A GEOMETRIC INTEGRATION ALGORITHM WITH APPLICATIONS TO MICROMAGNETICS

D. LEWIS* AND N. NIGAM†

Abstract. The area of geometric integration theory can be employed when a time-dependant ODE or PDE with constraints needs to be solved numerically. In this paper, we present an algorithm suitable for solving a PDE arising in micromagnetics which requires that the norm of the function be conserved point-wise, and also include some comparisons with traditional methods of solving this system.

September 29, 2000

1. Introduction. In this set of notes, we describe a set of algorithms on multiple copies of S^2 (with possibly nonlinear interactions), with applications in material science. We begin by introducing a problem in micromagnetics and restating it in a setting where geometric integration methods may be applied. An arbitrary function appears in a family of geometric integration schemes for this problem; different choices of this function yield distinct discrete trajectories. We describe a possible choice that is related both to moving frames (Fels and Olver (1998, 1999) and Lewis and Olver (in preparation)) and, in the case of a geometric version of the forward Euler method, to discretization error minimization. Work is in progress to identify analogous function choices for higher order methods (Lewis, Nigam, Olver, in preparation). Finally, we present some representative numerical results that indicate that these geometric integration schemes are competitive with conventional numerical algorithms.

2. A problem in micromagnetics. In this section, we construct a time-stepping algorithm to solve the Landau-Lifschitz-Gilbert (LLG) equation, which describes the evolution of the state of magnetization $\mathbf{M}$ in a ferromagnetic sensor, occupying a region $\mathcal{B}$ in space. Typically, the magnetization is constant in a small region (termed a domain); the switching of these domains from one state to another is the basis of the functioning of read-heads in disk drives. Ideally, the entire sensor would be one large domain, which would switch instantaneously when an applied field is imposed. In practice, there are several domains, and thus the net magnetization in a desired direction is not optimal. We are interested in tracking the evolution of these domains, which involves following the local behavior of $\mathbf{M}$. As the norm of $\mathbf{M}$ is conserved pointwise in the sensor, this system is a good test of geometric time-stepping algorithms. Since the magnetization vectors have constant length, and do not undergo any translational motion, the configuration manifold for the LLG system is the manifold $\mathcal{M}$ of maps $\mathbf{M} : \mathcal{B} \to S^2$, where $\mathcal{B}$, the region occupied by the sensor, is a compact subset of $\mathbb{R}^3$. Throughout this paper we shall extend standard vector operators on $\mathbb{R}^3$ to $\mathcal{M}$ by defining them pointwise or, in the discrete model used in numerical simulations, component by component. For example, $(\mathbf{M} \times \mathbf{H})(x) = \mathbf{M}(x) \times \mathbf{H}(x)$ for all $x \in \mathcal{B}$, while $(\mathbf{M} \times \mathbf{H})_j = \mathbf{M}_j \times \mathbf{H}_j$ for all $j = 1, \ldots, n$ in the discrete case.

The LLG equation for the magnetization $\mathbf{M}$ is given by

$$\dot{\mathbf{M}} = -\frac{1}{1+\lambda^2}\mathbf{M} \times H_{\text{eff}}(\mathbf{M}) - \frac{\lambda}{1+\lambda^2}\mathbf{M} \times (\mathbf{M} \times H_{\text{eff}}(\mathbf{M})), \qquad \|\mathbf{M}(\mathbf{x})\| = 1 \ \ \forall \mathbf{x} \in \mathcal{B}. \tag{2.1}$$

Here λ is a damping parameter and H_{eff} is the effective magnetic field.

The first term on the right hand side of (2.1) describes the (undamped) *Larmor precession* of $\mathbf{M}$ about H_{eff} and is derived from first principles [2, 1]. It is observed in physical experiments, however, that changes in magnetization decay in finite time. The second term in (2.1) is a phenomenological term added to describe this damping behavior; it cannot be derived from first principle. Therefore, several choices of λ are seen in the literature. For large applied fields, the precession term does not significantly influence the qualitative long-time behavior of the system. However, in the presence of near-zero applied fields, this term causes the LLG equation to be less stiff than if it were not present. In practice, it is typically dropped from the calculations. In the present work we shall follow convention and make this approximation while conducting the analysis, but will present some numerical results which incorporate the effects of this term.

*Department of Mathematical Sciences, U. of California at Santa Cruz (lewis@math.ucsc.edu).
†IMA, U. of Minnesota, Minneapolis MN 55455 (nigam@ima.umn.edu).

