$ for all complex s with $|s| < 1$, then [2, Lemma A.2] guarantees the existence of the required constants. The case $k = 1$ is clear. In the case $k = 2$, we let $\tau_n = \kappa \tau$ and $\tau_{n-1} = \tau$ for some $\kappa > 0$. From (3.3) we get

$$\delta_0 = \frac{1 + 2\kappa}{1 + \kappa}, \quad \delta_1 = -(1 + \kappa), \quad \delta_2 = \frac{\kappa^2}{1 + \kappa}.$$

We see immediately that δ has two zeros $s_1 = 1$ and $s_2 = (1 + 2\kappa)/\kappa^2 > 1$ as long as $\kappa < 1 + \sqrt{2}$. Thus, we have

$$\frac{\delta(s)}{1 - \eta s} = \frac{\delta_2(s-1)(s-s_2)}{1 - \eta s} = \frac{\delta_2(s-1)(s-s_2)(1 - \eta \bar{s})}{|1 - \eta s|}.$$

Clearly, the choice $\eta = 1/s_2$ would suffice to ensure $\text{Re}(\delta(s)/(1 - \eta s)) > 0$ for $|s| < 1$. However, we obtain a sharper bound by resolving the inequality

$$\text{Re}((s-1)(s-s_2)(1 - \eta \bar{s})) > 0 \quad \text{for } |s| < 1.$$

Considering the edge case $|s| = 1$ leads to the following quadratic inequality for the real part s_r of s (the cubic terms cancel)

$$2s_r^2 - (1 + \eta)(1 + s_2)s_r + (1 + \eta)s_2 + \eta - 1 > 0 \quad \text{for all } -1 < s_r < 1.$$

For given $0 < \kappa < 1 + \sqrt{2}$ this can be fulfilled if

$$\eta > \frac{3 - s_2}{s_2 + 1} = \frac{3\kappa^2 - 2\kappa - 1}{\kappa^2 + 2\kappa + 1}.$$

For $\kappa \leq 1$ we can take $\eta = 0$ and for $\kappa \in (1, 1 + \sqrt{2})$ we find $\eta < 1$. Note that this bound dates back to [21] but was proved with a different method.

Let us first take for simplicity $\mathbf{H}_{\text{ext}}^n = \mathbf{0}$. We proceed in recalling the weak equations for \mathbf{v}_h^n and \mathbf{v}_h^{n-1} , for $n \geq 2$,

$$\begin{aligned} \alpha(\mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + (\widehat{\mathbf{m}}_h^n \times \mathbf{v}_h^n, \boldsymbol{\varphi}_h)_\Omega + C_e(\nabla \mathbf{m}_h^n, \nabla \boldsymbol{\varphi}_h)_\Omega &= 0 \quad \text{for all } \boldsymbol{\varphi}_h \in \mathbb{T}_h(\widehat{\mathbf{m}}_h^n), \\ \alpha(\mathbf{v}_h^{n-1}, \boldsymbol{\varphi}_h)_\Omega + (\widehat{\mathbf{m}}_h^{n-1} \times \mathbf{v}_h^{n-1}, \boldsymbol{\varphi}_h)_\Omega + C_e(\nabla \mathbf{m}_h^{n-1}, \nabla \boldsymbol{\varphi}_h)_\Omega &= 0 \quad \text{for all } \boldsymbol{\varphi}_h \in \mathbb{T}_h(\widehat{\mathbf{m}}_h^{n-1}). \end{aligned}$$

In order to derive an equation for $\mathbf{v}_h^n - \eta \mathbf{v}_h^{n-1}$ we use the test function $\mathbf{v}_h^n \in \mathbb{T}_h(\widehat{\mathbf{m}}_h^n)$ in the first equation and $\widehat{\mathbf{P}}_h^{n-1} \mathbf{v}_h^n$ with $\widehat{\mathbf{P}}_h^{n-1} := \mathbf{P}(\widehat{\mathbf{m}}_h^{n-1}) \in \mathbb{T}_h(\widehat{\mathbf{m}}_h^{n-1})$ in the second equation. This latter term will be written in the form $\widehat{\mathbf{P}}_h^{n-1} \mathbf{v}_h^n = \mathbf{v}_h^n - (\widehat{\mathbf{P}}_h^n - \widehat{\mathbf{P}}_h^{n-1}) \mathbf{v}_h^n = \mathbf{v}_h^n - \mathbf{p}_h^n$. Insertion of these test functions results in

$$\begin{aligned} \alpha(\mathbf{v}_h^n, \mathbf{v}_h^n)_\Omega + C_e(\nabla \mathbf{m}_h^n, \nabla \mathbf{v}_h^n)_\Omega &= 0, \\ \alpha(\mathbf{v}_h^{n-1}, \mathbf{v}_h^n)_\Omega + C_e(\nabla \mathbf{m}_h^{n-1}, \nabla \mathbf{v}_h^n)_\Omega \\ &= \alpha(\mathbf{v}_h^{n-1}, \mathbf{p}_h^n)_\Omega - (\widehat{\mathbf{m}}_h^{n-1} \times \mathbf{v}_h^{n-1}, \mathbf{v}_h^n - \mathbf{p}_h^n)_\Omega + C_e(\nabla \mathbf{m}_h^{n-1}, \nabla \mathbf{p}_h^n)_\Omega \\ &= \alpha(\mathbf{v}_h^{n-1}, \mathbf{p}_h^n)_\Omega - (\widehat{\mathbf{m}}_h^{n-1} \times \mathbf{v}_h^{n-1}, \mathbf{v}_h^n - \sigma \mathbf{v}_h^{n-1} - \mathbf{p}_h^n)_\Omega + C_e(\nabla \mathbf{m}_h^{n-1}, \nabla \mathbf{p}_h^n)_\Omega, \end{aligned}$$

where we can choose either $\sigma = 0$ or $\sigma = 1$. By subtraction we get

$$(3.9) \quad \begin{aligned} \alpha(\mathbf{v}_h^n - \eta \mathbf{v}_h^{n-1}, \mathbf{v}_h^n)_\Omega + C_e(\nabla(\mathbf{m}_h^n - \eta \mathbf{m}_h^{n-1}), \nabla \mathbf{v}_h^n)_\Omega \\ = -\eta \alpha(\mathbf{v}_h^{n-1}, \mathbf{p}_h^n)_\Omega + \eta (\widehat{\mathbf{m}}_h^{n-1} \times \mathbf{v}_h^{n-1}, \mathbf{v}_h^n - \sigma \mathbf{v}_h^{n-1} - \mathbf{p}_h^n)_\Omega - \eta C_e(\nabla \mathbf{m}_h^{n-1}, \nabla \mathbf{p}_h^n)_\Omega. \end{aligned}$$

