An Efficient Structure Preserving Method for The Landau-Lifshitz Equation with Stable Double Diffusion

Changjian Xie[a,*]

[a]School of Mathematics and Physics, Xi'an-Jiaotong-Liverpool University, Re'ai Rd. 111, Suzhou, 215123, Jiangsu, China.

Abstract

In this paper, a structure-preserving numerical method stabilized by double diffusion is proposed for the Landau-Lifshitz equation. Such a method is constructed by combining two key numerical techniques: the solution of the heat equation via two iterations of the backward Euler method, and a linearized scheme of the Crank-Nicolson update. Numerical experiments are conducted to verify the temporal accuracy of first order and spatial accuracy of second order for the developed method. Furthermore, the norm-preserving property, numerical stability, and robustness of the method are demonstrated in detail, which collectively validate the effectiveness and reliability of the proposed numerical approach.

Keywords: Landau-Lifshitz equation, structure preserving, double diffusion, stability

1. Introduction

The Landau–Lifshitz (LL) equation [1, 2] which describes the evolution of magnetization in continuum ferromagnets plays an important role in the understanding of nonequilibrium magnetism. Such a ferromagnetic model resolve the nonlinear term which is the vector field orthogonal to the magnetization vector. The LL equation is given by

$$\partial_t \boldsymbol{m} = -\boldsymbol{m} \times \Delta \boldsymbol{m} - \alpha \boldsymbol{m} \times (\boldsymbol{m} \times \Delta \boldsymbol{m}),$$

where ∂_t and Δ means the time and space derivatives, respectively, α is the damping parameter. Such a model without damping in the atomistic level can be thought as the time-dependent Schrödinger equation.

The Schrödinger map equation (also known as the undamped Landau-Lifshitz equation)

$$\partial_t \boldsymbol{m} = -\boldsymbol{m} \times \Delta \boldsymbol{m} \tag{1}$$

describes the conservative dynamics of a unit vector field $\boldsymbol{m}(\boldsymbol{x}, t)$ with $|\boldsymbol{m}| = 1$. If one consider a quantum Heisenberg ferromagnet with exchange interaction, the discrete Hamiltonian is:

$$\hat{H} = -\frac{J}{2} \sum_{\langle i,j \rangle} \hat{\boldsymbol{S}}_i \cdot \hat{\boldsymbol{S}}_j \tag{2}$$

*Corresponding author.
Email address: Changjian.Xie@xjtlu.edu.cn (Changjian Xie)

where $J > 0$ is the exchange integral, $\hat{\boldsymbol{S}}_i$ denotes the quantum spin operator at lattice site i, and $\langle i, j \rangle$ denotes nearest-neighbor lattice sites. In the continuum limit, the discrete spin system becomes a continuous spin field $\hat{\boldsymbol{S}}(\boldsymbol{x})$, and the Hamiltonian takes the form:

$$\hat{H} = \frac{A}{2} \int_{\Omega} (\nabla \hat{\boldsymbol{S}}) \cdot (\nabla \hat{\boldsymbol{S}}) \, dV, \tag{3}$$

where A is the exchange stiffness constant, and Ω is the spatial domain. The time evolution of quantum operators follows the Heisenberg equation:

$$i\hbar \frac{d\hat{\boldsymbol{S}}}{dt} = \left[\hat{\boldsymbol{S}}, \hat{H} \right] \tag{4}$$

where $\hbar$ is the reduced Planck constant, and $[\cdot, \cdot]$ denotes the quantum commutator. Substituting the continuum Hamiltonian into the Heisenberg equation and evaluating the spin commutator algebra yield the quantum dynamical equation for the spin field. In the classical limit ($\hbar \to 0$), the quantum spin operator is replaced by its classical expectation value:

$$\boldsymbol{m} = \frac{\langle \hat{\boldsymbol{S}} \rangle}{S}, \quad |\boldsymbol{m}| = 1, \tag{5}$$

where S is the spin quantum number, and $\boldsymbol{m}$ is the dimensionless unit magnetization vector. The classical equation of motion for spin precession is:

$$\frac{\partial \boldsymbol{m}}{\partial t} = \gamma \boldsymbol{m} \times \boldsymbol{H}_{\text{eff}}, \tag{6}$$

where γ is the gyromagnetic ratio, and $\boldsymbol{H}_{\text{eff}}$ is the effective magnetic field derived from the variational derivative of the energy functional. The classical exchange energy functional corresponding to the quantum Hamiltonian is:

$$E[\boldsymbol{m}] = \frac{A}{2} \int_{\Omega} |\nabla \boldsymbol{m}|^2 dV. \tag{7}$$

The effective field is defined as the negative variational derivative of the energy:

$$\boldsymbol{H}_{\text{eff}} = -\frac{\delta E}{\delta \boldsymbol{m}}. \tag{8}$$

Using the calculus of variations, we compute:

$$\frac{\delta E}{\delta \boldsymbol{m}} = -A \Delta \boldsymbol{m}, \tag{9}$$

where $\Delta = \nabla \cdot \nabla$ is the Laplacian operator. Thus, we have

$$\boldsymbol{H}_{\text{eff}} = A \Delta \boldsymbol{m}. \tag{10}$$

Substituting the effective field into the classical spin equation below,

$$\partial_t \boldsymbol{m} = \gamma A \, \boldsymbol{m} \times \Delta \boldsymbol{m}. \tag{11}$$

By absorbing the constant γA into the time scale (re-scaling time), we obtain the standard dimensionless Schrödinger map equation:

$$\partial_t \boldsymbol{m} = -\boldsymbol{m} \times \Delta \boldsymbol{m} \tag{12}$$

This equation is a geometric generalization of the linear Schrödinger equation on the 2-sphere manifold S^2. It preserves the unit length constraint $|\boldsymbol{m}| = 1$ and conserves the exchange energy, representing the Hamiltonian dynamics of spin systems without dissipation.

Based on this property at the continuous level, we have the constraints that the magnetization vector is orthogonal to the change rate of magnetization vector along time. At the discrete level, the property that the norm of numerical magnetization at the current step is the same as that of the previous step, the same as the norm at initial step. A notable limitation persists:

- From the engineering application level, one can use the simple projection step to maintain the norm constraints. The popular existing work can be referred to Gauss-Seidel Projection method [3] and the second order semi-implicit projection method [4]. However, the projection is nonlinear, which is hard to get the stability or convergence analysis.
- From the mathematics level, the implicit method involves a nonlinear system, which is hard to solve and no guarantees for the solvability.
- From the model level, there's the parabolic equations with the initial and boundary conditions. No need to put other conditions into the model, say naively, we just kick off the numerics for the LL equation without constraints. Some existing work, refer to [5], [6] treat the norm constraint as an extra term to solve other equation.

