LaBonte's method revisited: An effective steepest descent method for micromagnetic energy minimization

Lukas Exl, Simon Bance, Franz Reichel, Thomas Schrefl, Hans Peter Stimming, and Norbert J. Mauser

LaBonte's method revisited: An effective steepest descent method for micromagnetic energy minimization

Lukas Exl,[1,a] Simon Bance,[2] Franz Reichel,[2] Thomas Schrefl,[2] Hans Peter Stimming,[1] and Norbert J. Mauser[3]

[1]*Fakultät für Mathematik, Universität Wien, A-1090 Vienna, Austria*
[2]*St. Poelten University of Applied Sciences, Matthias Corvinus Str. 15, 3100 St. Poelten, Austria*
[3]*Wolfgang Pauli Institute c/o Fakultät für Mathematik, Universität Wien, A-1090 Vienna, Austria*

We present a steepest descent energy minimization scheme for micromagnetics. The method searches on a curve that lies on the sphere which keeps the magnitude of the magnetization vector constant. The step size is selected according to a modified Barzilai-Borwein method. Standard linear tetrahedral finite elements are used for space discretization. For the computation of quasistatic hysteresis loops, the steepest descent minimizer is faster than a Landau-Lifshitz micromagnetic solver by more than a factor of two. The speed up on a graphic processor is 4.8 as compared to the fastest single-core central processing unit (CPU) implementation. © 2014 AIP Publishing LLC. [http://dx.doi.org/10.1063/1.4862839]

I. INTRODUCTION

A. Hysteresis in micromagnetics

Energy application and the quest for rare-earth free or rare-earth reduced permanent magnets[8,16] renewed the interest in micromagnetics of permanent magnets.[15] Most state-of-the-art micromagnetic solvers integrate the Landau-Lifshitz Gilbert (LLG) equation of motion in time. However, the accessible time scale of micromagnetics simulations is in the range of nanoseconds. This time scale is irrelevant for permanent magnet applications. The measurement time for hysteresis loops of permanent magnets is in the range of seconds. Therefore, micromagnetic solvers that minimize the energy directly instead of solving a time dependent equation might be more suitable for the analysis for magnetization reversal in permanent magnets.

Hysteresis in non-linear system results from the path formed by subsequent local minima. Kinderlehrer and Ma[9] computed hysteresis loops in ferromagnets from the continuation of solutions for decreasing and increasing applied fields. Due to the constraint, that the magnetization vector has to keep its length, a constrained non-linear optimization problem at each point of the hysteresis loop has to be solved.

In permanent magnet applications (e.g., wind turbines, electric motors), the sweeping time of the external field is in the range of seconds, which is orders of magnitude lower than the gyromagnetic precession (in contrast to, e.g., magnetic recording). Thus, we believe that the precession term can be dropped. So, in this paper, we do not compare the phenomenological equation of motion with energy minimization, rather than a numerical optimization approach with (adaptive) time integration, both, for Eq. (9).

B. Minimization techniques

Originally, LaBonte[11] computed equilibrium magnetic states in micromagnetics numerically. He used a finite

difference scheme, in order to discretize the Gibbs free energy of a Bloch-wall in a ferromagnetic film. At each step of an iterative procedure, he selected a computational cell and rotated the magnetization vector of the cell in the direction of the effective field: The normalized magnetization vector is replaced by the normalized effective field. Kosavisutte and Hayashi[10] showed that the original LaBonte method can be accelerated by the use of over- and under-relaxation. These methods rotate the magnetization towards the effective field. The effective field is proportional to the gradient of the energy. Therefore, these methods are steepest descent minimization schemes with a particular choice of step-length and a subsequent normalization step.

In unconstrained minimization, the use of conjugate directions rather than steepest descent direction method improves the convergence. Cohen and co-workers[4] introduced a conjugate gradient method for the computation of molecular orientation of liquid crystals. The orientation vectors in liquid crystals have a fixed length similar to the magnetization in ferromagnets. There are two modifications to the classical conjugate gradient method: (1) The gradients are projected onto a plane normal to the current orientation vector. (2) After the line search, the solution vector is normalized. In micromagnetics, a similarly modified conjugate gradient method has been applied, Viallix and co-workers[19] computed Bloch walls and Bloch lines in ferromagnetic films. Alouges and co-workers[1] computed equilibrium configurations and switching fields of small ferromagnetic particles.

Recently, steepest descent methods have been revisited for large scale minimization problems.[6] In combination with a special choice of a step length,[5] steepest descent methods might even out-perform preconditioned conjugate gradient methods.

In this work, we use a variant of these newly developed steepest descent methods and apply it to finite element micromagnetics. In order to take into account the constraint on the magnitude of the magnetization, we use a curvilinear search[7] approach on the sphere. We compare the performance of the minimization algorithm with fast LLG solvers for computation of the hysteresis loop of permanent magnets.