The effective field is derived from energy considerations [2], and varies nonlinearly with $\mathbf{M}$. More precisely,

$$H_{\text{eff}}(\mathbf{M}) = A \triangle \mathbf{M} + \mu_0 \left(-\nabla\phi + H_{\text{app}}\right) + K(\mathbf{M} \cdot \underline{\underline{\mathbf{e}}})\, \underline{\underline{\mathbf{e}}}\,. \tag{2.2}$$

The field $A \triangle \mathbf{M}$ is called the *exchange* field, and prevents rapid variations of $\mathbf{M}$. In particular, this prevents the formation of arbitrarily fine domains. The sensor consists of ferromagnetic crystals, which have an intrinsic preferred direction of magnetization, $\underline{\underline{\mathbf{e}}}$. This effect is incorporated in the LLG model through the *uniaxial anisotropy* field $K(\mathbf{M} \cdot \underline{\underline{\mathbf{e}}})\, \underline{\underline{\mathbf{e}}}$. Here A and K are material constants. The external *applied field* is denoted by H_{app}. The nonlinear, nonlocal contributions of $\mathbf{M}$ arise through the *demagnetizing field*, $-\nabla\phi$. In order to find a potential ϕ, we need to solve the potential problem

$$\begin{array}{rcll} \triangle\phi & = & \nabla \cdot \mathbf{M}, & \text{in } \mathcal{B}, \\ \triangle\phi & = & 0, & \text{in } \mathbb{R}^3 \setminus \overline{\mathcal{B}}, \\ \phi^- & = & \phi^+, & \text{on } \partial\mathcal{B}, \\ \frac{\partial\phi^-}{\partial n} & = & \frac{\partial\phi^+}{\partial n} + \mathbf{M} \cdot \mathbf{n}, & \text{on } \partial\mathcal{B}, \\ \phi & = & o(\frac{1}{|x|}), & \text{as } |x| \to \infty. \end{array}$$

The demagnetization field makes the LLG equation both difficult to study numerically and well–suited for geometric integrators. Renormalizing $\mathbf{M}$ to conserve the norm while using conventional integrators changes the potential ϕ in a nonlinear fashion; it is easy to construct examples in which the renormalization introduces a significant change in the demagnetizing field. Moreover, the computation of this field is the dominant cost of micromagnetic simulations. It is desirable, therefore, to construct a time-stepping algorithm that will produce accurate, norm-conserving solutions with a relatively small number of function evaluations. This can be achieved by adopting either a lower-order method, or a higher order method with a larger time-step. With a conventional update, either of these approaches typically leads to significant errors in the norm of $\mathcal{M}$.

The configuration manifold $\mathcal{M}$ for the LLG system (2.1) is a homogeneous manifold on which the group $\mathcal{G}$ of smooth maps $\mathcal{Q} : \mathcal{B} \to SO(3)$ acts transitively, but not freely. Hence we can set $\mathbf{M}(x,t) = Q(x,t)\mathbf{M}_0(x)$ for some constant (with respect to time) unit vector field $\mathbf{M}_0$, and some time dependent curve $Q : \mathcal{B} \to SO(3)$. In the discretized version of the system used in numerical simulations, the relevant manifold is the direct product $(S^2)^N$ of N spheres; the direct product group $\mathcal{G} = SO(3)^N$ acts transitively on this manifold.

Let $\omega : \mathcal{B} \times \mathbb{R} \to \mathbb{R}^3$ denote the right trivialization of the time derivative of Q, i.e., the time dependent curve in the vector space $\mathcal{V}$ of maps from the sensor $\mathcal{B}$ to $\mathbb{R}^3$ satisfying $\dot{Q} = \text{skew}\,[\omega]\, Q$, where skew : $\mathbb{R}^3 \to so(3)$ satisfies skew $[\xi]\; x = \xi \times x$ for all ξ, $x \in \mathbb{R}^3$. The LLG equation is then equivalent to the equations

$$\dot{\mathbf{M}} = \dot{Q}\mathbf{M}_0 = \text{skew}\,[\omega(\mathbf{M})]\, QM_0 = \omega(\mathbf{M}) \times \mathbf{M}$$

and

$$\omega(\mathbf{M}) = \Xi(\mathbf{M}) + \sigma(\mathbf{M})\mathbf{M}, \qquad \text{where} \qquad \Xi(\mathbf{M}) = \lambda\mathbf{M} \times H_{\text{eff}}(\mathbf{M}) \tag{2.3}$$

and $\sigma : \mathcal{M} \times \mathbb{R} \to \mathbb{R}$ is an arbitrary scalar function.

We can regard (2.3) as defining a family of ODEs

$$\dot{g} = \text{skew}\,[\omega(g \cdot \mathbf{M}_0)]\, g$$

parametrized by $\mathbf{M}_0 \in \mathcal{M}$ on the group $\mathcal{G}$ and, in the discrete case, use the techniques developed for geometric integration on Lie groups to determine approximate discrete solution curves of this ODE (see [5, 6] and the references therein). Using the action of $\mathcal{G}$ on $\mathcal{M}$ to map these discrete trajectories in $\mathcal{G}$ onto $\mathcal{M}$ yields a geometric integration scheme for the discretized LLG system. Given a (right) trivialized form $\dot{g} = \xi(g)g$ of an ODE on a Lie group G and an 'algorithmic exponential $\Phi : \mathfrak{g} \to G$ approximating the true exponential map of G, an update of the form $g_{n+1} = \Phi(\Delta t\, \xi_n)g_n$, where ξ_n is some approximation of $\xi(g_n)$ depending on g_n and the time-step Δt, determines a one step method on G. If G acts transitively on a manifold $\mathcal{M}$, then

an ODE on $\mathcal{M}$ can be expressed in the form $\dot{m} = (\tilde{\xi}(m))_{\mathcal{M}}(m)$, where $\tilde{\xi}_{\mathcal{M}}$ denotes the infinitesimal generator determined by $\tilde{\xi}$, with associated update $m_{n+1} = \Phi(\Delta t\, \tilde{\xi}_n) \cdot m_n$.