Based on these limitations, we design a numerical norm preserving method for LL equation without any constraints handling. Such a linear approach facilitates theoretical analysis.

In this paper, we propose a double diffusion approach (if we take $\alpha = 0$ for simplicity) as below,

$$\begin{aligned}
\frac{\tilde{\boldsymbol{m}}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} &= \Delta_h \tilde{\boldsymbol{m}}_h^{n+1} \\
\frac{\boldsymbol{m}_h^{*,n+1} - \tilde{\boldsymbol{m}}_h^{n+1}}{\Delta t} &= \Delta_h \boldsymbol{m}_h^{*,n+1} \\
\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} &= -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \boldsymbol{m}_h^{*,n+1}.
\end{aligned}$$

Herein, we notice that the single diffusion is not enough for its stability.

The rest of this paper is organized as follows. Section 2 begins with the LL equation without damping model, followed by a detailed description of the proposed numerical scheme. Section 3 introduces the consistency, stability and convergence analysis for such a proposed method. Section 4 presents extensive numerical results, encompassing verification of temporal and spatial accuracy in one-dimensional (1D) and three-dimensional (3D) settings. The norm preserving and stability are preserved as well. Concluding remarks and potential future research directions are provided in Section 5.

2. Proposed method

For the construction of the structure preserving method, we consider the LL equation with $\alpha = 0$ below,

$$\boldsymbol{m}_t = -\boldsymbol{m} \times \Delta \boldsymbol{m}. \tag{13}$$

If we consider the simple linear vectorial equation

$$\boldsymbol{m}_t = -\boldsymbol{m} \times \boldsymbol{a}, \tag{14}$$

where $\boldsymbol{a}^T = (a_1, a_2, a_3)$ is a constant vector. We use the Crank-Nicolson method to eq. (14), we have

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \boldsymbol{a}, \tag{15}$$

which is norm preserving, since that if $\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n$ do the inner product for both sides, leading to

$$\|\boldsymbol{m}_h^{n+1}\|_2 = \|\boldsymbol{m}_h^n\|_2.$$

After rearranging the compact form for eq. (15), we have

$$\begin{pmatrix} 1 & \frac{1}{2}\Delta t a_3 & -\frac{1}{2}\Delta t a_2 \\ -\frac{1}{2}\Delta t a_3 & 1 & \frac{1}{2}\Delta t a_1 \\ \frac{1}{2}\Delta t a_2 & -\frac{1}{2}\Delta t a_1 & 1 \end{pmatrix} \begin{pmatrix} m_1^{n+1} \\ m_2^{n+1} \\ m_3^{n+1} \end{pmatrix} = \begin{pmatrix} m_1^n + \frac{1}{2}\Delta t(a_2 m_3^n - a_3 m_2^n) \\ m_2^n + \frac{1}{2}\Delta t(a_3 m_1^n - a_1 m_3^n) \\ m_3^n + \frac{1}{2}\Delta t(a_1 m_2^n - a_2 m_1^n) \end{pmatrix}$$

We then have another form

$$\begin{pmatrix} m_1^{n+1} \\ m_2^{n+1} \\ m_3^{n+1} \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{2}\Delta t a_3 & -\frac{1}{2}\Delta t a_2 \\ -\frac{1}{2}\Delta t a_3 & 1 & \frac{1}{2}\Delta t a_1 \\ \frac{1}{2}\Delta t a_2 & -\frac{1}{2}\Delta t a_1 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 & -\frac{1}{2}\Delta t a_3 & \frac{1}{2}\Delta t a_3 \\ \frac{1}{2}\Delta t a_3 & 1 & -\frac{1}{2}\Delta t a_1 \\ -\frac{1}{2}\Delta t a_2 & \frac{1}{2}\Delta t a_1 & 1 \end{pmatrix} \begin{pmatrix} m_1^n \\ m_2^n \\ m_3^n \end{pmatrix} = A \begin{pmatrix} m_1^n \\ m_2^n \\ m_3^n \end{pmatrix}$$

where

$$A = \frac{1}{S} \begin{pmatrix} 1 + \beta^2 a_1^2 & -2\beta a_3 + \beta^2 a_1 a_2 & 2\beta a_2 + \beta^2 a_1 a_3 \\ 2\beta a_3 + \beta^2 a_1 a_2 & 1 + \beta^2 a_2^2 & -2\beta a_1 + \beta^2 a_2 a_3 \\ -2\beta a_2 + \beta^2 a_1 a_3 & 2\beta a_1 + \beta^2 a_2 a_3 & 1 + \beta^2 a_3^2 \end{pmatrix}$$

where $S = \det(A) = 1 + \beta^2(a_1^2 + a_2^2 + a_3^2)$ and $\beta = \frac{\Delta t}{2}$. We can prove that A is an orthogonal matrix. For any $x \in \mathbb{R}^3$,

$$\|Ax\|_2^2 = (Ax)^\top (Ax) = x^\top (A^\top A) x = x^\top x = \|x\|_2^2,$$

so $\|Ax\|_2 = \|x\|_2$ for all x. By definition of the induced 2-norm,

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} = 1.$$

Hence $\|A\|_2$ is bounded by 1 and is independent of the time step Δt.

For eq. (13), we propose the following structure preserving schemes:

- Sheme I: explicit for $\Delta \boldsymbol{m}$,

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \boldsymbol{m}_h^n, \tag{16}$$

  suffers from a CFL-type condition for the stability.

- Sheme II: implicit for $\Delta \boldsymbol{m}$,

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \boldsymbol{m}_h^{n+1}, \tag{17}$$

  which poses a difficulty for the nonlinear systems with high complexity.