---

[a]Electronic mail: lukas.exl@univie.ac.at.

115, 17D118-1

Further, we report on the graphic processor unit (GPU) implementation of the algorithm.

II. METHOD

We are interested in finding local minima of the total Gibbs free energy, i.e., we want to solve

$$\min \phi_t(\boldsymbol{m}) \quad \text{subject to} \quad \|\boldsymbol{m}\| = 1, \tag{1}$$

where the total energy ϕ_t consists of stray field, exchange, anisotropy, and external energy.

A. Steepest descent direction

Let $g : \boldsymbol{m} \mapsto \boldsymbol{m}/\|\boldsymbol{m}\|$ denote the map onto the unit sphere. We get for the variational derivative of $h := \phi_t \circ g$ for unit magnetization, i.e., $\|\boldsymbol{m}\| = 1$,

$$
\begin{aligned}
\frac{\delta}{\delta \boldsymbol{m}} h(\boldsymbol{m}) &= J_g^T(\boldsymbol{m}) \frac{\delta}{\delta \boldsymbol{m}} \phi_t(g(\boldsymbol{m})) \\
&= \frac{\delta}{\delta \boldsymbol{m}} \phi_t(\boldsymbol{m}) - \left(\boldsymbol{m} \cdot \frac{\delta}{\delta \boldsymbol{m}} \phi_t(\boldsymbol{m}) \right) \boldsymbol{m},
\end{aligned} \tag{2}
$$

where $J_g(\boldsymbol{m})$ is the Jacobian of g at $\boldsymbol{m}$.

The steepest ascent direction $\frac{\delta}{\delta \boldsymbol{m}} h(\boldsymbol{m})$ corresponds to the orthogonal projection of $\frac{\delta}{\delta \boldsymbol{m}} \phi_t(\boldsymbol{m})$ onto the orthogonal complement of $\boldsymbol{m}$. Using *Lagrange's formula*, i.e.,

$$\boldsymbol{a} \times (\boldsymbol{b} \times \boldsymbol{c}) = (\boldsymbol{a} \cdot \boldsymbol{c})\boldsymbol{b} - (\boldsymbol{a} \cdot \boldsymbol{b})\boldsymbol{c}, \tag{3}$$

we can rewrite Eq. (2)

$$\frac{\delta}{\delta \boldsymbol{m}} h(\boldsymbol{m}) = \boldsymbol{m} \times \left(-\boldsymbol{m} \times \frac{\delta}{\delta \boldsymbol{m}} \phi_t(\boldsymbol{m}) \right). \tag{4}$$

Assume now that ϕ_t is a spacial approximation of the total Gibbs free energy, e.g., by finite element discretization. By the definition

$$H(\boldsymbol{m}) := -\boldsymbol{m} \times \nabla \phi_t(\boldsymbol{m}), \tag{5}$$

a steepest descent method would calculate a new iteration $\boldsymbol{m}^{n+1}$ from a given normalized approximation $\boldsymbol{m}^n$ by

$$\boldsymbol{m}^{n+1} = \boldsymbol{m}^n - \tau_n \boldsymbol{m}^n \times H(\boldsymbol{m}^n), \tag{6}$$

for a certain step size τ_n. Note that the new approximation $\boldsymbol{m}^{n+1}$ is not normalized in general.

B. Curvilinear search on the sphere

In our iteration method, we replace the steepest descent direction by

$$-\frac{\boldsymbol{m}^n + \boldsymbol{m}^{n+1}}{2} \times H(\boldsymbol{m}^n), \tag{7}$$

yielding the iteration scheme

$$\boldsymbol{m}^{n+1} = \boldsymbol{m}^n - \tau \frac{\boldsymbol{m}^n + \boldsymbol{m}^{n+1}}{2} \times H(\boldsymbol{m}^n). \tag{8}$$

This update scheme preserves the length of the iterates, i.e., $\|\boldsymbol{m}^{n+1}\| = \|\boldsymbol{m}^n\|$, which can be easily checked by multiplying Eq. (8) by $\boldsymbol{m}^n + \boldsymbol{m}^{n+1}$. Moreover, the new state $\boldsymbol{m}^{n+1}$ can be computed by explicit formulas.[7]

We want to stress that the update scheme (8) can be seen as an implicit integration rule for the flow equation

$$\partial_t \boldsymbol{m} = -\boldsymbol{m} \times \left(-\boldsymbol{m} \times \frac{\delta}{\delta \boldsymbol{m}} \phi_t(\boldsymbol{m}) \right), \tag{9}$$

which is, up to constants, the Landau-Lifshitz equation with only damping, since the effective field is defined as $\boldsymbol{h}_{\text{eff}} = -\delta \phi_t / \delta \boldsymbol{m}$.