By construction of η in the first part of the proof we have for the left hand side of (3.9) the lower bound [2, Lem. 8.1(proof)]

$$\begin{aligned} \alpha(\mathbf{v}_h^n - \eta \mathbf{v}_h^{n-1}, \mathbf{v}_h^n)_\Omega + C_e(\nabla(\mathbf{m}_h^n - \eta \mathbf{m}_h^{n-1}), \nabla \mathbf{v}_h^n)_\Omega \\ \geq \left(1 - \frac{\eta}{2}\right) \alpha \|\mathbf{v}_h^n\|_\Omega^2 - \frac{\eta}{2} \alpha \|\mathbf{v}_h^{n-1}\|_\Omega^2 + \frac{1}{\tau_n} (\|\nabla \mathbf{E}_h^n\|_{\mathcal{G}, \Omega}^2 - \|\nabla \mathbf{E}_h^{n-1}\|_{\mathcal{G}, \Omega}^2), \end{aligned}$$

where $\mathbf{E}^n = [\mathbf{v}_h^{n-1}, \mathbf{v}_h^n]$ and the norm is taken with some symmetric positive definite $\mathbf{G} \in \mathbb{R}^{2,2}$ whose eigenvalues γ_2^\pm depend on η . For details see [2, Appendix]. In the following we give an upper bound for the right hand side of (3.9).

To simplify the following terms we will introduce the abbreviations

$$\begin{aligned} \mu_{0,n} &= \|\widehat{\mathbf{m}}_h^n - \widehat{\mathbf{m}}_h^{n-1}\|_\Omega, \quad \mu_{1,n} = \|\nabla(\widehat{\mathbf{m}}_h^n - \widehat{\mathbf{m}}_h^{n-1})\|_\Omega, \quad \nu_{0,n} = \min \left\{ \frac{\|\mathbf{v}_h^n - \mathbf{v}_h^{n-1}\|_\Omega}{\|\mathbf{v}_h^n\|_\Omega}, 1 \right\}, \\ \Gamma_n &= \max \left\{ \|\nabla \widehat{\mathbf{m}}_h^n\|_{\infty, \Omega}, \|\nabla \widehat{\mathbf{m}}_h^{n-1}\|_{\infty, \Omega} \right\}, \quad \Lambda(\mathbf{v}_h^n) = \frac{\|\nabla \mathbf{v}_h^n\|_\Omega}{\|\mathbf{v}_h^n\|_\Omega}. \end{aligned}$$

As a further preparation we provide bounds for \mathbf{p}_h^n

$$\begin{aligned} \|\mathbf{p}_h^n\|_\Omega &\leq 2 \|\widehat{\mathbf{m}}_h^n - \widehat{\mathbf{m}}_h^{n-1}\|_\Omega \|\mathbf{v}_h^n\|_\Omega \leq 2 \mu_{0,n} \|\mathbf{v}_h^n\|_\Omega, \\ \|\nabla \mathbf{p}_h^n\|_\Omega &\leq 2 \Gamma_n \|\widehat{\mathbf{m}}_h^n - \widehat{\mathbf{m}}_h^{n-1}\|_\Omega \|\mathbf{v}_h^n\|_\Omega + 2 \|\nabla(\widehat{\mathbf{m}}_h^n - \widehat{\mathbf{m}}_h^{n-1})\|_\Omega \|\mathbf{v}_h^n\|_\Omega \\ &\quad + 2 \|\widehat{\mathbf{m}}_h^n - \widehat{\mathbf{m}}_h^{n-1}\|_\Omega \|\nabla \mathbf{v}_h^n\|_\Omega \\ &\leq 2 \Gamma_n \mu_{0,n} \|\mathbf{v}_h^n\|_\Omega + 2 \mu_{1,n} \|\mathbf{v}_h^n\|_\Omega + 2 \mu_{0,n} \Lambda(\mathbf{v}_h^n) \|\mathbf{v}_h^n\|_\Omega \\ &\leq 2 (\mu_{0,n} \Gamma_n + \mu_{1,n} + \mu_{0,n} \Lambda(\mathbf{v}_h^n)) \|\mathbf{v}_h^n\|_\Omega. \end{aligned}$$

Thus we get the following estimate for the three terms on the right hand side in (3.9). For the first term

$$\eta \alpha(\mathbf{v}_h^{n-1}, \mathbf{p}_h^n)_\Omega \leq 2 \eta \alpha \mu_{0,n} \|\mathbf{v}_h^{n-1}\|_\Omega \|\mathbf{v}_h^n\|_\Omega \leq \mu_{0,n} \eta \alpha (\|\mathbf{v}_h^{n-1}\|_\Omega^2 + \|\mathbf{v}_h^n\|_\Omega^2).$$

For the second term we can either choose

$$\begin{aligned} & \eta(\widehat{\mathbf{m}}_h^{n-1} \times \mathbf{v}_h^{n-1}, \mathbf{v}_h^n - \mathbf{v}_h^{n-1} - \mathbf{p}_h^n)_\Omega \\ & \leq \eta \|\mathbf{v}_h^{n-1}\|_\Omega (\|\mathbf{v}_h^n - \mathbf{v}_h^{n-1}\|_\Omega + 2\mu_{0,n} \|\mathbf{v}_h^n\|_\Omega) \\ & \leq \frac{1}{2} \left(\frac{\|\mathbf{v}_h^n - \mathbf{v}_h^{n-1}\|_\Omega}{\|\mathbf{v}_h^n\|_\Omega} + 2\mu_{0,n} \right) \eta (\|\mathbf{v}_h^{n-1}\|_\Omega^2 + \|\mathbf{v}_h^n\|_\Omega^2) \end{aligned}$$

for $\sigma = 1$, or

$$\begin{aligned} & \eta(\widehat{\mathbf{m}}_h^{n-1} \times \mathbf{v}_h^{n-1}, \mathbf{v}_h^n - \mathbf{v}_h^{n-1} - \mathbf{p}_h^n)_\Omega \\ & \leq \frac{1}{2} (1 + 2\mu_{0,n}) \eta (\|\mathbf{v}_h^{n-1}\|_\Omega^2 + \|\mathbf{v}_h^n\|_\Omega^2) \end{aligned}$$

for $\sigma = 0$. Thus we can take

$$\eta(\widehat{\mathbf{m}}_h^{n-1} \times \mathbf{v}_h^{n-1}, \mathbf{v}_h^n - \mathbf{v}_h^{n-1} - \mathbf{p}_h^n)_\Omega \leq \frac{1}{2} (\nu_{n,0} + 2\mu_{0,n}) \eta (\|\mathbf{v}_h^{n-1}\|_\Omega^2 + \|\mathbf{v}_h^n\|_\Omega^2).$$