- Scheme III: Semi-implicit method,

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \tilde{\boldsymbol{m}}_h^{n+1}. \tag{18}$$

  To be specific, we propose

$$\begin{pmatrix} 1 & \frac{1}{2}\Delta t\Delta_h \tilde{m}_3^{n+1} & -\frac{1}{2}\Delta t\Delta_h \tilde{m}_2^{n+1} \\ -\frac{1}{2}\Delta t\Delta_h \tilde{m}_3^{n+1} & 1 & \frac{1}{2}\Delta t\Delta_h \tilde{m}_1^{n+1} \\ \frac{1}{2}\Delta t\Delta_h \tilde{m}_2^{n+1} & -\frac{1}{2}\Delta t\Delta_h \tilde{m}_1^{n+1} & 1 \end{pmatrix} \begin{pmatrix} m_1^{n+1} \\ m_2^{n+1} \\ m_3^{n+1} \end{pmatrix} = \begin{pmatrix} m_1^n + \frac{1}{2}\Delta t(\Delta_h \tilde{m}_2^{n+1} m_3^n - \Delta_h \tilde{m}_3^{n+1} m_2^n) \\ m_2^n + \frac{1}{2}\Delta t(\Delta_h \tilde{m}_3^{n+1} m_1^n - \Delta_h \tilde{m}_1^{n+1} m_3^n) \\ m_3^n + \frac{1}{2}\Delta t(\Delta_h \tilde{m}_1^{n+1} m_2^n - \Delta_h \tilde{m}_2^{n+1} m_1^n) \end{pmatrix}$$

  The trick is to handling the $\tilde{\boldsymbol{m}}_h^{n+1}$. If $\tilde{\boldsymbol{m}}_h^{n+1} = (I - k\Delta_h)^{-1} \boldsymbol{m}_h^n$, such a scheme is proved to be mildly better than CFL condition. We should get a much better estimation for $\tilde{\boldsymbol{m}}_h^{n+1}$.

Based on the stability, we propose a structure preserving method A for LL equation without damping term below,

- Implicit iteration:

$$\frac{\tilde{\boldsymbol{m}}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = \Delta_h \tilde{\boldsymbol{m}}_h^{n+1},$$

- preserving iteration:

$$\begin{pmatrix} 1 & \frac{1}{2}\Delta t\Delta_h \tilde{m}_3^{n+1} & -\frac{1}{2}\Delta t\Delta_h \tilde{m}_2^{n+1} \\ -\frac{1}{2}\Delta t\Delta_h \tilde{m}_3^{n+1} & 1 & \frac{1}{2}\Delta t\Delta_h \tilde{m}_1^{n+1} \\ \frac{1}{2}\Delta t\Delta_h \tilde{m}_2^{n+1} & -\frac{1}{2}\Delta t\Delta_h \tilde{m}_1^{n+1} & 1 \end{pmatrix} \begin{pmatrix} m_1^{n+1} \\ m_2^{n+1} \\ m_3^{n+1} \end{pmatrix} = \begin{pmatrix} m_1^n + \frac{1}{2}\Delta t(\Delta_h \tilde{m}_2^{n+1} m_3^n - \Delta_h \tilde{m}_3^{n+1} m_2^n) \\ m_2^n + \frac{1}{2}\Delta t(\Delta_h \tilde{m}_3^{n+1} m_1^n - \Delta_h \tilde{m}_1^{n+1} m_3^n) \\ m_3^n + \frac{1}{2}\Delta t(\Delta_h \tilde{m}_1^{n+1} m_2^n - \Delta_h \tilde{m}_2^{n+1} m_1^n) \end{pmatrix}$$

We choose an exact solution as shown in the section 4 in 1D, we get the result which is presented in table 1. Such a method is not unconditionally stable.

In turn, we propose a new structure preserving method B for LL equation without damping term below,

- Implicit iteration:

$$\frac{\tilde{\boldsymbol{m}}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = \Delta_h \tilde{\boldsymbol{m}}_h^{n+1},$$

Table 1: Scheme A when $h = 1D-4$, $T = 1d-1$.

k	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_\infty$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_2$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_{H^1}$
2.0D-2	0.018000150280207	0.013706382552114	0.109659143257535
1.0D-2	0.012210508922421	0.008808362054035	0.069865510903475
5.0D-3	0.006883089183841	0.004488168870136	0.033485734897977
2.5D-3	0.003297959739207	0.002226154439315	0.016331342861239
1.25D-3	0.195917109094554	0.062865700179793	4.156293009245663e+02
6.25D-4	1.947048552154076	1.414308905489441	6.822468197719807e+03
3.125D-4	2.005334814166140	1.414276537507208	6.851280277634909e+03

- double diffusion iteration:

$$m_i^{*,n+1} = (I - \Delta t \Delta_h)^{-1} \tilde{m}_i^{n+1}.$$

- preserving iteration:

$$\begin{pmatrix} 1 & \frac{1}{2}\Delta t\Delta_h m_3^{*,n+1} & -\frac{1}{2}\Delta t\Delta_h m_2^{*,n+1} \\ -\frac{1}{2}\Delta t\Delta_h m_3^{*,n+1} & 1 & \frac{1}{2}\Delta t\Delta_h m_1^{*,n+1} \\ \frac{1}{2}\Delta t\Delta_h m_2^{*,n+1} & -\frac{1}{2}\Delta t\Delta_h m_1^{*,n+1} & 1 \end{pmatrix} \begin{pmatrix} m_1^{n+1} \\ m_2^{n+1} \\ m_3^{n+1} \end{pmatrix} = \begin{pmatrix} m_1^n + \frac{1}{2}\Delta t(\Delta_h m_2^{*,n+1} m_3^n - \Delta_h m_3^{*,n+1} m_2^n) \\ m_2^n + \frac{1}{2}\Delta t(\Delta_h m_3^{*,n+1} m_1^n - \Delta_h m_1^{*,n+1} m_3^n) \\ m_3^n + \frac{1}{2}\Delta t(\Delta_h m_1^{*,n+1} m_2^n - \Delta_h m_2^{*,n+1} m_1^n) \end{pmatrix}$$

We choose an exact solution as shown in the section 4 in 1D, we get the result which is presented in table 2 for the temporal accuracy and the spatial accuracy in table 3.