For the rotation group $SO(3)$ both the true matrix exponential and the Cayley transform $\mathrm{cay}(\xi) = (I + \mathrm{skew}\,[\xi/2])(I - \mathrm{skew}\,[\xi/2])^{-1}$ can be efficiently evaluated and have frequently been used as algorithmic exponentials. (We identify $so(3)$ with $\mathbb{R}^3$ using the map skew.) Here we use the Cayley transform, making use of the particularly simple expression

$$\mathrm{cay}(\nu)\mathbf{M} = \mathbf{M} + \frac{1}{1 + |\nu/2|^2}\left[-\nu \times \mathbf{M} + \frac{1}{4}\nu \times (\nu \times \mathbf{M})\right].$$

Thus we consider updates of the form

$$\mathbf{M}_{n+1} = \Phi(\triangle t\, \omega_n)\mathbf{M}_n, \tag{2.4}$$

where $\Phi : (\mathbb{R}^3)^N \to (SO(3))^N$ satisfies $(\Phi(\xi))_j = \mathrm{cay}(\xi_j)$.

Using (2.4), we now define one-step methods that are natural analogs of the standard explicit and implicit Euler methods:

$$\omega_n = \begin{cases} \Xi(\mathbf{M}_n) + \sigma(\mathbf{M}_n)\mathbf{M}_n & \text{forward Euler,} \\ \tilde{\Xi}(\triangle t, \mathbf{M}_n) + \sigma(\mathbf{M}_n)\mathbf{M}_n & \text{implicit Euler,} \end{cases} \tag{2.5}$$

where $\tilde{\Xi}(\triangle t, \mathbf{M})$ denotes the solution of the implicit equation $\xi = \Xi(\Phi(\triangle t\, \xi)\mathbf{M})$ and the scalar functions σ are as yet unspecified.

The flexibility in the choice of map σ to be used in the update (2.4, 2.5) arises from the implementation of an update on a sphere using an update in the rotation group $SO(3)$, combined with the action of $SO(3)$ on the sphere. The three dimensional group $SO(3)$ acts transitively, but not freely, on the two dimensional sphere S^2; distinct ODEs

$$\dot{g} = \mathrm{skew}\,[\omega(g \cdot m_0)]\, g \qquad \text{and} \qquad \dot{g} = \mathrm{skew}\,[\tilde{\omega}(g \cdot m_0)]\, g,$$

where $\omega(m) - \tilde{\omega}(m) \in \mathrm{span}[m]$ for all $m \in S^2$, will typically have distinct solution curves in $SO(3)$, but the images in S^2 of those solution curves under the map $g \mapsto g \cdot m_0$ will coincide. However, when the solution curves in $SO(3)$ are approximated using a time discretization, the images of those discrete trajectories typically do not agree. In section 5 we provide numerical results showing the effect of the parameter σ on the trajectories taken in case of the forward Euler algorithm. In particular, larger values of σ cause the trajectories to diverge sharply from those of the ordinary forward Euler; however, the final state is the same. A closer look at the equation shows that a larger value of σ corresponds to the inclusion of more precession in the trajectory. These numerical results clearly show that different choices of the parameter σ lead to significantly different numerical trajectories and thus motivate the search for an "optimal" value of σ. In the following section we describe a general geometric approach to selecting values for σ; in section 4 we show that when used with the forward Euler method, this choice of σ minimizes the discretization error. (See Lewis and Olver (in preparation) for a description of this approach for more general homogeneous manifolds.)

3. Moving frames. In what follows, let $\mathcal{M}$ denote a manifold acted on by a Lie group G. A *moving frame* is an equivariant map $\rho : \mathcal{M} \to G$ i.e., a map satisfying

$$\rho(g \cdot m) = g\rho(m) \text{ (a left moving frame), or} \qquad \rho(g \cdot m) = \rho(m)g \text{ (a right moving frame).}$$

(See Fels and Olver (1998, 1999), for a comprehensive treatment of moving frames as equivariant maps.)

If G acts freely and transitively on $\mathcal{M}$, then each curve $m(t)$ on $\mathcal{M}$ determines a unique curve $g(t) \in G$ such that $m(t) = g(t)m(0)$. If we are given a left (resp. right) moving frame on $\mathcal{M}$, then $g(t) = \rho(m(t))\rho(m(0))^{-1}$ (resp. $g(t) = \rho(m(0))^{-1}\rho(m(t))$). In particular, an ODE $\dot{m} = X(m)$ on $\mathcal{M}$, a (left) moving frame ρ on $\mathcal{M}$, and a point $m \in \mathcal{M}$ determine an associated ODE

$$\dot{g} = T\rho\, X(g \cdot m)\rho(m)^{-1}$$

on G. (See Lewis and Olver (in preparation) for a discussion of the use of moving frames in determining Lie group methods for numerical simulations.)