For the third term

$$\begin{aligned} & \eta(\nabla \mathbf{m}_h^{n-1}, \nabla \mathbf{p}_h^n)_\Omega \leq \eta \|\nabla \mathbf{m}_h^{n-1}\|_\Omega \|\nabla \mathbf{p}_h^n\|_\Omega \\ & \leq 2\eta (\mu_{0,n} \Gamma_n + \mu_{1,n} + \mu_{0,n} \Lambda(\mathbf{v}_h^n)) \|\nabla \mathbf{m}_h^{n-1}\|_\Omega \|\mathbf{v}_h^n\|_\Omega \\ & \leq \eta (\mu_{0,n} \Gamma_n + \mu_{1,n} + \mu_{0,n} \Lambda(\mathbf{v}_h^n)) \left(\frac{1}{\xi} \|\nabla \mathbf{m}_h^{n-1}\|_\Omega^2 + \xi \|\mathbf{v}_h^n\|_\Omega^2 \right) \end{aligned}$$

for any $\xi > 0$. If we now take also $\mathbf{H}_{\text{ext}}^n \neq \mathbf{0}$ into account, the right hand side will have the additional term

$$(1 + 2\mu_{0,n}) (\|\mathbf{H}_h^n\|_\Omega + \|\mathbf{H}_h^{n-1}\|_\Omega) \|\mathbf{v}_h^n\|_\Omega \leq \zeta \alpha \|\mathbf{v}_h^n\|_\Omega^2 + \frac{1 + 2\mu_{0,n}}{\zeta \alpha} (\|\mathbf{H}_h^n\|_\Omega^2 + \|\mathbf{H}_h^{n-1}\|_\Omega^2)$$

We collect the formulas and get

$$\begin{aligned} & \left((1 - \zeta - (\frac{1}{2} + \mu_{0,n})\eta)\alpha - \frac{1}{2}(\nu_{n,0} + 2\mu_{0,n})\eta - \xi\eta(\Gamma_n\mu_{0,n} + \mu_{1,n} + \mu_{0,n}\Lambda(\mathbf{v}_h^n)) \right) \|\mathbf{v}_h^n\|_\Omega^2 \\ & - \left((\frac{1}{2} + \mu_{0,n})\eta\alpha + \frac{1}{2}(\nu_{n,0} + 2\mu_{0,n})\eta \right) \|\mathbf{v}_h^{n-1}\|_\Omega^2 \\ & + \frac{1}{\tau_n} (\|\nabla \mathbf{E}_h^n\|_\Omega^2 - \|\nabla \mathbf{E}_h^{n-1}\|_\Omega^2) \\ & \leq \frac{1}{\xi} \eta (\mu_{0,n} \Gamma_n + \mu_{1,n} + \mu_{0,n} \Lambda(\mathbf{v}_h^n)) \|\nabla \mathbf{m}_h^{n-1}\|_\Omega^2 + \frac{1 + 2\mu_{0,n}}{\zeta \alpha} (\|\mathbf{H}_h^n\|_\Omega^2 + \|\mathbf{H}_h^{n-1}\|_\Omega^2). \end{aligned}$$

By [2, p. 1028] we have that $\mu_{0,n} = O(h)$ using $\tau_n = O(h)$. Further by [2, (8.5)] we see $\mu_{1,n} = O(1)$ and $\Gamma_n = O(1)$. Clearly, $\nu_{n,0} = O(1)$ at least. Finally, $\mu_{0,n} \Lambda(\mathbf{v}_h^n) = O(1)$ since by inverse estimate $\Lambda(\mathbf{v}_h^n) = O(1/h)$.

We fix h_0 so that $\mu_{0,n}, \mu_{1,n}, \nu_{n,0}, \Gamma_n, \mu_{0,n} \Lambda(\mathbf{v}_h^n)$ have a definite (but not necessarily small) bound for $h \leq h_0$, say, $\mu_{0,n} \Gamma_n + \mu_{1,n} + \mu_{0,n} \Lambda(\mathbf{v}_h^n) \leq \mu_*$. In order to derive the energy bound we have to add up these estimates for $n > k$ and we have to guarantee

$$(3.10) \quad (1 - \zeta - (1 + 2\mu_{0,n})\eta)\alpha - (\nu_{n,0} + 2\mu_{0,n})\eta - \xi\mu_* > 0.$$

So we choose for given η and sufficiently small ζ (depending on η)

$$\alpha > \frac{(\nu_{n,0} + 2\mu_{0,n})\eta}{1 - \zeta - (1 + 2\mu_{0,n})\eta}$$

and then ξ small (but independent on h) to meet (3.10). \square

Remark 2. (1) *If we limit ourselves to $\kappa = \sqrt{2}$, the proof of Theorem 1 works for $\eta_0 = 0.38$. If we then take (for example) $\nu_{n,0} = 1$, $\zeta = 0.1$ and $2\mu_{0,n} = 0.1$, we would find $\alpha > 0.85$ and $\gamma_- \geq 0.04$.*

- (2) We found numerically that one cannot find a positive symmetric \mathbf{G} for $\kappa > 1$ in case of $\eta = 0$ exploiting [3, (1.7)] in the proof of Theorem 1. In fact, the defining equations have two solutions for $\kappa < 1$ and a turning point in $\kappa = 1$.
- (3) For each $\eta > 0$ there exists $\kappa_* > 1$ such that γ_2^- ceases to be positive for $\kappa > \kappa_*$.

4. ADAPTIVITY IN SPACE

4.1. The recovery error indicator. Since $\|\nabla \mathbf{m}(t)\|_{L^2(\Omega)}$ is a quantity that appears in the energy (1.5) it is natural to also measure the error in this norm. A quite general technique for this is given by gradient recovery [1]. For the finite element space W_h this defines a mapping $\mathbf{G}_h(v_h)$ that projects the gradient of $v_h \in W_h$ into W_h^3 . Precise conditions can be obtained from [1]. Having defined such \mathbf{G}_h (componentwise) we then define a local error indicator by

$$(4.1) \quad \eta_{K,n} = \|\nabla \mathbf{m}_h^n - \mathbf{G}_h(\mathbf{m}_h^n)\|_{L^2(K)}.$$

This can lead to asymptotically exact errors in case of superconvergence. Although this is unlikely to be the case on locally refined meshes, the estimate can be compared to the best approximation in H^1 if the solution is smooth and the mesh is sufficiently fine [7].

Here we defined \mathbf{G}_h as the $L^2(\Omega)$ -projection from ∇W_h to W_h^3 , i.e.

$$(\mathbf{G}_h v_h, \boldsymbol{\varphi}_h)_\Omega = (\nabla v_h, \boldsymbol{\varphi}_h)_\Omega \quad \text{for all } \boldsymbol{\varphi}_h \in W_h^3.$$

For vector fields this is applied componentwise.