Table 2: Scheme B without damping when $h = 1D-4$, $T = 1d-1$ for the temporal accuracy test in 1D.

k	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_\infty$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_2$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_{H^1}$
2.0D-2	0.027431139825021	0.021059712879716	0.162330163741064
1.0D-2	0.020622698909958	0.016102989740672	0.129004979620961
5.0D-3	0.011736902360569	0.009387929780584	0.074611897142490
2.5D-3	0.006669066410959	0.004617287861393	0.034639205691838
1.25D-3	0.003313430321005	0.002258615521905	0.016569078430246
6.25D-4	0.001655921012879	0.001120144652285	0.008173278202993
3.125D-4	8.268339362937499e-04	5.586732218835936e-04	0.004082737533487

Table 3: Scheme B without damping when $k = 1D-6$, $T = 1d-1$ for the spatial accuracy test in 1D.

h	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_\infty$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_2$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_{H^1}$
1/16	4.283245102155650e-04	2.933233567603344e-04	0.002206843323022
1/24	1.923501077731871e-04	1.310320360718346e-04	9.732798465020191e-04
1/32	1.094531449847350e-04	7.442153181987804e-05	5.445039170833464e-04
1/48	5.014106142266911e-05	3.404629946008760e-05	2.393048451167733e-04
1/64	2.936092317325917e-05	1.992700809480838e-05	1.328139595023174e-04

We choose an initial condition and set the source term to be zeros as shown in the section 4 in 1D, we get the result for the norm preserving test which is presented in table 4.

Table 4: Scheme B without damping when $h = 1D-4$, $T = 1d-1$ for the norm preserving test in 1D.

k	$\|\|\|\boldsymbol{m}_h\|\|_2 - 1\|\|_\infty$
2.0D-2	1.665334536937735e-15
1.0D-2	1.776356839400250e-15
5.0D-3	2.775557561562891e-15
2.5D-3	2.775557561562891e-15
1.25D-3	4.440892098500626e-15
6.25D-4	7.438494264988549e-15
3.125D-4	9.992007221626409e-15

For the model with damping term below,

$$\boldsymbol{m}_t = -\boldsymbol{m} \times \Delta \boldsymbol{m} - \alpha \boldsymbol{m} \times (\boldsymbol{m} \times \Delta \boldsymbol{m}). \tag{19}$$

We consider the construction below,

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \tilde{\boldsymbol{m}}_h^{n+1} - \alpha \frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \left(\boldsymbol{m}_h^n \times \Delta_h \tilde{\boldsymbol{m}}_h^{n+1} \right), \tag{20}$$

which is also a structure preserving method with $\|\boldsymbol{m}_h^{n+1}\|_2 = \|\boldsymbol{m}_h^n\|_2$. For such case, we propose the structure preserving method A below,

- Implicit iteration:

$$\frac{\tilde{\boldsymbol{m}}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = \Delta_h \tilde{\boldsymbol{m}}_h^{n+1},$$

- preserving iteration:

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \tilde{\boldsymbol{m}}_h^{n+1} - \alpha \frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \left(\boldsymbol{m}_h^n \times \Delta_h \tilde{\boldsymbol{m}}_h^{n+1} \right),$$

We choose an exact solution for the damping case as shown in the section 4 in 1D, we get the result which is presented in table 5.

Table 5: Scheme A with damping $\alpha = 0.01$ when $h = 1D-4$, $T = 1d-1$.

k	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_\infty$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_2$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_{H^1}$
2.0D-2	0.017880718906775	0.013640706794455	0.109059697329623
1.0D-2	0.011980429986506	0.008729678745185	0.069123293167653
5.0D-3	0.006781726750878	0.004446893220117	0.033156803919354
2.5D-3	0.003265847846062	0.002207758202347	0.016177581093120
1.25D-3	0.128452430721680	0.042342698779687	2.823711660952686e+02
6.25D-4	1.955912125544102	1.414290554734918	6.828775028100555e+03

In turn, we propose a new structure preserving method B for LL equation with damping term below,

Table 6: The temporal accuracy and spatial accuracy for the proposed method A with damping $\alpha = 0.01$, $T = 0.1$ with $k = h^2$ in 3D.

k	h	$\|\boldsymbol{m}_h - \boldsymbol{m}_e\|_\infty$	$\|\boldsymbol{m}_h - \boldsymbol{m}_e\|_2$	$\|\boldsymbol{m}_h - \boldsymbol{m}_e\|_{H^1}$
T/10	1/10	5.002945404173342e-04	2.886502567329273e-04	3.524885833784011e-04
T/40	1/20	1.265006422894732e-04	7.246731045058230e-05	1.356777607100581e-04
T/57	1/24	1.301505042391642e-04	5.563010756359837e-05	4.525190533793324e-04
T/78	1/28	0.006951199298437	0.001135634685630	0.021906785530406
T/102	1/32	0.741213552286026	0.137808562858426	2.714389381464438

- Implicit iteration:

$$\frac{\tilde{\boldsymbol{m}}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = \Delta_h \tilde{\boldsymbol{m}}_h^{n+1},$$

- double diffusion iteration:

$$m_i^{*,n+1} = (I - \Delta t \Delta_h)^{-1} \tilde{m}_i^{n+1}.$$

- preserving iteration:

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \boldsymbol{m}_h^{*,n+1} - \alpha \frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \left(\boldsymbol{m}_h^n \times \Delta_h \boldsymbol{m}_h^{*,n+1} \right),$$

We choose an exact solution for the damping case as shown in the section 4 in 1D, we get the result which is presented in table 7 for the temporal accuracy and the spatial accuracy in table 8.

We choose an initial condition and set the source term to be zeros as shown in the section 4 in 1D, we get the result for the norm preserving test which is presented in table 10.

3. Property of proposed method

3.1. Consistency analysis

For the local truncation error of first diffusion $\tilde{\boldsymbol{m}}_h^{n+1}$, we put the exact solution inside, and have

$$\frac{\boldsymbol{m}(t^{n+1}) - \boldsymbol{m}(t^n)}{\Delta t} = \Delta \boldsymbol{m}(t^{n+1}) + \tau_1.$$

It follows from the Taylor's expansion, we have

$$\boldsymbol{m}(t^{n+1}) = \boldsymbol{m}(t^n) + \Delta t \partial_t \boldsymbol{m}(t^n) + \frac{\Delta t^2}{2} \partial_t^2 \boldsymbol{m}(t^n) + O(\Delta t^3),$$

thus, we have

$$\begin{aligned}
\tau_1 &= \partial_t \boldsymbol{m}(t^n) + \frac{\Delta t}{2} \partial_t^2 \boldsymbol{m}(t^n) + O(\Delta t^2) - \Delta \boldsymbol{m}(t^{n+1}) \\
&= \partial_t \boldsymbol{m}(t^n) + \frac{\Delta t}{2} \partial_t^2 \boldsymbol{m}(t^n) + O(\Delta t^2) - \Delta_h \boldsymbol{m}(t^n) - \Delta t \Delta_h \partial_t \boldsymbol{m}(t^n) + O(\Delta t^2 + h^2) \\
&= O(\Delta t + h^2),
\end{aligned}$$

where we have use the heat equation $\partial_t \boldsymbol{m} = \Delta \boldsymbol{m}$.