The group of interest here is the rotation group $SO(3)$, which acts transitively on the sphere S^2 and freely on the tangent planes $\mathcal{S} = TS^2 \setminus \{\text{zero section}\}$. To find a moving frame $\rho : \mathcal{S} \to SO(3)$, we begin with $\mathbf{m} \in S^2$, and a nonzero vector $\delta\mathbf{m} \in T_m S^2$. We define

$$\rho(\mathbf{m}, \delta\mathbf{m}) := \begin{pmatrix} \mathbf{m} \\ \nu(\mathbf{m} \times \delta\mathbf{m}) \\ -\nu(\delta\mathbf{m}) \end{pmatrix}^T, \qquad \text{where} \qquad \nu(\mathbf{x}) := \frac{\mathbf{x}}{\|\mathbf{x}\|}.$$

(Note that $\delta\mathbf{m} = \mathbb{P}_m \delta\mathbf{m} = -\mathbf{m} \times (\mathbf{m} \times \delta\mathbf{m})$, where $\mathbb{P}_m : \mathbb{R}^3 \to T_m S^2$ denotes orthogonal projection onto the plane orthogonal to $f.$). For any rotation $Q \in SO(3)$,

$$\rho(Q \cdot (\mathbf{m}, \delta\mathbf{m})) = \rho(Q\,\mathbf{m}, Q\,\delta\mathbf{m}) = \begin{pmatrix} Q\mathbf{m} \\ Q\,\nu(\mathbf{m} \times \delta\mathbf{m}) \\ -Q\nu(\delta\mathbf{m}) \end{pmatrix}^T = Q\rho(\mathbf{m}, \delta\mathbf{m}).$$

This establishes that ρ is a left moving frame. In this case, $\rho(\mathbf{m}, \delta\mathbf{m})$ is simply the orthogonal matrix determined by the positively oriented (orthonormal) basis $\{e_1, e_2, e_3\} = \{\mathbf{m}, \nu(\mathbf{m} \times \delta\mathbf{m}), -\nu(\delta\mathbf{m})\}$ Using the construction described above, we can define a moving frame $\mathcal{R} : \mathcal{P} \to \mathcal{G}$, where

$$\mathcal{P} := \left\{(\mathbf{M}, \delta\mathbf{M}) \in T(S^2))^N) : \delta\mathbf{M}_j \neq 0, j = 1...N\right\} \approx \mathcal{S}^N$$

or, in the continuous case,

$$\mathcal{P} := \left\{(\mathbf{M}, \delta\mathbf{M}) \in T\mathcal{M} : \delta\mathbf{M}(x) \neq 0, \forall x \in \mathcal{B}\right\}.$$

Given an ODE $\dot{\mathbf{M}} = X(\mathbf{M})$ on $\mathcal{M}$, the moving frame $\mathcal{R}$ can be used to determine an ODE on $\mathcal{G}$ and thus a geometric integration scheme on $\mathcal{M}$. We pull back the map $\mathcal{R}$ to the submanifold

$$\widetilde{\mathcal{M}} := \{\mathbf{M} \in \mathcal{M} : X(\mathbf{M}) \in \mathcal{P}\}.$$

Note that $\mathcal{M} \backslash \widetilde{\mathcal{M}}$ consists precisely of the equilibria of the ODE; such points can naturally be treated as fixed points of the discrete time update. Let $\Xi : \mathcal{M} \to (\mathbb{R}^3)^N$ be a map satisfying $(X(\mathbf{m}))_j = \Xi(\mathbf{M})_j \times \mathbf{m}_j$, for $j = 1, \ldots, N$. We fix some index $j, 1 \leq j \leq N$, and set $\mathbf{m} = \mathbf{M}_j, \delta\mathbf{m} = \delta\mathbf{M}_j, \xi = \Xi(\mathbf{M})_j, U = (\mathcal{R}(\mathbf{M}, \delta\mathbf{M}))_j$, (with the obvious definitions for the continuous case). If ζ is the j–th component of $(D\Xi(\mathbf{M}) \cdot \delta\mathbf{M})$, then

$$(T_M X \cdot \delta\mathbf{M})_j = (\mathbf{m}, \delta\mathbf{m}, \xi \times \mathbf{m}, \tau), \qquad \tau := \zeta \times \mathbf{m} + \xi \times \delta\mathbf{m}.$$

Setting $\chi := \|\delta\mathbf{m}\| = \|\mathbf{m} \times \delta\mathbf{m}\|$, $\xi_i := \langle \xi, e_i \rangle$, and $\tau_i := \langle \tau, e_i \rangle, i = 1, 2, 3$, and using the identities

$$D\left(\frac{f}{\|f\|}\right)(\mathbf{m}) = \frac{1}{\|f\|}\frac{\partial f_i}{\partial m_j} - \frac{1}{\|f\|^3} f_j \sum_{k=1}^{N} f_k \frac{\partial f_k}{\partial m_j} = \frac{1}{\|f(m)\|}\mathbb{P}_{f(m)} Df(m),$$
$$\tau = -\xi_2 e_1 + (\xi_1 \chi + \zeta_3) e_2 - \zeta_2 e_3,$$