4.2. Refinement and coarsening. Having computed $[\eta_K]_{K \in \mathcal{K}_h}$ as in (4.1), we extract a minimal subset $\mathcal{A}_h \subset \mathcal{K}_h$ such that

$$(4.2) \quad \sum_{K \in \mathcal{A}_h} \eta_K^2 \geq (1 - \theta_r) \sum_{K \in \mathcal{K}_h} \eta_K^2$$

for some $\theta_r \in (0, 1)$. A new mesh will be established by two bisections of elements in \mathcal{A}_h . To choose a set of elements to be coarsened, we extract a maximal subset $\mathcal{B}_h \subset \mathcal{K}_h$ such that

$$(4.3) \quad \sum_{K \in \mathcal{B}_h} \eta_K^2 \leq (1 - \theta_c) \sum_{K \in \mathcal{K}_h} \eta_K^2.$$

For local refinement and coarsening techniques for triangulations see for example [8].

5. ADAPTIVITY IN SPACE AND TIME

Our algorithm works in the following way: In each time step it first fixes a suitable time step size. Different from [2] we use extrapolation in time of order k (instead of $k - 1$) as a predictor. This defines the spatial problem (2.3) which is solved with adaptive mesh refinement.

Algorithm 3.

Input:

A coarse initial mesh \mathcal{K}_h^{ini} , final time $T > 0$ and tolerances TOL_s (in space) and TOL_t (in time), refinement/coarsening controls $\theta_r, \theta_c > 0$ and the spatial error indicator η (4.1).

Precomputation

- We iteratively refine \mathcal{K}_h^{ini} to \mathcal{K}_h^0 using mesh refinement on elements in \mathcal{A}_h chosen by (4.2) with respect to the given initial condition $\mathbf{m}(0)$ and threshold θ_r until $\eta \leq \text{TOL}_s$.
- Then we apply two time steps solving (2.3) with the implicit Euler method, i.e., BDF-1, on the mesh \mathcal{K}_h^0 . First we define τ_1^* by (3.6), (3.7) and let $\tau_1 = \max\{\tau_{\min}, \min\{\tau_1^*/4, \tau_{\max}\}\}$. With this we compute $\mathbf{m}_h^1, \mathbf{v}_h^1$. We then define τ_2^* by (3.6) with $\|\partial_t^2 \mathbf{m}\|_{L^2(\Omega)} \approx \|\mathbf{v}_h^1 - \mathbf{v}_h^0\|_{L^2(\Omega)}/\tau_1$ and define $\tau_2 = \max\{\tau_{\min}, \frac{1}{2}\tau_1, \min\{\tau_2^*, \sqrt{2}\tau_1, \tau_{\max}\}\}$ to compute $\mathbf{m}_h^2, \mathbf{v}_h^2$. Thus we end up with the states $\mathbf{m}_h^n, \mathbf{v}_h^n$, $n = 0, 1, 2$.

Time stepping

- Given for $n \geq 3$: Approximation \mathbf{m}_h^{n-1} at time t_{n-1} on the grid \mathcal{K}_h^{n-1} and approximations $\mathbf{m}_h^{n-2}, \mathbf{m}_h^{n-3}$ at times t_{n-2}, t_{n-3} interpolated to the grid \mathcal{K}_h^{n-1} . Extrapolate $\widehat{\mathbf{m}}_h^n$.
- We compute the next time step size τ_n for BDF-2 as described in Section 3.2 with TOL_t , (3.8), and $\|\partial_t^3 \mathbf{m}\|_{L^2(\Omega)} \approx \|\partial_\tau^3 \mathbf{m}_h(t_{n-1})\|_{L^2(\Omega)} \approx \|\partial_\tau^2 \mathbf{v}_h(t_{n-1})\|_{L^2(\Omega)}$ using (3.4).
- We use the spatial indicator η from (4.1) for \mathbf{m}_h^{n-1} to first refine the mesh using the threshold θ_r until the tolerance TOL_s is achieved and then we coarsen the elements selected with θ_c , both described in Section 4.2. This results in the new mesh \mathcal{K}_h^n .
- We now solve (2.3) for \mathbf{m}_h^n at time $t_n = t_{n-1} + \tau_n$ and increase n as long as $t_n < T$. In the numerical experiments below, we might also include an extra normalisation step $\mathbf{m}_h^n \leftarrow \mathbf{m}_h^n / |\mathbf{m}_h^n|$.

6. NUMERICAL EXPERIMENTS

In the following we apply our algorithm to a number of examples in order to study its performance. In order to measure the error we choose (for $T = t_N$)

$$(6.1) \quad \text{err}_T = \max_{0 \leq n \leq N} \|\nabla(\mathbf{m}(t_n) - \mathbf{m}_h^n)\|_{L^2(\Omega)}$$

which corresponds to the theoretical result in Section 2.5 and to the chosen error indicator (4.1). The algorithm was implemented with the finite element package FEniCS [4].

We solved the saddle point problem (2.1). The resulting linear systems were solved either directly with the MUltifrontal Massively Parallel Sparse direct Solver (MUMPS) or iteratively with the stabilized biconjugate gradient method (bicgstab) featured with the incomplete LU-Decomposition as preconditioner, which both come with FEniCS. Since FEniCS does not provide a tool for local coarsening we have to perform a remeshing instead. Therefore, to perform a coarsening step, we revert to a coarser grid and refine it by applying the gradient recovery estimator.

6.1. Example 1. The following example is taken from [2, Sect. 9.2, (9.2)]. In this case we let $\Omega = (0, 1) \times (0, 1)$ and a source term \mathbf{H}_{ext} is provided such that the exact solution of (1.1)–(1.3) is

$$\mathbf{m}(t, \mathbf{x}) = \begin{bmatrix} -(x_1^3 - 3/2 x_1^2 + 1/4) \sin(3\pi t/T) \\ \sqrt{1 - (x_1^3 - 3/2 x_1^2 + 1/4)^2} \\ -(x_1^3 - 3/2 x_1^2 + 1/4) \cos(3\pi t/T) \end{bmatrix}.$$

We choose $T = 0.1$, $C_e = 1$ and $\alpha = 0.2$. In this example the temporal error dominates the total error, so we fix 2 for the polynomial degree in space. Furthermore we applied an extra normalisation step as mentioned in Algorithm 3.

In Figure 6.1 (left) we show the error (6.1) of the adaptive time-stepping method for BDF-1 to BDF-3 in dependence of the prescribed tolerance in time TOL_t for a fixed mesh. For BDF- k we have the convergence order k , but local truncation error $k + 1$, for uniform steps. Since we equilibrate the local truncation error we have $\tau^{k+1} \sim \text{TOL}_t$ and therefore we observe $\text{err}_T \sim \text{TOL}_t^{k/(k+1)}$. For a fixed choice of TOL_t we show in Figure 6.1 (right) the development of τ during time.