For the local truncation error of first diffusion $\boldsymbol{m}_h^{*,n+1}$ and denote $\boldsymbol{m}_h^n = \tilde{\boldsymbol{m}}_h^{n+1}$ without any generality, we put the exact solution inside, and have

$$\frac{\boldsymbol{m}(t^{n+1}) - \boldsymbol{m}(t^n)}{\Delta t} = \Delta \boldsymbol{m}(t^{n+1}) + \tau_2.$$

Indeed, we have $\tau_2 = O(\Delta t + h^2)$.

For the third step, we put the exact solution inside, and have

$$\frac{\boldsymbol{m}(t^{n+1}) - \boldsymbol{m}(t^n)}{\Delta t} = -\frac{\boldsymbol{m}(t^{n+1} + \boldsymbol{m}(t^n))}{2} \times \Delta \boldsymbol{m}^*(t^{n+1}) + \tau_3.$$

Thus,

$$\begin{aligned}
\tau_3 &= \partial_t \boldsymbol{m}(t^n) + \frac{\Delta t}{2}\partial_t^2 \boldsymbol{m}(t^n) + O(\Delta t^2) + \left[\boldsymbol{m}(t^n) + \frac{\Delta t}{2}\partial_t \boldsymbol{m}(t^n) + O(\Delta t^2)\right] \times \Delta \boldsymbol{m}^*(t^{n+1}) \\
&= \partial_t \boldsymbol{m}(t^n) + \frac{\Delta t}{2}\partial_t^2 \boldsymbol{m}(t^n) + O(\Delta t^2) + \left[\boldsymbol{m}(t^n) + \frac{\Delta t}{2}\partial_t \boldsymbol{m}(t^n) + O(\Delta t^2)\right] \times (\Delta_h \boldsymbol{m}(t^n) + O(\Delta t + h^2)) \\
&= O(\Delta t + h^2).
\end{aligned}$$

3.2. Stability

Note that for the first diffusion step, we have

$$(I - \Delta t \Delta_h)^{-1} \tilde{\boldsymbol{m}}_h^{n+1} = \boldsymbol{m}_h^n.$$

The matrix Δ_h is negative symmetric matrix with eigenvalues $\lambda_h < 0$. The amplified factor $G_k = \frac{1}{1-\Delta t \lambda_k}$, so $|G_k| < 1$. Therefore, the first two steps are linearly and unconditionally stable.

For the single diffusion method, we have

$$\boldsymbol{m}_h^{*,n+1} = (I - \Delta t \Delta_h)^{-1} \boldsymbol{m}_h^n.$$

For the double diffusion method, we have

$$\boldsymbol{m}_h^{*,n+1} = (I - \Delta t \Delta_h)^{-1}(I - \Delta t \Delta_h)^{-1} \boldsymbol{m}_h^n = (I - \Delta t \Delta_h)^{-2} \boldsymbol{m}_h^n.$$

If we denote the eigenvalues of Δ_h as λ_k ($\lambda_k > 0$), we then have the amplified factor for single diffusion is

$$G_k^s = \frac{1}{1 + \Delta t \lambda_k},$$

and the amplified factor for double diffusion is

$$G_k^d = \left(\frac{1}{1 + \Delta t \lambda_k}\right)^2.$$

We have $|G_k^s| < 1$, $|G_k^d| < 1$ and $|G_k^d| < |G_k^s| < 1$. Thus, double diffusion method brings stronger numerical dissipation, faster error decay, and better stability than that of single diffusion.

3.3. Convergence analysis

Based on the Lax-equivalence theorem, consistency and stability to ensure that the convergence of such proposed method. Indeed, we can easily get the error estimate of first two diffusion step. Combining the stability, we can proceed a linear error estimate for the third preserving step. Rigorous convergence analysis will be given in the future work.

Remark 1. *Consider solving the heat equation $u_t = u_{xx}$ on $[0, 2\pi]$ with periodic boundary condition. Numerically, we can solve it using second order finite difference in space discretization and Euler-Backward in time.*

One can get the following error estimate:

$$\|U^n - u^n\|_{\infty,h} \leq Ct_n \left(h^2 + \Delta t\right) \max_{t \leq t_n} |u(\cdot, t)|_{C^4}, \quad \text{for } t_n \geq 0$$

where $u^n = u(\cdot, t_n)$, U^n is the numerical solution at time t_n, h is the size of the mesh and Δt is the size of time step.

Remark 2. *For Linear finite difference scheme below,*

$$u^{(k+1)} = Au^{(k)},$$

where A is the iteration matrix. Thus, we have $u^{(k+1)} = A^{k+1}u^{(0)}$. Stability means

$$\|\boldsymbol{A}^k \boldsymbol{u}^{(0)}\|_r \leq C\|\boldsymbol{u}^{(0)}\|_r, \quad \text{for all } k \geq 0 \text{ and } \boldsymbol{u}^{(0)} \in \mathbb{R}^N.$$

We define the matrix norm below,

$$\|M\|_r = \sup_{\substack{u \in \mathbb{R}^N \\ u \neq 0}} \frac{\|Mu\|_r}{\|u\|_r}.$$

Then the stability means

$$\|A^k\|_r \leq C, \quad \text{for all } k \geq 0.$$

Let $e_j^k := u_j^k - u(t_k, x_j)$ and $e^{(k)} := (e_1^k, e_2^k, \ldots, e_N^k)^T$. Assume that $u_j^0 = u_0(x_j)$. Then,

$$\lim_{\Delta t, \Delta x \to 0} \left(\sup_{t_k \leq T} \|e^{(k)}\|_r \right) = 0 \quad \text{for all } T > 0.$$

Moreover, if the scheme of order p in time and q in space, then there exists a constant $C_T > 0$ s.t.

$$\sup_{t_k \leq T} \|e^{(k)}\|_r \leq C_T \left((\Delta t)^p + (\Delta x)^q \right).$$

Indeed, we have $u^{(k+1)} = Au^{(k)}$; A: iteration matrix; $\widetilde{u}_j^k = u(t_k, x_j)$. Consistency means there exists $\epsilon^{(k)}$ s.t.

