A fast and accurate numerical method for the computation of unstable micromagnetic configurations

Sören Bartels, Mario Bebendorf, and Michael Bratsch

Abstract We present a fast and accurate numerical method to compute unstable micromagnetic configurations. The proposed scheme, which combines various state of the art methods, is able to treat the pointwise unit-length constraint of the magnetization field and to efficiently compute the stray field energy. Furthermore, numerical results are presented which are in agreement with the expected results in simple situations and allow predictions beyond theory.

Acknowledgements The authors wish to thank Felix Otto and Matthias Kurzke for stimulating discussions on the topic and Dirk Praetorius for pointing out the reference [14] on the employed splitting method for the computation of the stray field.

1 Introduction

The computation of minimal switching energies between two given stable states and the detection of a corresponding unstable critical configuration is an important task in the mathematical modeling of many physical phenomena [2, 12, 23]. In this paper we address this problem and thereby aim at contributing to the understanding of the energy landscape for a mathematically challenging and well established model energy functional in micromagnetics; cf. [9, 25]. Its particular features are that it involves a unit-length constraint for the magnetization field and requires the computation of a stray field.

The finite element treatment of minimization problems and partial differential equations with pointwise constraints has been investigated intensively in recent years [1, 2]. Reliable and efficient

Sören Bartels
Universität Freiburg, Hermann-Herder-Str. 10, 79104 Freiburg, e-mail: bartels@mathematik.uni-freiburg.de

Mario Bebendorf
Universität Bonn, Wegelerstr. 6, 53115 Bonn, e-mail: bebendorf@ins.uni-bonn.de

Michael Bratsch
Universität Bonn, Wegelerstr. 6, 53115 Bonn, e-mail: bratsch@ins.uni-bonn.de

methods are now available that typically impose the constraint at the nodes of a triangulation [4, 5]. In iterative schemes the constraint is linearized and afterwards updates are made which consist of corrections of tangent spaces and a subsequent nodewise nearest-neighbor projection onto the given target manifold which lead to linear systems of equations.

Computing the solution of an exterior domain problem is often formulated as a boundary equation with a non-local operator and then requires the solution of linear systems of equations with large, fully populated matrices. This is can be done efficiently with the technique of so-called $\mathscr{H}$-matrices which can approximate the arising matrices with almost linear complexity, cf. [19, 7, 27].

Various methods are available to compute unstable critical points, i.e., saddle points, of energy functionals known as mountain pass algorithms [8]. They typically assume that the functional under consideration is defined on a linear space and therefore cannot be employed if this is not the case, i.e., if the linear interpolation between admissible configurations does not belong to the domain of the functional. A method that is capable to cope with related difficulties is the recently developed string method, see [11, 13], that evolves an entire path that connects two given states in the set of admissible configurations.

For the Landau-Lifschitz energy in micromagnetics certain stable critical configurations such as the so-called flower and vortex state are known [29, 22]. To efficiently switch between such states, e.g., by an applied field, it is important to determine the minimal energy required to achieve this change. We use the string method in combination with finite element methods to deal with the pointwise unit-length constraint and the $\mathscr{H}$-matrix technology to efficiently compute this energy and to identify a corresponding magnetization. The resulting numerical method is verified for a standard problem [26] and a so-called minimum energy path connecting the flower and vortex state is presented.

2 The Landau-Lifschitz model

Let $\Omega \subset \mathbb{R}^3$ be a domain and let the magnetization $\mathbf{m} : \mathbb{R}^3 \to \mathbb{R}^3$ satisfying $\|\mathbf{m}(x)\|_2 = 1$ in Ω and $\mathbf{m}(x) = 0$ for all $x \notin \Omega$ be given. The energy associated with $\mathbf{m}$ is

$$E(\mathbf{m}) := \frac{1}{2}\int_\Omega \|D\mathbf{m}\|_F^2 \,\mathrm{d}x + \int_\Omega \phi(\mathbf{m}) - \mathbf{f}\cdot\mathbf{m}\,\mathrm{d}x + E_s(\mathbf{m}), \tag{1}$$