In Figure 6.2, we compare the error err_T for uniform step sizes versus adaptive step sizes. For this purpose, we first compute the time-adaptive BDF-3 scheme for various TOL_t and store the total number of time steps N_t . Subsequently, we perform simulations with uniform step sizes, where we chose $\tau = T/N_t$, i. e., we need the same N_t as in the time-adaptive simulations. We see that the adaptive algorithm leads to a smaller error for the same amount of work.

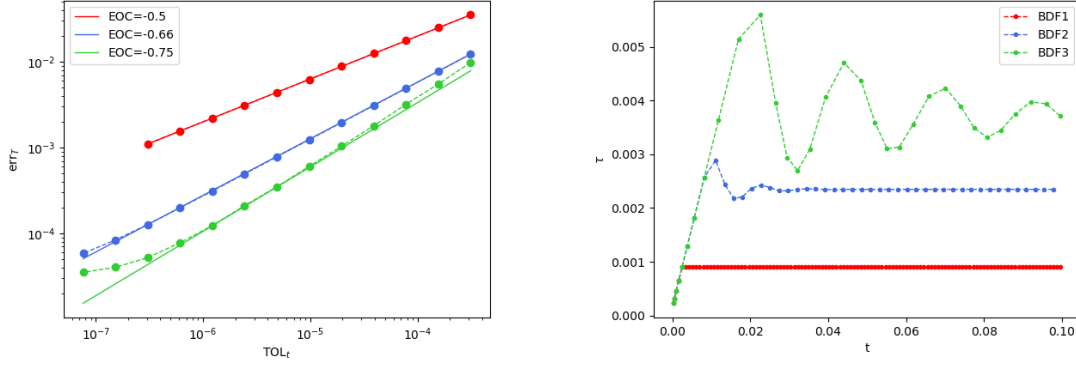


FIGURE 6.1. Example 6.1: Left: Convergence of err_T for adaptive time-stepping for BDF-1 (red), BDF-2 (blue) and BDF-3 (green) with fixed $h = 1/\sqrt{N_c}$, where $N_c = 15488$ is the number of cells. Right: Evolution of the timestep size τ for $TOL_t = 6.25 \cdot 10^{-4}$.

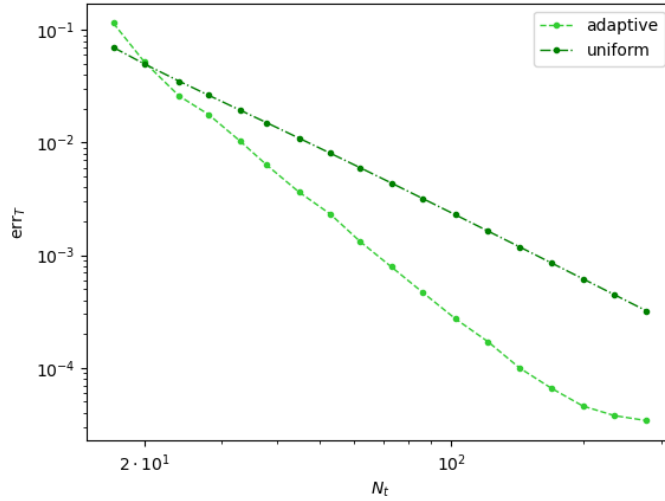


FIGURE 6.2. Example 6.1: Comparison of uniform step size (dark green) vs. adaptive step size (light green) computation for the BDF-3 method with fixed $h = 1/\sqrt{N_c}$, where $N_c = 15488$ is the number of cells.

6.2. Example 2. The following example is taken from [2, Sect. 9.2, (9.1)]. In this case we let $\Omega = (0, 1) \times (0, 1)$ and a source term \mathbf{H}_{ext} is provided such that the exact solution of (1.1)–(1.3) is

$$\mathbf{m}(t, \mathbf{x}) = \begin{cases} \begin{bmatrix} C_0(x_1-1/2) e^{-\frac{g(t)}{1/4-d(\mathbf{x})}} \\ C_0(x_2-1/2) e^{-\frac{g(t)}{1/4-d(\mathbf{x})}} \\ \sqrt{1-C_0^2 d(\mathbf{x})} e^{-\frac{2g(t)}{1/4-d(\mathbf{x})}} \end{bmatrix} & \text{if } d(\mathbf{x}) < 1/4, \\ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & \text{otherwise,} \end{cases}$$

with $\mathbf{m}^0 = \mathbf{m}(0)$, $g(t) = (T_0 + 0.1)/(T_0 + 0.1 - t)$, $d(\mathbf{x}) = |\mathbf{x} - [1, 1]/2|^2$, $C_0 = 400$. We choose $C_e = 1$, $\alpha = 0.2$, $T_0 = 0.06$ and $T = 0.05$ as the final time. In this example the spatial error dominates the total error. We use $\tau = T/100$ as well as $\theta_r = 0.6$.

We perform numerical experiments for linear, quadratic and cubic finite elements on uniform and adaptive spatial meshes. Time integration is done with BDF-2. We execute Algorithm 3 with additional normalisation step for several mesh sizes h and in case of space-adaption for

several tolerances TOL_s . In Figure 6.3 (left) we notice convergence in the error norm err_T given by (6.1) of order one in TOL_s for all polynomial degrees until saturation occurs due to the fixed time step. However, in Figure 6.3 (right) we notice smaller errors in the error norm err_T for higher polynomial degree if we relate these errors to the maximal degrees of freedom we obtained at a discrete time step during the simulation.

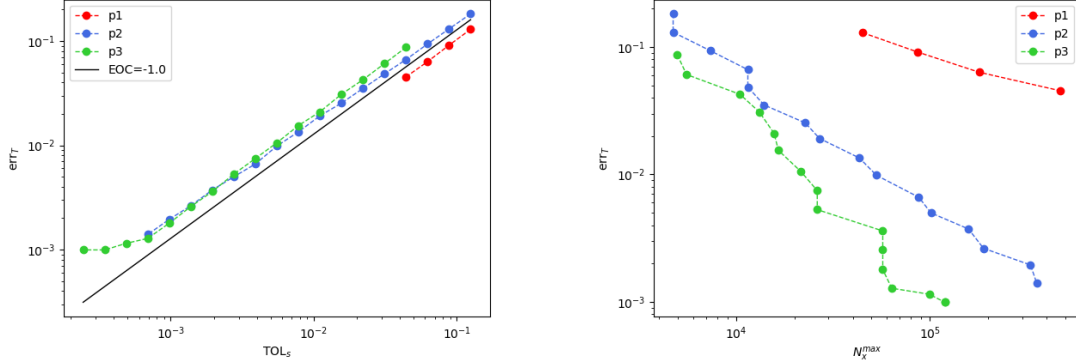


FIGURE 6.3. Example 6.2: Left: space tolerance TOL_s vs. err_T . Right: maximal degrees of freedom $N_x^{max} := \max_{0 \leq n \leq N_t} \{N_x(t_n)\}$ vs. err_T where N_x denotes the number of degrees of freedom and N_t the number of time steps. Various polynomial degrees of the finite element space are presented, i.e., linear (red), quadratic (blue) and cubic (green).