$$\widetilde{u}^{(k+1)} = A\widetilde{u}^{(k)} + (\Delta t)\epsilon^{(k)} \quad \text{and} \quad \lim_{\Delta t, \Delta x \to 0} \|\epsilon^{(k)}\|_r = 0,$$

uniformly in k. Scheme of order p in time and q in space gives

$$\|\epsilon^{(k)}\|_r \leq C \left((\Delta t)^p + (\Delta x)^q \right).$$

We have

$$e^{(k+1)} = Ae^{(k)} - \Delta t \epsilon^{(k)}.$$

By induction,

$$e^{(k)} = A^k e^{(0)} - \Delta t \sum_{l=1}^{k} A^{k-l} \epsilon^{(l-1)}.$$

Stability gives

$$\|A^k\|_r \leq C'$$

for some positive constant C'. Thus we have

$$\|e^{(k)}\|_r \leq (\Delta t) k C C' \left((\Delta t)^p + (\Delta x)^q\right) \leq T C C' \left((\Delta t)^p + (\Delta x)^q\right).$$

4. Numerical experiments

4.1. Accuracy tests

For the model problem, analytical exact solutions are derived for both one-dimensional (1D) and three-dimensional (3D) scenarios to serve as benchmarks for error quantification.

For the 1D case, the exact magnetization solution $\boldsymbol{m}_e$ is:

$$\boldsymbol{m}_e = (\cos(\cos(\pi x)) \sin t, \sin(\cos(\pi x)) \sin t, \cos t)^T,$$

while the corresponding 3D exact solution is:

$$\boldsymbol{m}_e = (\cos(XYZ) \sin t, \sin(XYZ) \sin t, \cos t)^T,$$

where $X = x^2(1-x)^2$, $Y = y^2(1-y)^2$ and $Z = z^2(1-z)^2$.

These exact solutions satisfy the governing equation **??** when the forcing term is defined as $\boldsymbol{f}_e = \partial_t \boldsymbol{m}_e + \boldsymbol{m}_e \times \Delta \boldsymbol{m}_e + \alpha \boldsymbol{m}_e \times (\boldsymbol{m}_e \times \Delta \boldsymbol{m}_e)$. They also comply with the homogeneous Neumann boundary condition, ensuring consistency with simulation constraints.

To isolate the temporal approximation error from spatial discretization effects, the spatial resolution in the 1D test is fixed at $h = 5 \times 10^{-4}$—a sufficiently fine grid that makes spatial error negligible compared to temporal error. The Gilbert damping parameter is set to $\alpha = 0.01$, and simulations run until final time $T = 0.1$. Under this configuration, the measured error primarily reflects temporal discretization inaccuracy.

The 3D temporal accuracy test faces inherent constraints from spatial resolution, as excessively fine grids incur prohibitive computational cost. To balance spatial and temporal error contributions, we adopt a coordinated refinement strategy for spatial mesh sizes (h_x, h_y, h_z) and temporal step-size (Δt) tailored to the proposed method's order: $\Delta t = h_x^2 = h_y^2 = h_z^2 = h^2 = T/N_0$. Here, N_0 is a refinement level parameter, with specific values given in subsequent results. Consistent with the 1D test, $\alpha = 0.01$, and the final time T is specified later. The first order temporal accuracy is verified from table 7 and table 9 for 1D and 3D tests, respectively.

Following temporal accuracy evaluation, spatial accuracy tests were conducted to quantify the spatial discretization performance of the proposed method. To prevent temporal errors from interfering with spatial accuracy assessment, the temporal step size was fixed at a sufficiently small $k = 10^{-6}$ for 1D test shown in table 8—making temporal errors negligible compared to spatial errors. We take $k = h^2$, such that the second order spatial accuracy is observed, which is consistent with 1D test.

Table 7: Scheme B with damping $\alpha = 0.01$ when $h = 1D - 4$, $T = 1d - 1$ for the temporal accuracy test in 1D.

k	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_\infty$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_2$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_{H^1}$
1.0D-2	0.020464314211176	0.016014615929487	0.128240703403403
5.0D-3	0.011524387531183	0.009308285127451	0.073903174126197
2.5D-3	0.006587677752975	0.004578244425849	0.034348851670859
1.25D-3	0.003282424384219	0.002240709014796	0.016450167628061
6.25D-4	0.001641316265073	0.001111333942365	0.008117312581311
3.125D-4	8.197228308458940e-04	5.541894018399869e-04	0.004053387297514
order	0.932841560262563	0.985538272716470	1.015421667518498

Table 8: Scheme B with damping $\alpha = 0.01$ when $k = 1D - 6$, $T = 1d - 1$ for the spatial accuracy test in 1D.

h	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_\infty$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_2$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_{H^1}$
1/16	4.249311877004142e-04	2.909401823360151e-04	0.002206661266589
1/24	1.909025468404060e-04	1.299542983311708e-04	9.733782583656764e-04
1/32	1.086444914674481e-04	7.380675983693032e-05	5.445742575088350e-04
1/48	4.977527242430030e-05	3.376418553190121e-05	2.393076196468634e-04
1/64	2.914764213506060e-05	1.976171193754711e-05	1.327797500654879e-04
order	1.934798641954242	1.941559387448154	2.026575238749890

Table 9: The temporal accuracy and spatial accuracy for the proposed method with damping $\alpha = 0.01$, $T = 0.1$ with $k = h^2$ in 3D.

k	h	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_\infty$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_2$	$\|\|\boldsymbol{m}_h - \boldsymbol{m}_e\|\|_{H^1}$
T/10	1/10	5.003235093097302e-04	2.886606015976042e-04	3.551360545179897e-04
T/40	1/20	1.265920847332014e-04	7.251658333899107e-05	1.378003932600100e-04
T/57	1/24	8.944032162405691e-05	5.111478167390660e-05	1.134032905670683e-04
T/78	1/28	6.579355925695474e-05	3.761556385664001e-05	9.745804260261748e-05
T/102	1/32	5.065167817441818e-05	2.905345727738149e-05	8.704136168574896e-05
T/129	1/36	4.030682599653890e-05	2.328729264189056e-05	7.986373929773676e-05
order		0.985379057530855	0.986442519896792	0.592150284932376
	order	1.966461737227700	1.968598638612654	1.181844521865077

4.2. Norm preserving tests

We choose the initial condition with below,

$$\boldsymbol{m}_0 = (\cos(\cos(\pi x))\sin(0.01), \sin(\cos(\pi x))\sin(0.01), \cos(0.01))^T,$$

and

$$\boldsymbol{m}_0 = (\cos(XYZ)\sin(0.01), \sin(XYZ)\sin(0.01), \cos(0.01))^T,$$

in 1D and 3D, respectively. The results are presented in table 10 and table 11.