we obtain

$$\begin{aligned} T_m(\rho \circ X)(\mathbf{m}, \delta\mathbf{m}) &= \begin{pmatrix} \xi \times \mathbf{m} \\ \frac{1}{\chi}\mathbb{P}_{\mathbf{m} \times \delta\mathbf{m}}((\xi \times \mathbf{m}) \times \delta\mathbf{m} + \mathbf{m} \times \tau) \\ -\frac{1}{\chi}\mathbb{P}_{\delta\mathbf{m}}\tau \end{pmatrix}^T \\ &= \begin{pmatrix} \xi_3 e_2 - \xi_2 e_3 \\ \frac{1}{\chi}\mathbb{P}_{e_2}(-\xi_3 e_1 - \xi_2 e_2 + \tau_2 e_3 - \tau_3 e_2) \\ -\frac{1}{\chi}\mathbb{P}_{e_3}\tau \end{pmatrix}^T \\ &= U \begin{pmatrix} 0 & \xi_3 & -\xi_2 \\ -\xi_3 & 0 & \frac{\tau_2}{\chi} \\ -\frac{\tau_1}{\chi} & \frac{\tau_2}{\chi} & 0 \end{pmatrix}^T \\ &= U\,\mathrm{skew}\left[\left(\xi_1 + \frac{\zeta_3}{\chi}, \xi_2, \xi_3\right)\right]. \end{aligned}$$

Thus, since $U \operatorname{skew}[x] U^T = \operatorname{skew}[Ux]$ for any $U \in SO(3)$ and $x \in \mathbb{R}$, we have

$$T_m(\rho \circ X)(\mathbf{m}, \delta\mathbf{m}) U^T = \operatorname{skew}\left[U\left(\xi_1 + \frac{\zeta_3}{\chi}, \xi_2, \xi_3\right)\right] = \operatorname{skew}\left[\xi - \frac{\langle \zeta, \delta\mathbf{m}\rangle}{\|\delta\mathbf{m}\|^2}\mathbf{m}\right]. \tag{3.1}$$

We now specialize to the case $\delta\mathbf{M} = X(\mathbf{M})$. If $\mathbf{M}(t)$ is a solution curve of the equation $\dot{\mathbf{M}} = X(\mathbf{M})$, and $\mathbf{m}(t) = (\mathbf{M}(t))_j$, then

$$\ddot{\mathbf{m}} = \dot{\xi} \times \mathbf{m} + \xi \times (\xi \times \mathbf{m}) = \zeta \times \mathbf{m} - \|\xi\|^2 \mathbf{m} + \langle \xi, \mathbf{m}\rangle \xi.$$

In this case, the projection of ζ onto $T_m S^2$ is $\mathbb{P}_m \zeta = \mathbf{m} \times \ddot{\mathbf{m}} + \langle \xi, \mathbf{m}\rangle \dot{\mathbf{m}}$; hence

$$\langle \zeta, \dot{\mathbf{m}}\rangle = \langle \mathbb{P}_m \zeta, \dot{\mathbf{m}}\rangle = \langle \mathbf{m} \times \ddot{\mathbf{m}}, \dot{\mathbf{m}}\rangle + \langle \xi, \mathbf{m}\rangle \|\Delta t \mathbf{m}\|^2. \tag{3.2}$$

Comparison with (3.1) shows that

$$\left(T_M(\mathcal{R} \circ X)(X(\mathbf{M}))\mathcal{R}(X(\mathbf{M}))^T\right)_j = \operatorname{skew}\left[\mathbb{P}_m \xi - \frac{\langle \mathbf{m} \times \ddot{\mathbf{m}}, \dot{\mathbf{m}}\rangle}{\|\dot{\mathbf{m}}\|^2}\mathbf{m}\right]. \tag{3.3}$$

If $g(t)$ is a solution curve of the ODE $\dot{g} = \Omega g$, where $\Omega \in so(3)$ is given by (3.3), then $\mathbf{m}(t) = g(t)\mathbf{m}(0)$. We can regard (3.3) as determining a choice of the function σ; specifically, if $\mathbb{P}_M \Xi = \Xi$, as in the micromagnetism system, then

$$\sigma(\mathbf{M})_j = \frac{\langle \mathbf{m} \times \ddot{\mathbf{m}}, \dot{\mathbf{m}}\rangle}{\|\dot{\mathbf{m}}\|^2} \tag{3.4}$$

The variation of σ as time progresses in some sample numerical runs is described in section 5. Note that (3.4) is equivalent to $\sigma(\mathbf{M})_j = K_g \|\dot{\mathbf{m}}\|$, where K_g denotes the geodesic curvature of $\mathbf{m}(t)$. For more details on this work, we refer to ([3],[4], Lewis and Olver (in preparation)).

4. Optimal choice of parameter σ for the forward Euler method. Using the notation developed in the section on moving frames, we see that the normal component of $\xi = \Xi(\mathbf{M})$ does not influence the solution curves of the original ODE. Thus, if we have a numerical algorithm of order n, this component does not affect the solution up to order n. However, it typically does appear in the higher order terms of the approximation, and theoretically a suitable choice of this component will reduce the discretization error. The discussion on moving frames suggests a choice of σ in the algorithms (2.4, 2.5). For the forward Euler scheme this choice of σ is optimal in the sense that it minimizes the discretization error, as we shall now show. This is a particular example of a more general result covering a large class of manifolds and higher order algorithms. (See Lewis and Olver (in preparation).) Work is in progress (Lewis, Nigam, Olver) to possibly extend these or related results to an even larger class of systems, including the full discretized LLG system.