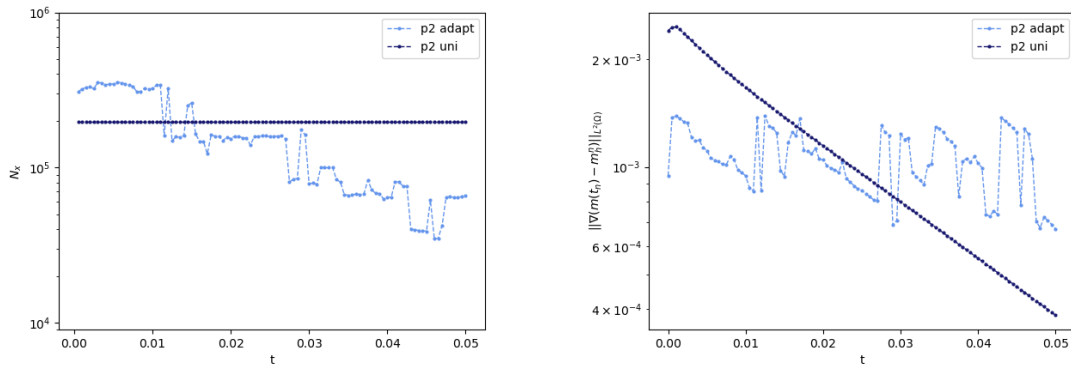


FIGURE 6.4. Example 6.2: Comparison between runs on uniform and adaptive spatial meshes for polynomial degree 2. Left: Evolution of the degrees of freedom. Right: Evolution of the H^1 error. We used $TOL_s = 6.91 \cdot 10^{-4}$ for the spatial tolerance.

6.3. Example 3. Here we consider an example that appears to show singular behaviour and has been considered in [10], [9], [11]. However, we will demonstrate that computations on finer meshes and particularly adaptive computations prevent a singularity from forming. We let $\Omega = (-1/2, 1/2)^2$ and choose constants $C_e = 1$ and $\alpha = 1$. We solve (1.1)–(1.3) with initial data is given by

$$\mathbf{m}^0(\mathbf{x}) = \begin{cases} \frac{1}{A^2 + |\mathbf{x}|^2} \begin{bmatrix} 2A\mathbf{x} \\ A^2 - |\mathbf{x}|^2 \end{bmatrix} & \text{if } |\mathbf{x}|^2 \leq 1/2, \\ \begin{bmatrix} 0 \\ -1 \end{bmatrix} & \text{otherwise,} \end{cases}$$

where $A = (1 - 2|\mathbf{x}|)^4/s$, $s = 16$. No external field is applied, i.e., $\mathbf{H}_{\text{ext}} = \mathbf{0}$. An exact solution is not known and we have to rely on qualitative behavior of the solution in order to judge the validity of the results.

We performed numerical experiments for the space- and time-adaptive Algorithm 3 and choose the mesh refinement parameter $\theta_r = 0.95$. The time integration was done with BDF-2.

In Figure 6.6 we run the algorithm for the time tolerance $\text{TOL}_t = 10^{-5}$ and space tolerance $\text{TOL}_s = 0.4$. We compare the adaptive algorithm with uniform discretizations on several meshsizes. For both, we did not apply the additional normalization step. The takaway of these experiments is that the adaptive algorithm prevents the singularity from forming with far less degrees of freedoms and thus far less computational effort than the uniform approach. This indicates that the error estimator is also useful in non-smooth situations, despite the theory only working under smoothness assumptions on the solution.

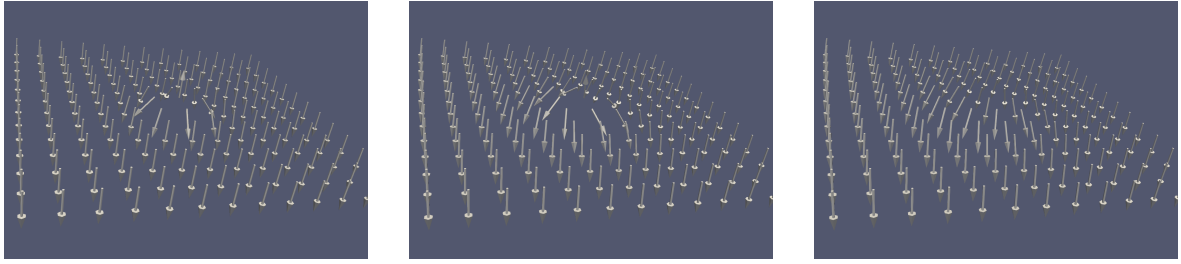


FIGURE 6.5. Example 6.4: Solution \mathbf{m}_h for the uniform mesh $h = 1/14$ and uniform timestep $\tau = 10^{-3}$ at $t = 0$ (left), $t = 0.083$ (middle) and $t = 0.092$ (right) according to (2.3). The singularity appears in the center of the vectorfield, where \mathbf{m} points upward in a very small neighborhood only.

6.4. Example 4. The following solution simulates a moving domain wall. We take $\Omega = (0, 1) \times (0, 0.2)$, $T = 0.35$, $C_e = 0.1$, $\alpha = 1$ and provide initial data

$$\mathbf{m}^0(\mathbf{x}) = \begin{cases} \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} & \text{if } x_1 < c - d, \\ \begin{bmatrix} 0 \\ \cos(\pi\zeta/2) \\ \sin(\pi\zeta/2) \end{bmatrix} & \text{if } c - d \leq x_1 \leq c + d, \\ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & \text{if } c + d \leq x_1. \end{cases}$$

for $\zeta = \sin(\pi(x_1 - c)/2d)$, $c = 0.2$, $d = 0.125$. An exact solution is not known. We choose the mesh refinement parameter as $\theta_r = 0.8$. Furthermore we apply a constant external field $\mathbf{H}_{\text{ext}} = [0, 0, -50]$.

We perform numerical experiments for adaptive quadratic finite elements and adaptive time-stepping with BDF-2, where we apply the fully adaptive Algorithm 3 with additional normalization step. We choose the tolerances $\text{TOL}_s = 10^{-4}$ and $\text{TOL}_t = 10^{-6}$.