Table 10: Scheme B with damping $\alpha = 0.01$ when $h = 1D-4$, $T = 1d-1$ for the norm preserving test in 1D.

k	$\|\|\|\boldsymbol{m}_h\|\|_2 - 1\|\|_\infty$
2.0D-2	1.554312234475219e-15
1.0D-2	1.998401444325282e-15
5.0D-3	2.553512956637860e-15
2.5D-3	3.330669073875470e-15
1.25D-3	5.329070518200751e-15
6.25D-4	6.994405055138486e-15
3.125D-4	1.043609643147647e-14

Table 11: The norm preserving test for the proposed method with damping $\alpha = 0.01$, $T = 0.1$ with $k = h^2$ in 3D.

k	h	$\|\|\|\boldsymbol{m}_h\|\|_2 - 1\|\|_\infty$
T/10	1/10	1.112665515279332e-12
T/40	1/20	5.366818101038007e-13
T/57	1/24	3.871347686867921e-13
T/78	1/28	2.735589532676386e-13

4.3. Initial conditions and stability tests

In this section, we choose different initial conditions specified in each test.

In 1D, we specify the initial condition as

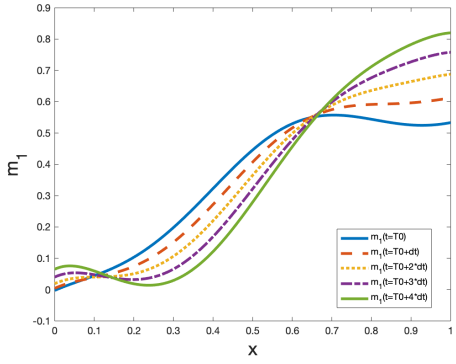
$$\boldsymbol{m}_0 = (\cos(\cos(\pi x))\sin(x), \sin(\cos(\pi x))\sin(x), \cos(x))^T$$
$$\boldsymbol{m}_0 = \left(\cos(x^2(1-x)^2)\sin(x), \sin(x^2(1-x)^2)\sin(x), \cos(x)\right)^T.$$

The results for 1D case using the proposed method to show the numerical profiles are presented in Figure 1 with the parameters $\alpha = 0.01$, $N_x = 2000$ and $N_t = 5$.

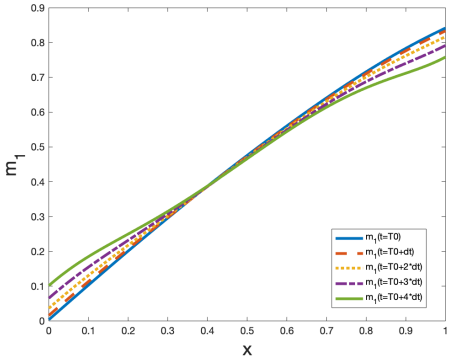
In 3D, we choose the initial conditions below,

$$\boldsymbol{m}_0 = (\cos(\cos(\pi x))\sin(0.01), \sin(\cos(\pi x))\sin(0.01), \cos(0.01))^T.$$

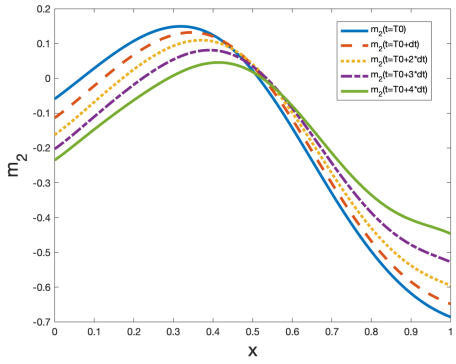
The results for those initial conditions are presented in Figure 2 which verify the consistency.