We consider consistent algorithms using standard methods on the tangential component of ξ. The normal component is treated as a function of the tangential; we shall see that a component of the local discretization error at second order can be eliminated by a suitable choice of the normal component of the lowest order term in ξ.

We begin by examining the flow $\mathcal{F}_t$ of the ODE $\dot{\mathbf{M}} = \omega(\mathbf{M}) \times \mathbf{M}$ on $\mathcal{M}$ determined by a map ω satisfying $(\omega(\mathbf{M}))_j \in T_{m_j} S^2$ for $j = 1, \ldots, N$. The flow of this ODE satisfies

$$\mathcal{F}_t(\mathbf{M}) = \mathbf{M} + \triangle t\, \omega \times \mathbf{M} + \frac{\triangle t^2}{2}\left(\omega \times (\omega \times \mathbf{M}) + \dot{\omega} \times \mathbf{M}\right) + \mathcal{O}(\Delta t^2).$$

If the algorithmic update $\mathcal{F}_{\triangle t} : \mathcal{M} \to \mathcal{M}$, is given by

$$\mathcal{F}_{\triangle t}(\mathbf{M}) := \Phi\left(\triangle t\, \Xi(\mathbf{M}, \triangle t)\right) \cdot \mathbf{M},$$

where $\Xi(\mathbf{M}, \triangle t) := \sum_{j=0}^{\infty} \xi_j(\mathbf{M}) \frac{\triangle t^j}{(j+1)!}$ (note the nonstandard scaling of the coefficients in the expansion) and $\Phi : \mathbb{R}^3 \approx so(3) \to SO(3)$ agrees with the exponential map to second order (e.g. Φ is the Cayley transform),

then

$$\begin{aligned}\mathcal{F}_{\triangle t}(\mathbf{M}) &= \left(I + \triangle t\,\mathrm{skew}\,[\Xi] + \frac{1}{2}\triangle t^2 \mathrm{skew}\,[\Xi]^2 + \mathcal{O}(\Delta t^3)\right)\mathbf{M} \\ &= \left(I + \triangle t\,\mathrm{skew}\,[\xi_0] + \frac{1}{2}\triangle t^2\left(\mathrm{skew}\,[\xi_1] + \mathrm{skew}\,[\xi_0]^2\right) + \mathcal{O}(\Delta t^3)\right)\mathbf{M} \\ &= \mathbf{M} + \triangle t\,\xi_0 \times \mathbf{M} + \frac{\triangle t^2}{2}\left(\xi_0 \times (\xi_0 \times \mathbf{M}) + \xi_1 \times \mathbf{M}\right) + \mathcal{O}(\Delta t^3).\end{aligned}$$

The consistency condition for $\mathcal{F}_{\triangle t}$ is $\mathbb{P}_{\mathbf{M}}\xi_0 = \omega$. If $\mathcal{F}_{\triangle t}$ is consistent, then, setting $\sigma_0 := \langle \xi_0, \mathbf{M}\rangle$, the local discretization error is

$$\begin{aligned}\frac{\mathcal{F}_{\triangle t} - \mathcal{F}}{\triangle t} &= \frac{\triangle t}{2}\left(-\sigma_0 \mathbf{M} \times (\xi_0 \times \mathbf{M}) + (\xi_1 - \dot{\omega}) \times \mathbf{M})\right) + \mathcal{O}(\Delta t^2) \\ &= \frac{\triangle t}{2}\left(-\sigma_0 \omega + (\xi_1 - \dot{\omega}) \times \mathbf{M})\right) + \mathcal{O}(\Delta t^2).\end{aligned}$$

The algorithm is thus second–order accurate iff

$$\langle \xi_1 - \dot{\omega}, \omega\rangle = 0 \qquad \text{and} \qquad \sigma_0\langle \omega, \omega\rangle = \langle (\xi_1 - \dot{\omega}) \times \mathbf{M}, \omega\rangle. \tag{4.1}$$

In our geometric version of the forward Euler method, $\xi_j = 0$ for $j > 0$; thus this method will not be second order. However, we are free to choose σ_0 so as to satisfy the second equality in (4.1), yielding (3.4).

5. Some preliminary numerical investigations. In this section, we describe a few experiments which were performed to examine the feasibility of using this geometric time-stepping method. Some of the results are shown in figures 5.1–3. (These numerical studies were conducted using MATLAB.)