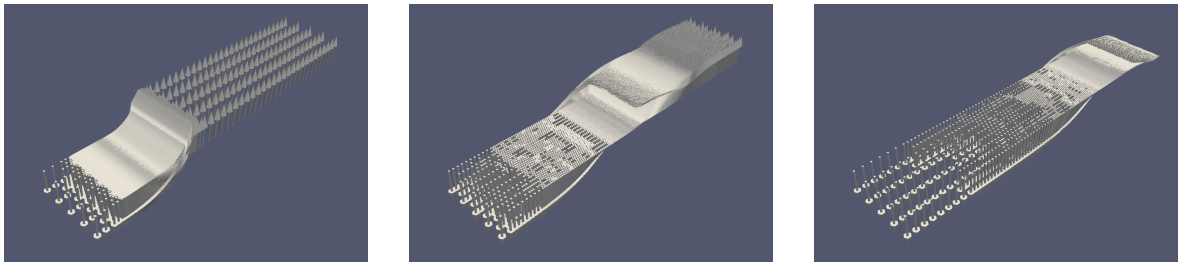


FIGURE 6.7. Example 6.4: Solution \mathbf{m} at $t = 0$ (left), $t = 0.15$ (middle) and $t = 0.3$ (right). At the final time $t = T = 0.35$, all magnets are pointing downwards.

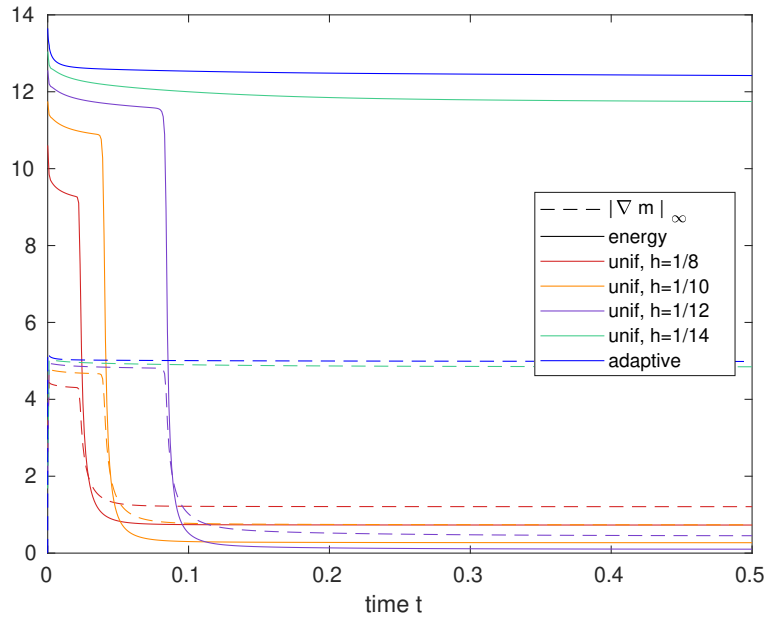


FIGURE 6.6. Example 6.3: We compare simulations with uniform meshes and timesteps with a fully adaptive simulation. The time step size for the uniform simulations is $\tau = 10^{-3}$ and the mesh size ranges from $h = 1/8$ to $h = 1/14$. The adaptive simulation starts with a mesh size of $h = 1/2$. The singularity, i.e., the rotation of the entire magnetization towards $[0, 0, -1]$, is observed via a sharp decay of the energy. We observe that only the finest uniform simulation avoids the singularity (that requires a total number of 1261500 degrees of freedom). In contrast, the adaptive simulation achieves the same with just 209223 degrees of freedom, less than the coarsest uniform simulation.

The initial configuration consists of layers of downwards and upwards pointing magnets with an intermediate layer. During the simulation the magnets pointing upwards flip their direction due to the external field and in this way the layer moves to the right (Figure 6.7). Figure 6.8 shows the obtained meshes at corresponding time instances.

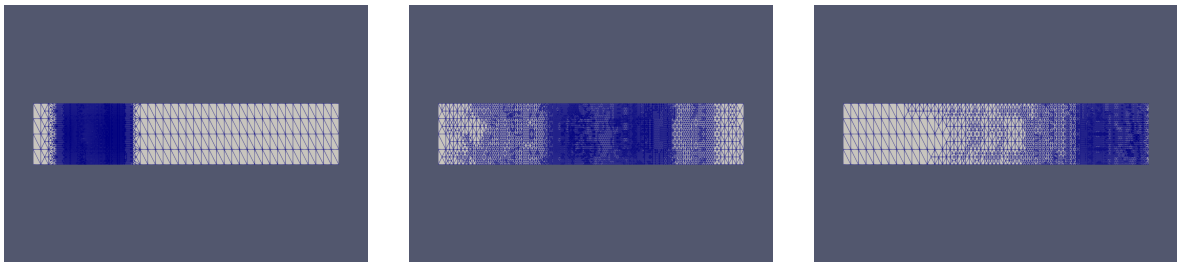


FIGURE 6.8. Example 6.4: Mesh at $t = 0$ (left), $t = 0.15$ (middle) and $t = 0.3$ (right). The refined mesh moves along with the magnetic wave.

In Figure 6.9 we show the temporal development of the time step size τ (left) and the degrees of freedom N_x (right) in case of the fully adaptive algorithm. In total, we compute $N_t = 587$ time steps and a maximum of 440559 degrees of freedom. After all vectors change direction, the coarsening procedure will finally lead to the initially given macro-triangulation.

CONCLUSION

We propose a time and space adaptive algorithm for the numerical approximation of the Landau–Lifshitz–Gilbert equation. Under certain regularity assumptions on the exact solutions, we show that the numerical approximation satisfies an energy inequality similar to that

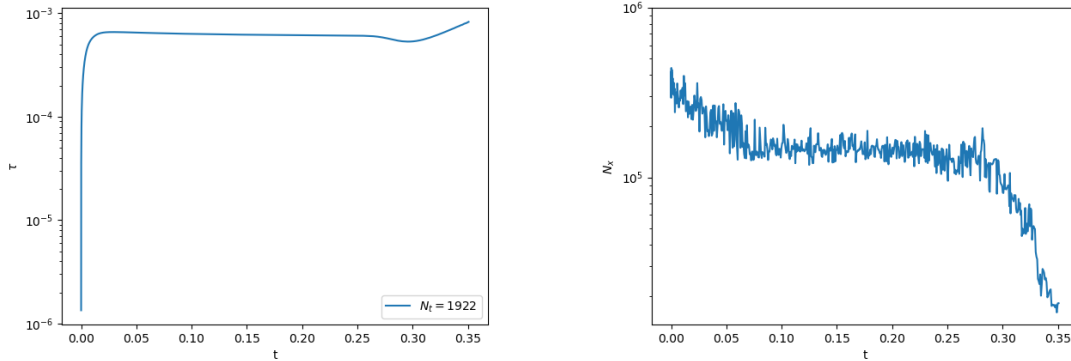


FIGURE 6.9. Example 6.4: Temporal development of τ (left) and the degrees of freedom N_x (right) in case of the space- and time-adaptive algorithm.

of the exact solution. Numerical experiments demonstrate the advantages of the adaptive algorithms over uniform approaches such as increased convergence speed (Example 6.1) or increased stability (Example 6.3). Many interesting theoretical questions still remain open for future research: Is the error estimator an upper bound for the error? Does the adaptive algorithm converge towards the exact solution? Does the energy inequality (Theorem 1) hold under reduced regularity assumptions?