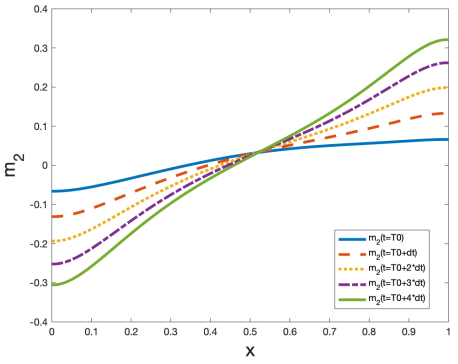
(a) m_1



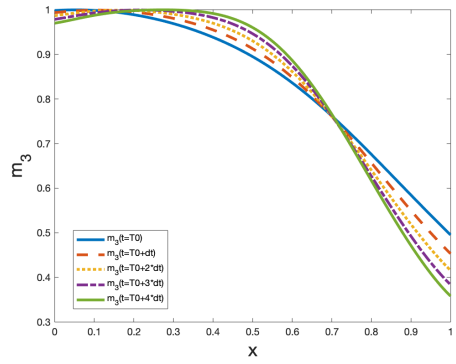
(b) m_1



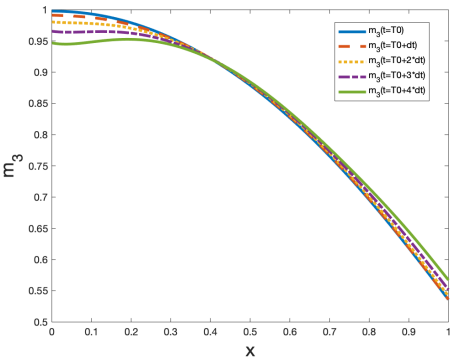
(c) m_2



(d) m_2

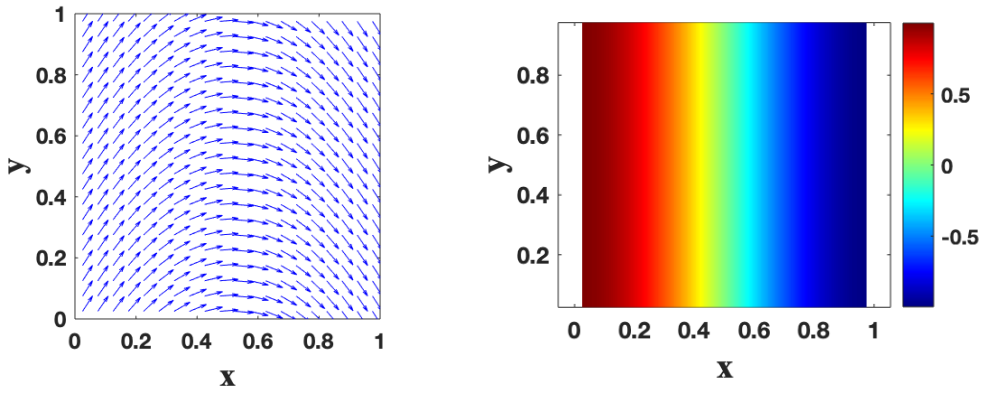


(e) m_3



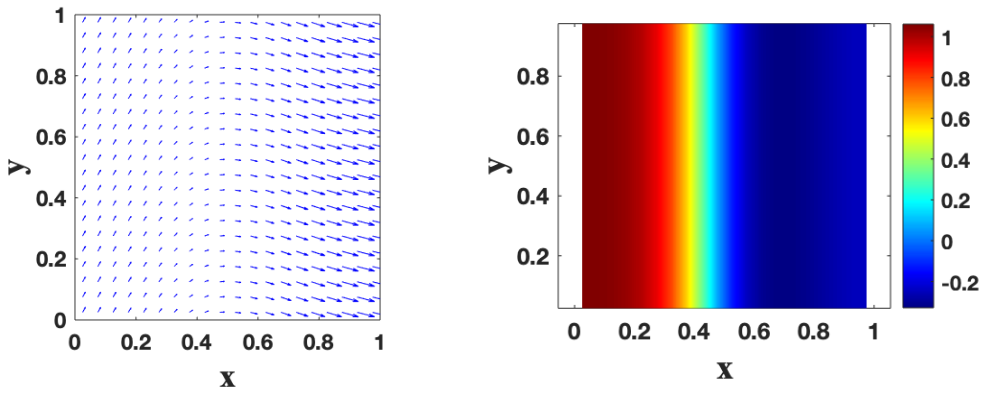
(f) m_3

Figure 1: The solution profile using proposed method in 1D given the initial condition m_0 without source term, $\alpha = 0.01$ and $T = 0.1$, $N_x = 2000$, $N_t = 5$. Left panel: the initial condition is given by $\boldsymbol{m}_0 = (\cos(\cos(\pi x))\sin(x), \sin(\cos(\pi x))\sin(x), \cos(x))^T$; Right panel: the initial condition is given by $\boldsymbol{m}_0 = \left(\cos(x^2(1-x)^2)\sin(x), \sin(x^2(1-x)^2)\sin(x), \cos(x)\right)^T$.



(a) arrow profile

(b) angle profile



(c) arrow profile

(d) angle profile

Figure 2: The solution profile using the proposed method in 3D given the initial condition m_0 with initial condition specified without source term, $\alpha = 0$ and $T = 0.1$, $N_x = N_y = N_z = 20$, $N_t = 400$. Initial condition given: $\boldsymbol{m}_0 = [\cos(\cos(\pi x))\sin(0.01), \sin(\cos(\pi x))\sin(0.01), \cos(0.01)]$

5. Conclusions and discussions

In this paper, we propose a structure preserving method with the first order accuracy in time and the second order accuracy in space. Such a method combines a first order double diffusion iteration, and a Crank-Nicolson's type iteration. The first two step with heat diffusion iteration gives a stable configuration for the higher order term. The second step to preserve the norm constraint. Such a method is constructed only based on the equation, not using a projection step. In the future work, we will analyze the stability of such a method and apply for micromagnetics simulations. We will modify such a method to be a finite element methods for the space discretization. The proposed method is stable, length preserving, accurate. Meanwhile, the convergence analysis and stability analysis for the proposed method along with the normalizing step can be proved.

For the discussion of the model, we can derive a Gilbert equation below,

$$\boldsymbol{m}_t = -\boldsymbol{m} \times \Delta \boldsymbol{m} + \alpha \boldsymbol{m} \times \boldsymbol{m}_t.$$

Thus, besides the double diffusion step, the proposed method for the norm preserving step can be written as below,

$$\frac{\boldsymbol{m}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} = -\frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \Delta_h \tilde{\boldsymbol{m}}_h^{n+1} + \alpha \frac{\boldsymbol{m}_h^{n+1} + \boldsymbol{m}_h^n}{2} \times \left(\frac{\tilde{\boldsymbol{m}}_h^{n+1} - \boldsymbol{m}_h^n}{\Delta t} \right).$$

For the discussion of the energy stability analysis, we can further develop some energy dissipation scheme combined with the Crank-Nicolson's update to preserve the norm constraint. The energy stability analysis will be explored in future work.

Acknowledgments

This work is supported in part by the Jiangsu Science and Technology Programme-Fundamental Research Plan Fund (BK20250468), Research and Development Fund of Xi'an Jiaotong Liverpool University (RDF-24-01-015).

References

[1] L. Landau, E. Lifshits, On the theory of the dispersion of magnetic permeability in ferromagnetic bodies, Phys. Z. Sowjet. 63 (1935) 153–169.

[2] T. Gilbert, Phys. Rev. 100 (1955) 1243. [Abstract only; full report, Armor Research Foundation Project No. A059, Supplementary Report, May 1, 1956 (unpublished)].

[3] X. Wang, C. García-Cervera, W. E, A Gauss-Seidel projection method for micromagnetics simulations, J. Comput. Phys. 171 (2001) 357–372.

[4] C. Xie, C. García-Cervera, C. Wang, Z. Zhou, J. Chen, Second-order semi-implicit methods for micromagnetics simulations, J. Comput. Phys. 404 (2020) 109104.

[5] Q. Cheng, J. Shen, Length preserving numerical schemes for Landau–Lifshitz equation based on Lagrange multiplier approaches, SIAM Journal on Scientific Computing 45 (2023). URL: http://dx.doi.org/10.1137/22M1501143. doi:10.1137/22m1501143.

[6] X. Li, N. Zheng, J. Shen, On a class of higher-order length preserving and energy decreasing IMEX schemes for the Landau-Lifshitz equation (2024). URL: https://arxiv.org/abs/2404.08902. doi:10.48550/ARXIV.2404.08902.