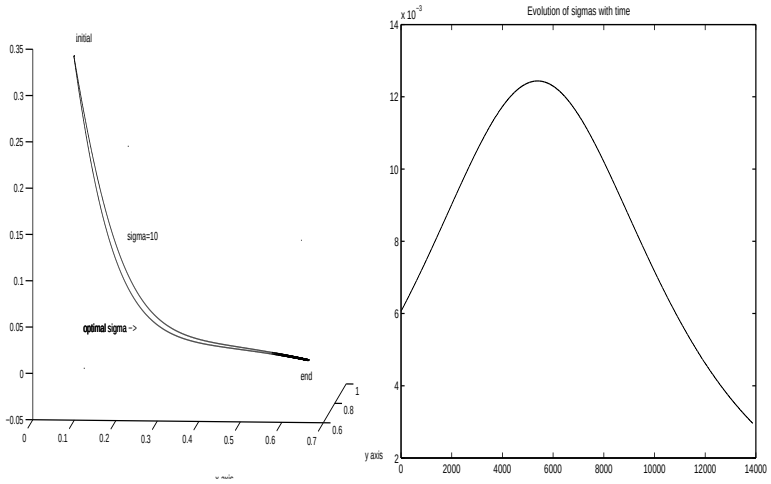

FIG. 5.1. *The trajectories followed by the usual forward Euler* $\Delta t = 0.0001$ *and with the geometric forward Euler* ($\Delta t = 0.01$) *with optimal* σ *are almost identical; if we assume* $\sigma = 10$, *the trajectory precesses more before reaching the final point. The figure on the right shows the evolution of the optimal* σ. *The applied field is uniform and weak,* $H_{\mathrm{app}} = (0.05, 0.05, 0)$

As part of the numerical experiments, we studied the effect of choosing various parameters σ in a forward-Euler implementation of the LLG equation. We then tracked the evolution of the choice of σ given by (3.4). We approximate the acceleration $\ddot{\mathbf{M}}_n$ in (3.4) using a one-sided derivative of $\dot{\mathbf{M}}_n = X_n \times \mathbf{M}_n$. While most of the experiments involved the simplified LLG equation with zero damping, i.e. $\lambda = 0$, we performed some experiments using the full LLG equations. We expected that with the addition of the precession term, the trajectories would twist more; since the parameter σ was related to the curvature of the trajectory, we expected that it would change more when λ was nonzero. This was borne out by the numerical experiments.

The industry standard employs fourth-order Runge-Kutta methods; these conserve the norm to fifth order. For comparison, we describe the geometric analogue of the fourth-order Runge-Kutta algorithm:

$$M_{n+1} = \Phi(\Delta t\, \omega_n)\mathbf{M}$$

where

$$\omega_n = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) + \sigma_n \mathbf{M}_n$$

$$\begin{aligned} k_1 &= X(\triangle t, M_n, \tilde{X}_n) \\ k_2 &= X(\triangle t, M_n, \tilde{X}_n + \frac{\triangle t}{2} k_1) \\ k_3 &= X(\triangle t, M_n, \tilde{X}_n + \frac{\triangle t}{2} k_2) \\ k_4 &= X(\triangle t, M_n, \tilde{X}_n + \triangle t\, k_3) \end{aligned}$$

and $\tilde{X}_n = \tilde{X}(\mathbf{M}_n)$. To date, for the sake of computational speed, we have only implemented the fourth-order Runge-Kutta algorithm with $\sigma = 0$. Future work includes the development of efficient methods for computing the appropriate choices of σ for higher order algorithms. We conjecture that even without an explicit computation of the refining parameter σ, the geometric integrator is competitive while preserving the norms to all orders.

In the numerical experiments presented here, we use a one-dimensional ferromagnetic slab, with variations of $\mathbf{M}$ only along one direction. This enabled us to find an analytic expression for the demagnetizing field, which in turn helped us locate the exact solution of the (LLG) equation. In most applications, this option is not available to us; indeed, it is the lack of precise analytical results for comparison which makes numerical micromagnetics a challenging field. In our examples, we chose the largest possible time-steps for a given method which would lead the system to the analytic solution (within prescribed tolerance).

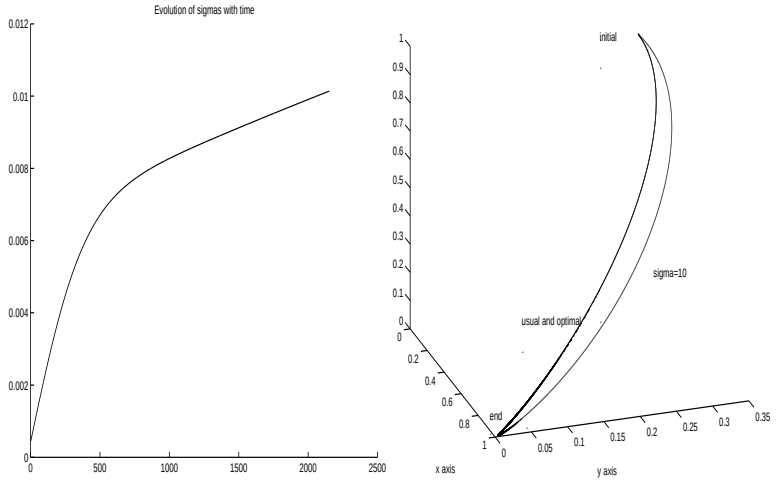

FIG. 5.2. *Shown here is the evolution of one point in the ferromagnetic sample. The trajectories followed by the usual forward Euler* $\Delta t = 0.0001$ *and with the geometric forward Euler* ($\Delta t = 0.01$) *with optimal* σ *are almost identical; if we assume* $\sigma = 10$*, the trajectory precesses more before reaching the final point. The figure on the right shows the evolution of the optimal* σ*. The applied field is uniform,* $H_{\text{app}} = (5, 0., 0)$

Figures (5.1,5.2) describe the trajectories followed by the overdamped LLG system (without precession term) for two different applied fields. We present, for comparison, trajectories computed using the usual forward Euler method with time step $\Delta t = 0.0001$, and the geometric forward Euler method with a much

larger time step $\Delta t = 0.01$. Both trajectories end at the same final point. As we are only interested in the final equilibrium state of the system, we see the obvious merit of using the geometric integrator — accurate final solution with much larger time steps. We also show the effect of varying scalar functions σ on the trajectories. In one case, we set $\sigma = 10$, in the other, we use the equation (3.4). The latter trajectory is almost indistinguishable from the forward Euler trajectory with a much smaller time step, while the larger σ trajectory precesses more before arriving at the same final state. We notice certain trends in the function $\sigma(t)$ in both figures; these bear further investigation.