C. Step length selection

The first step size τ_0 is calculated by an inexact line search and all subsequent steps τ_n by the so-called *Barzilian-Borwein* (BB) rule.[2] We therefore define $\boldsymbol{g}^n := \nabla h(\boldsymbol{m}^n) = \boldsymbol{m}^n \times (-\boldsymbol{m}^n \times \nabla \phi_t(\boldsymbol{m}^n))$, $\boldsymbol{s}^{n-1} := \boldsymbol{m}^n - \boldsymbol{m}^{n-1}$ and $\boldsymbol{y}^{n-1} := \boldsymbol{g}^n - \boldsymbol{g}^{n-1}$. The step size τ_n is determined such that $D^n := \tau_n^{-1} I$ is an approximation of the Hessian of h at $\boldsymbol{m}^n$, i.e., the *secant equation* $D^n \boldsymbol{s}^{n-1} = \boldsymbol{y}^{n-1}$ holds. The two possible solutions to this equation are

$$\tau_n^1 = \frac{(\boldsymbol{s}^{n-1})^T \boldsymbol{s}^{n-1}}{(\boldsymbol{s}^{n-1})^T \boldsymbol{y}^{n-1}}, \quad \tau_n^2 = \frac{(\boldsymbol{s}^{n-1})^T \boldsymbol{y}^{n-1}}{(\boldsymbol{y}^{n-1})^T \boldsymbol{y}^{n-1}}. \tag{10}$$

One possibility is to alternately switch between τ_n^1 and τ_n^2. However, in our numerical tests we used the more elaborate strategy proposed for step selection in gradient projection methods.[12]

Note that the BB rule yields a non-monotonic method; as globalization strategy, we therefore used inexact line search if the new computed energy was still larger than the previous 20 energies.

III. RESULTS

A. Comparison with LLG solver

Here, we compare the steepest descent solver with a state-of-the-art finite element solver. We computed the demagnetization curve of a spherical $Nd_2Fe_{14}B$ particle with a diameter of 20 nm. The number of tetrahedral elements is 80 000. The edge length of a tetrahedron is 0.76 nm, which is smaller than the exchange length $\sqrt{A/K_1} = 1.35$ nm for $Nd_2Fe_{14}B$.[14] Owing to the small particle size and the fine mesh, the problem is stiff. Stiffness in micromagnetics has been identified by the collaborative motion of many magnetic moments[18] at once. Generally, stiffness leads to slow convergence when explicit solvers are used.

The steepest descent solver and the LLG solver compute the magnetostatic field via a magnetic scalar potential. The LLG solver uses a hybrid finite element/boundary element method for the computation of the magnetostatic field. The boundary element method is accelerated using hierarchical matrices. For ease of porting the software to the GPU, we use a space transformation method[3] instead of the accelerated boundary element method for the treatment of the open boundary problem in the steepest descent solver. In the LLG

TABLE I. Comparison of the steepest descent solver and the LLG solver: Relative error in the coercive field, number of function evaluations, time to solution on a netbook with an AMD E2-1800 processor (1.7 GHz). The CPU time does not include the setup time (matrix assembly and Lower Upper (LU) decomposition.)

Method	Error	Evaluations	CPU time (s)
Steepest descent	6×10^{-4}	13 250	3801
LLG solver	2×10^{-3}	6379	8114

TABLE II. Main tasks in the steepest descent solver.

Computation of	Linear algebra operation
Energy gradient	Sparse matrix vector multiplication
Search direction	Vector add and multiply
Step length	Dot product
Scalar potential	Solve linear system

solver, we set the torque term to zero. Time integration is performed using a preconditioned, implicit method with adaptive time step selection.[17]

The external field is ramped from 0 to -6 T in steps of 0.01 T. In the static solver, the external field is decreased when the l^2 norm of the projected gradient is smaller than 10^{-6}, i.e., $\nabla h^T \nabla h < 10^{-12}$, cf. Eq. (2). In the dynamic solver, the field is changed at a rate of -0.01 T/ns, which is slow enough to calculate a good approximation to the analytical value of the coercive field. The damping constant in the dynamic approach is not relevant because an adaptive time step scheme is used. The field is applied at $45°$ with respect to the uniaxial anisotropy axis.

Table I compares the time to reach the solution and the number of function evaluation for two algorithms. The function evaluations denote the energy gradient evaluations for the steepest descent method and the effective field evaluation in the LLG solver. The computed coercive field is close to the analytic value (half the anisotropy field) for both simulation methods. Although the LLG solver requires less function evaluations, the steepest descent method is faster by a factor of 2.13. In the steepest descent method, more then 90% of the time is spent for solving the linear system for the magnetic scalar potential.