where $\phi(\mathbf{m}) := 1 - (\mathbf{e}\cdot\mathbf{m})^2$ with given $\mathbf{e},\mathbf{f} \in \mathbb{R}^3$ satisfying $\|\mathbf{e}\|_2 = 1$ and

$$E_s(\mathbf{m}) := \frac{\mu_0}{2}\int_{\mathbb{R}^3} \|H\|_2^2 \,\mathrm{d}x, \quad \mu_0 := 4\pi\cdot 10^{-7},$$

denotes the energy of the stray field H corresponding to $\mathbf{m}$. H can be computed from the Maxwell equations in the absence of electrical currents and charges

$$\operatorname{div} B = 0, \tag{2a}$$

$$\operatorname{curl} H = 0. \tag{2b}$$

H and the magnetic induction B are coupled by the equation $B = \mu_0(H + \mathbf{m})$.

From equation (2b) it follows that there is the so-called magnetostatic potential $u_\mathbf{m} : \mathbb{R}^3 \to \mathbb{R}$ satisfying $H = -\nabla u_\mathbf{m}$. Then (2a) becomes

$$\Delta u_\mathbf{m} = \operatorname{div}\mathbf{m},$$

which is equivalent with the weak formulation

$$\int_{\mathbb{R}^3} \nabla u_{\mathbf{m}} \cdot \nabla w \, \mathrm{d}x = \int_{\Omega} \mathbf{m} \cdot \nabla w \, \mathrm{d}x \quad \text{for all } w \in H^1(\mathbb{R}^3). \tag{3}$$

Notice that this defines a linear mapping $\mathbf{m} \mapsto u_{\mathbf{m}}$ with

$$\|\nabla u_{\mathbf{m}}\|_{L^2(\mathbb{R}^3)} \leq \|\mathbf{m}\|_{L^2(\Omega)}. \tag{4}$$

Using $w = u_{\mathbf{m}}$ in (3), it follows that

$$E_s(\mathbf{m}) = \frac{\mu_0}{2} \int_{\Omega} \mathbf{m} \cdot \nabla u_{\mathbf{m}} \, \mathrm{d}x. \tag{5}$$

In addition to the energy $E(\mathbf{m})$ also its derivative $E'(\mathbf{m})[\mathbf{v}]$ will be important. Let the direction $\mathbf{v}$ be given such that $\mathbf{v}(x) \cdot \mathbf{m}(x) = 0$ for almost every $x \in \Omega$. Then we obtain that

$$E'(\mathbf{m})[\mathbf{v}] = \int_{\Omega} \operatorname{trace}(D\mathbf{m})^T (D\mathbf{v}) \, \mathrm{d}x - 2 \int_{\Omega} (\mathbf{e} \cdot \mathbf{m})(\mathbf{e} \cdot \mathbf{v}) \, \mathrm{d}x - \int_{\Omega} f \cdot \mathbf{v} \, \mathrm{d}x + E_s'(\mathbf{m})[\mathbf{v}],$$

where

$$E_s'(\mathbf{m})[\mathbf{v}] = \mu_0 \int_{\mathbb{R}^3} H(\mathbf{v}) H(\mathbf{m}) \, \mathrm{d}x = \mu_0 \int_{\Omega} \mathbf{v} \cdot \nabla u_{\mathbf{m}} \, \mathrm{d}x$$

due to (3).

3 Efficient computation of the stray field energy

The stray field energy represents the non-local effects of the magnetization. Hence, it is the numerically most challenging part of the computation of the Landau-Lifschitz model; see (1). In the following, a reformulation will be shown so that $\mathscr{H}$-matrices can be applied to approximate the local and non-local parts of the stray field energy.