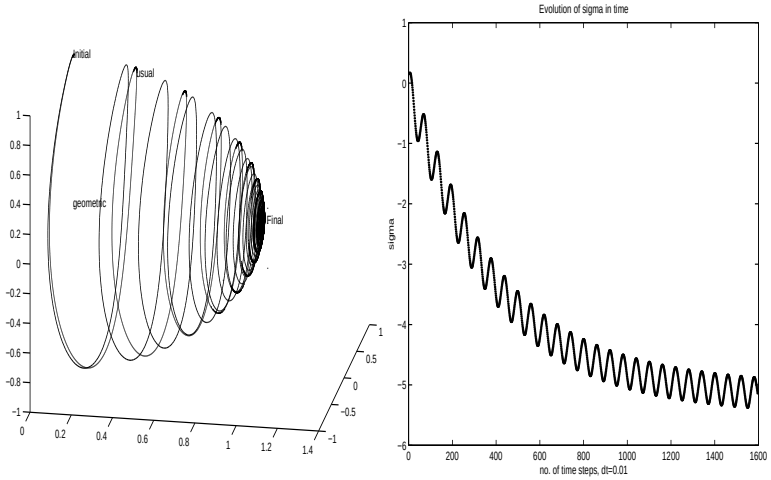

FIG. 5.3. *An example with the full LLG system.*

In figure (5.3,5.4), we implemented the code for the full LLG system, including precession. The trajectories followed by the usual forward Euler $\Delta t = 0.0001$ and with the geometric forward Euler ($\Delta t = 0.01$) with optimal σ diverge appreciably, yet end at the same final state. The drift of the norm is now clearly visible (see figure (5.4)); while the usual forward-Euler trajectory moves off the unit sphere, the geometrically integrated one does not. We see that the optimal σ now varies more (figure (5.3). The applied field is uniform, $H_{\text{app}} = (5, 0, 0)$. The damping parameter λ was set to a low value, specifically $\lambda = 0.05$.

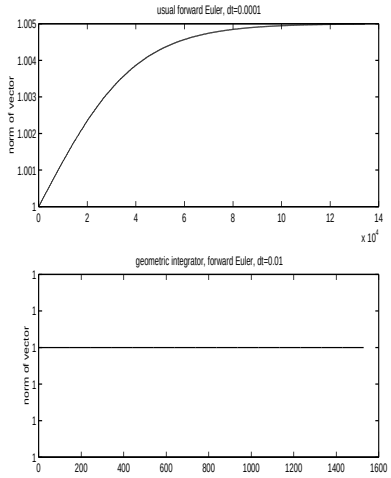

FIG. 5.4. *Norm of test vector. The geometric integrator exactly preserves the norm, even with a time step of* 0.01. *The usual forward Euler method shows a drift in norm, even with a time-step of* 0.0001.

In the last experiment, we varied the time-step Δt for a fixed applied field, in order to study possible variations of σ. Encouragingly, the qualitative behavior of the σ remained the same; leading us to the

conclusion that σ is related to the local geometry of the trajectory rather than to the time taken to get to a particular point on it. Since this figure is rather dense, we have not included it here; however, it is available at *http://www.ima.umn.edu/~nigam/graph.eps*.

6. Conclusions. The previous discussions, and the numerical experiments raise several interesting issues which the authors are currently investigating. In particular,

1. Stability of these algorithms
2. Possible operation-saving modifications
3. Less computationally-intensive choices of σ
4. Comparison against standard methods while computing hysteresis loops
5. Higher-order corrections of orbits for larger systems, and their possible links to moving frame theory.

As mentioned, this is currently work in progress.

REFERENCES

[1] A. Aharoni, *Introduction to the theory of ferromagnetism*, in Monographs on Physics, Oxford University Press, 1996.

[2] W. Brown, *Micromagnetics*, New York Interscience, 1963.

[3] M. Fels and P. Olver, *Moving coframes. i. a practical algorithm*, Acta Appl. Math, 51 (1998), pp. 161–213.

[4] M. Fels and P. Olver, *Moving coframes. ii. regularization and theoretical foundations*, Acta Appl. Math., 55 (1999), pp. 127–208.

[5] D. Lewis and J. C. Simo, *Conserving algorithms for the dynamics of Hamiltonian systems on Lie groups*, J. Nonlinear Sci., 4 (1994), pp. 253–299.

[6] Munthe-Kaas, *Runge-kutta methods on lie groups*, BIT, 38 (1998), pp. 92–111.