B. GPU implementation

The steepest descent method was also implemented on a GPU. The implementation of the steepest descent method is based on sparse matrix operations and basic linear algebra. The required linear algebra operations are listed in Table II. We use OpenBLAS for the single core CPU implementation and the PARALUTION library[13] (version 0.3.0) for GPU implementation.

We compared the time to reach the solution for computed the demagnetization curve. The CPU was an Intel i7. The GPU was a NVIDIA m2070. We obtained a speed-up of

4.8 on the GPU as compared to the fastest single core CPU implementation.

IV. CONCLUSION

The hysteresis loop of permanent magnets can be computed using energy minimization techniques. Steepest descent methods with modified Barzilai-Borwein step length selection outperform state-of-the-art LLG solvers. With fast sparse matrix libraries, the proposed solver shows reasonable performance on graphic cards.

[1]F. Alouges, S. Conti, A. Desimone, and Y. Pokern, "Energetics and switching of quasi-uniform states in small ferromagnetic particles," ESAIM: Math. Modell. Numer. Anal. **38**(2), 235–248 (2003).

[2]J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," IMA J. Numer. Anal. **8**(1), 141–148 (1988).

[3]X. Brunotte, S. Martin, G. Meunier, and J.-F. Imhoff, "Finite element modeling of unbounded problems using transformations: A rigorous, powerful and easy solution," IEEE Trans. Magn. **28**(2), 1663–1666 (1992).

[4]R. Cohen, S. Y. Lin, and M. Luskin, "Relaxation and gradient methods for molecular orientation in liquid crystals," Comput. Phys. Commun. **53**, 455–465 (1989).

[5]R. Fletcher, "On the barzilai-borwein method," in *Optimization and Control with Applications*, edited by L. Qi, K. L. Teo, and X. Q. Yang (Springer, 2005), pp. 235–256.

[6]R. Fletcher, "A limited memory steepest descent method," Math. Prog., Ser. A **135**, 413–436 (2012).

[7]D. Goldfarb, Z. Wen, and W. Yin, "A curvilinear search method for p-harmonic flows on spheres," SIAM J. Imaging Sci. **2**(1), 84–109 (2009).

[8]O. Gutfleisch, M. A. Willard, E. Brück, C. H. Chen, S. G. Sankar, and J. P. Liu, "Magnetic materials and devices for the 21st century: Stronger, lighter, and more energy efficient," Adv. Mater. **23**(7), 821–842 (2011).

[9]D. S. Kinderlehrer and L. Ma, "Simulation of hysteresis in nonlinear systems," in *North American Conference on Smart Structures and Materials* edited by H. T. Banks (International Society for Optics and Photonics, 2004), pp. 78–87.

[10]K. Kosavisutte and N. Hayaslii, "A numerical study of labonte's iteration: An approach to acceleration," IEEE Trans. Magn. **32**(5), 4243–4245 (1996).

[11]A. E. LaBonte, "Two-dimensional Bloch-type domain walls in ferromagnetic films," J. Appl. Phys. **40**(6), 2450 (1969).

[12]I. Loris, M. Bertero, C. De Mol, R. Zanella, and L. Zanni, "Accelerating gradient projection methods for l1-constrained signal recovery by steplength selection rules," Appl. Comput. Harmonic Anal. **27**(2), 247–254 (2009).

[13]D. Lukarski, *Paralution Project*, see http://www.paralution.com/.

[14]W. Rave, K. Ramstöck, and A. Hubert, "Corners and nucleation in micromagnetics," J. Magn. Magn. Mater. **183**(3), 329–333 (1998).

[15]H. Sepehri-Amin, T. Ohkubo, S. Nagashima, M. Yano, T. Shoji, A. Kato, T. Schrefl, and K. Hono, "High-coercivity ultrafine-grained anisotropic Nd-Fe-B magnets processed by hot deformation and the Nd-Cu grain boundary diffusion process," Acta Mater. **61**(17), 6622–6634 (2013).

[16]R. Skomski, P. Manchanda, P. K. Kumar, B. Balamurugan, A. Kashyap, and D. J. Sellmyer, "Predicting the future of permanent-magnet materials," IEEE Trans. Magn. **49**(7), 3215–3220 (2013).

[17]D. Suess, V. Tsiantos, and T. Schrefl, "Time resolved micromagnetics using a preconditioned time integration method," J. Magn. Magn. Mater. **248**, 298–311 (2002).

[18]E. D. Torre, "Stiff nodes in numerical micromagnetic problems," IEEE Trans. Magn. **29**(6), 2371–2373 (1993).

[19]A. Viallix, F. Boileau, and R. Klein, "A new method for finite element calculation of micromagnetic problems," IEEE Trans. Magn. **24**, 2371–2373 (1988).