3.1 Different formulations

Our aim is to find an explicit expression for the magnetostatic potential $u_{\mathbf{m}}$. Equation (3) is equivalent to the following boundary value problem

$$\Delta u_{\mathbf{m}} = \begin{cases} \operatorname{div} \mathbf{m}, & \text{in } \Omega, \\ 0, & \text{in } \Omega^c := \mathbb{R}^3 \setminus \overline{\Omega}, \end{cases} \tag{6a}$$

$$[u_{\mathbf{m}}] = 0 \text{ on } \partial\Omega, \tag{6b}$$

$$[\partial_\nu u_{\mathbf{m}}] = -\mathbf{m} \cdot \nu \text{ on } \partial\Omega, \tag{6c}$$

which has the solution

$$u_{\mathbf{m}}(x) = \int_{\Omega} \nabla S(x-y) \cdot \mathbf{m}(y) \, \mathrm{d}y,$$

where $S(x) := -\frac{1}{4\pi} \|x\|_2^{-1}$ denotes the singularity function of the Laplacian. Note that $[\cdot]$ in (6) denotes the jump across the boundary $\partial\Omega$.

The computation of the stray field energy using the latter representation of $u_{\mathbf{m}}$ in combination with hierarchical matrices was already done in [27]. We favor the following representation (see [14]), because it leads to the interaction of Ω with its boundary $\partial\Omega$. Let u_1 and u_2 satisfy the following boundary value problems

$$\Delta u_1 = \operatorname{div} \mathbf{m} \text{ in } \Omega,$$
$$u_1 = 0 \text{ on } \partial\Omega$$

and

$$\Delta u_2 = 0 \text{ in } \Omega \cup \Omega^c, \tag{7a}$$
$$[u_2] = 0 \text{ on } \partial\Omega, \tag{7b}$$
$$[\partial_\nu u_2] = g \text{ on } \partial\Omega, \tag{7c}$$

where $g := (\nabla u_1 - \mathbf{m}) \cdot \nu$. Then $u_{\mathbf{m}} = u_1 + u_2$ and the solution of the homogeneous problem (7) is

$$u_2(x) = \int_{\partial\Omega} S(x-y) g(y) \, \mathrm{d}s_y.$$

Hence, from (5)

$$E_s(\mathbf{m}) = \frac{\mu_0}{2} \int_\Omega \mathbf{m} \cdot \nabla u_1 \, \mathrm{d}x + \frac{\mu_0}{2} \int_\Omega \int_{\partial\Omega} \mathbf{m}(x) \cdot \nabla S(x-y) g(y) \, \mathrm{d}s_y \, \mathrm{d}x.$$

From

$$\int_\Omega \mathbf{m}(x) \cdot \nabla S(x-y) \, \mathrm{d}x = -\int_\Omega S(x-y)(\operatorname{div} \mathbf{m})(x) \, \mathrm{d}x + \int_{\partial\Omega} S(x-y) \mathbf{m}(x) \cdot \nu_x \, \mathrm{d}s_x$$

we obtain that

$$\begin{aligned} E_s(\mathbf{m}) = {} & \frac{\mu_0}{2} \int_\Omega \mathbf{m} \cdot \nabla u_1 \, \mathrm{d}x - \frac{\mu_0}{2} \int_\Omega \int_{\partial\Omega} S(x-y)(\operatorname{div} \mathbf{m}) g(y) \, \mathrm{d}s_y \, \mathrm{d}x \\ & + \frac{\mu_0}{2} \int_{\partial\Omega} \int_{\partial\Omega} S(x-y) \mathbf{m}(x) \cdot \nu_x \, g(y) \, \mathrm{d}s_y \, \mathrm{d}s_x. \end{aligned}$$

3.2 Discretization

We assume that the computational domain Ω is decomposed into a set of tetrahedra $\mathscr{T}_h$ such that $\Omega = \cup_{\tau \in \mathscr{T}_h} \tau$. The finite element space consisting of linear ansatz functions $\Phi = (\varphi_i)_{i \in I}$ is denoted by $\mathscr{S}^1(\mathscr{T}_h)$, the corresponding set of nodes will be referred to as $\mathscr{N}_h$. We discretize the magnetization $\mathbf{m}_h \in \mathscr{S}^1(\mathscr{T}_h)^3$ such that

$$\mathbf{m}_h = \sum_{i \in I} \alpha_i \varphi_i, \quad \alpha_i \in \mathbb{R}^3.$$