Acknowledgement. Jan Bohn, Willy Dörfler, Michael Feischl and Stefan Karch gratefully acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) within the SFB 1173 “Wave Phenomena” (Project-ID 258734477).

REFERENCES

- [1] Mark Ainsworth and J. Tinsley Oden. *A posteriori error estimation in finite element analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2000.
- [2] Georgios Akrivis, Michael Feischl, Balázs Kovács, and Christian Lubich. Higher-order linearly implicit full discretization of the Landau–Lifshitz–Gilbert equation. *Math. Comp.*, 90(329):995–1038, 2021.
- [3] Georgios Akrivis and Emmanuil Katsoprinakis. Backward difference formulae: new multipliers and stability properties for parabolic equations. *Math. Comp.*, 85(301):2195–2216, 2016.
- [4] Martin S. Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E. Rognes, and Garth N. Wells. The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100):9–23, 2015.
- [5] François Alouges. A new finite element scheme for Landau–Lifshitz equations. *Discrete Contin. Dyn. Syst. Ser. S*, 1(2):187–196, 2008.
- [6] Rong An. Optimal error estimates of linearized Crank–Nicolson Galerkin method for Landau–Lifshitz equation. *J. Sci. Comput.*, 69(1):1–27, 2016.
- [7] Randolph E. Bank and Harry Yserentant. A note on interpolation, best approximation, and the saturation property. *Numer. Math.*, 131(1):199–203, 2015.
- [8] E. Bänsch. Local mesh refinement in 2 and 3 dimensions. *Impact Comput. Sci. Engrg.*, 3:181–191, 1991.
- [9] Sören Bartels, Joy Ko, and Andreas Prohl. Numerical analysis of an explicit approximation scheme for the Landau-Lifshitz-Gilbert equation. *Math. Comp.*, 77(262):773–788, 2008.
- [10] Sören Bartels and Andreas Prohl. Convergence of an implicit finite element method for the Landau–Lifshitz–Gilbert equation. *SIAM J. Numer. Anal.*, 44(4):1405–1419, 2006.
- [11] L’ubomír Bañas, Sören Bartels, and Andreas Prohl. A convergent implicit finite element discretization of the Maxwell–Landau–Lifshitz–Gilbert equation. *SIAM J. Numer. Anal.*, 46(3):1399–1422, 2008.
- [12] J.-M. Coron. Nonuniqueness for the heat flow of harmonic maps. *Ann. Inst. H. Poincaré C Anal. Non Linéaire*, 7(4):335–344, 1990.
- [13] Giovanni Di Fratta, Michael Innerberger, and Dirk Praetorius. Weak-strong uniqueness for the Landau-Lifshitz-Gilbert equation in micromagnetics. *Nonlinear Anal. Real World Appl.*, 55:103122, 13, 2020.
- [14] Giovanni Di Fratta, Carl-Martin Pfeiler, Dirk Praetorius, Michele Ruggeri, and Bernhard Stiftner. Linear second-order IMEX-type integrator for the (eddy current) Landau-Lifshitz-Gilbert equation. *IMA J. Numer. Anal.*, 40(4):2802–2838, 2020.

- [15] Michael Feischl and Thanh Tran. The eddy current–LLG equations: FEM-BEM coupling and a priori error estimates. *SIAM J. Numer. Anal.*, 55(4):1786–1819, 2017.
- [16] Michael Feischl and Thanh Tran. Existence of regular solutions of the Landau–Lifshitz–Gilbert equation in 3D with natural boundary conditions. *SIAM J. Math. Anal.*, 49(6):4470–4490, 2017.
- [17] Riccardo Ferrero and Alessandra Manzin. Adaptive geometric integration applied to a 3d micromagnetic solver. *Journal of Magnetism and Magnetic Materials*, 518:167409, 2021.
- [18] Albert Fert, Vincent Cros, and Joao Sampaio. Skyrmions on the track. *Nat. Nanotechnol.*, 8:152–156, 2013.
- [19] Huadong Gao. Optimal error estimates of a linearized backward Euler FEM for the Landau–Lifshitz equation. *SIAM J. Numer. Anal.*, 52(5):2574–2593, 2014.
- [20] T. Gilbert. A Lagrangian formulation of the gyromagnetic equation of the magnetic field. *Phys Rev*, 100:1243–1255, 1955.
- [21] Rolf Dieter Grigorieff. Stability of multistep-methods on variable grids. *Numer. Math.*, 42(3):359–377, 1983.
- [22] E. Hairer and G. Wanner. *Solving ordinary differential equations. II*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2010. Stiff and differential-algebraic problems, Second revised edition, paperback.
- [23] Gino Hrkac, Carl-Martin Pfeiler, Dirk Praetorius, Michele Ruggeri, Antonio Segatti, and Bernhard Stiftner. Convergent tangent plane integrators for the simulation of chiral magnetic skyrmion dynamics. *Adv. Comput. Math.*, 45(3):1329–1368, 2019.
- [24] E. Kritsikis, A. Vaysset, L. D. Buda-Prejbeanu, F. Alouges, and J.-C. Toussaint. Beyond first-order finite element schemes in micromagnetics. *J. Comput. Phys.*, 256:357–366, 2014.
- [25] L. Landau and E. Lifshitz. On the theory of the dispersion of magnetic permeability in ferromagnetic bodies. *Phys. Z. Sowjetunion*, 8:153–168, 1935.
- [26] Dirk Praetorius, Michele Ruggeri, and Bernhard Stiftner. Convergence of an implicit-explicit midpoint scheme for computational micromagnetics. *Comput. Math. Appl.*, 75(5):1719–1738, 2018.
- [27] Andreas Prohl. *Computational micromagnetism*. Advances in Numerical Mathematics. B. G. Teubner, Stuttgart, 2001.
- [28] Thomas Schrefl. Finite elements in numerical micromagnetics. Part I: Granular hard magnets. *Journal of Magnetism and Magnetic Materials*, 207(1):45–65, 1999.
- [29] D. Weller et al. FePt heat assisted magnetic recording media. *Journal of Vacuum Science & Technology B, Nanotechnology and Microelectronics*, 34, 2016.