Then $D\mathbf{m}_h = \sum_{i \in I} \alpha_i (\nabla \varphi_i)^T$, and the first term in (1) can be computed from

$$\begin{aligned} \int_\Omega \|D\mathbf{m}_h\|_F^2 \, \mathrm{d}x &= \int_\Omega \operatorname{trace} (D\mathbf{m}_h)^T (D\mathbf{m}_h) \, \mathrm{d}x = \sum_{i,j \in I} \int_\Omega \operatorname{trace} \nabla\varphi_i \alpha_i^T \alpha_j (\nabla\varphi_j)^T \, \mathrm{d}x \\ &= \sum_{i,j \in I} \alpha_i \cdot \alpha_j \int_\Omega \operatorname{trace} \nabla\varphi_i (\nabla\varphi_j)^T \, \mathrm{d}x = \sum_{i,j \in I} \alpha_i \cdot \alpha_j \int_\Omega \nabla\varphi_i \cdot \nabla\varphi_j \, \mathrm{d}x. \end{aligned}$$

The second and the third term in (1) have the values

$$\int_\Omega \phi(\mathbf{m}_h) \, \mathrm{d}x = \int_\Omega 1 - (\mathbf{e} \cdot \mathbf{m}_h)^2 \, \mathrm{d}x = \operatorname{vol}(\Omega) - \sum_{i,j \in I} (\mathbf{e} \cdot \alpha_i)(\mathbf{e} \cdot \alpha_j) \int_\Omega \varphi_i \varphi_j \, \mathrm{d}x$$

and

$$\int_\Omega \mathbf{f}\cdot\mathbf{m}_h\,\mathrm{d}x = \sum_{i\in I}\mathbf{f}\cdot\alpha_i\int_\Omega \varphi_i\,\mathrm{d}x.$$

Since u_1 vanishes on $\partial\Omega$, for the discretization of u_1 only the inner degrees of freedom are used, i.e.,

$$u_1^h = \sum_{j\in I_{in}}\beta_j\varphi_j$$

with $I_{in} := I\setminus I_{bd}$, where I_{bd} are the boundary indices. Let $I_{lay}\subset I_{in}$ be the vertices which have a neighbor in the set of boundary indices I_{bd}. Then the restriction of u_1^h to the boundary $\partial\Omega$ reads $(\nabla u_1^h)|_{\partial\Omega} = \sum_{i\in I_{lay}}\beta_i\nabla\varphi_i$ and

$$g_h = \sum_{j\in I_{lay}}\beta_j\nu\cdot\nabla\varphi_j - \sum_{j\in I_{bd}}\nu\cdot\alpha_j\varphi_j.$$

Hence,

$$\begin{aligned}
E_s(\mathbf{m}_h) &= \frac{\mu_0}{2}\sum_{i\in I}\left(\sum_{j\in I_{in}}\beta_j\alpha_i\cdot\int_\Omega\varphi_i\nabla\varphi_j\,\mathrm{d}x - \alpha_i\cdot\int_\Omega\int_{\partial\Omega}\nabla\varphi_i(x)S(x-y)g(y)\,\mathrm{d}s_y\,\mathrm{d}x\right)\\
&\quad+\frac{\mu_0}{2}\sum_{i\in I_{bd}}\alpha_i\cdot\int_{\partial\Omega}\int_{\partial\Omega}\nu_x\varphi_i(x)S(x-y)g(y)\,\mathrm{d}s_y\,\mathrm{d}x\\
&= \frac{\mu_0}{2}\sum_{i\in I}\left(\sum_{j\in I_{in}}\beta_j\alpha_i\cdot\int_\Omega\varphi_i\nabla\varphi_j\,\mathrm{d}x - \sum_{j\in I_{lay}}\beta_j\alpha_i\cdot\mathbf{a}_{ij} + \sum_{j\in I_{bd}}\alpha_i\cdot(B_{ij}\alpha_j)\right)\\
&\quad+\frac{\mu_0}{2}\sum_{i\in I_{bd}}\left(\sum_{j\in I_{lay}}\beta_j\alpha_i\cdot\mathbf{c}_{ij} - \sum_{j\in I_{bd}}\alpha_i\cdot(D_{ij}\alpha_j)\right),
\end{aligned}$$

where

$$\mathbf{a}_{ij} = \int_\Omega\int_{\partial\Omega}\nabla\varphi_i(x)S(x-y)\nu\cdot\nabla\varphi_j(y)\,\mathrm{d}s_y\,\mathrm{d}x, \tag{8}$$

$$B_{ij} = \int_\Omega\int_{\partial\Omega}\nabla\varphi_i(x)S(x-y)\varphi_j(y)\nu_y^T\,\mathrm{d}s_y\,\mathrm{d}x, \tag{9}$$

and

$$\mathbf{c}_{ij} = \int_{\partial\Omega}\int_{\partial\Omega}\nu_x\varphi_i(x)S(x-y)\nu_y\cdot\nabla\varphi_j(y)\,\mathrm{d}s_y\,\mathrm{d}s_x, \tag{10}$$

$$D_{ij} = \int_{\partial\Omega}\int_{\partial\Omega}\nu_x\varphi_i(x)S(x-y)\varphi_j(y)\nu_y^T\,\mathrm{d}s_y\,\mathrm{d}s_x. \tag{11}$$

The efficient numerical evaluation of the singular integrals (8), (9), (10), and (11) using the Duffy transformation is described in the appendix.

4 Hierarchical matrices

For the computation of the different energies in (1) it is necessary to efficiently treat fully populated matrices arising from the discretization of non-local operators, e.g. finite or boundary element discretizations of integral operators and inverses or the factors of the LU decomposition of FE discretizations of elliptic partial differential operators. For this purpose, Tyrtyshnikov [30] and Hackbusch et al. [18, 20, 19] introduced the structure of *mosaic skeleton matrices* or *hierarchical*

($\mathscr{H}$-) matrices. A similar approach, which is designed towards only the fast multiplication of a matrix by a vector, are the earlier developed fast summation methods *tree code* [3], *fast multipole methods* [16, 17], and *panel clustering* [21].

Many existing fast methods are based on multi-level structures. In contrast to multigrid methods, the efficiency of $\mathscr{H}$-matrices is based on a suitable hierarchy of partitions of the matrix indices. Let $I = \{1,\dots,M\}$ and $J = \{1,\dots,N\}$ be sets of indices corresponding to the rows and columns of a matrix $A \in \mathbb{R}^{M\times N}$. The efficiency of hierarchical matrices is based on the low-rank representation of sub-blocks from an appropriate partition $\mathscr{P}$ of the set of matrix indices $I \times J$; see Fig. 1. The construction of $\mathscr{P}$ is usually done in the following way. First, cluster trees T_I and T_J

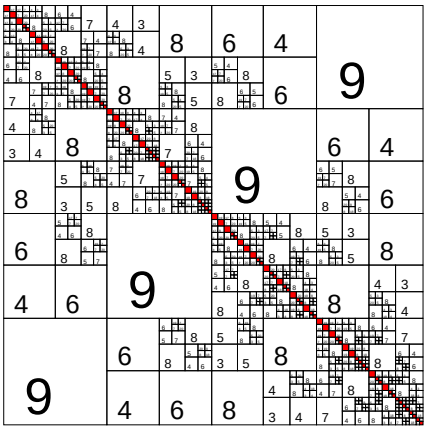


Fig. 1 A hierarchical matrix with its rank distribution.

are constructed by recursive subdivision of I and J. Each subdivision step is done such that indices that in some sense are close to each other are grouped into two clusters. In a second step the block cluster tree $T_{I\times J}$ is built by recursive subdivision of $I \times J$. Each block $t \times s$ is subdivided into the sons $t' \times s'$, where t' and s' are taken from the list of sons of t and s in T_I and T_J, respectively. The recursion stops at blocks $t \times s$ which either are small enough or satisfy a so-called *admissibility condition*. This condition guarantees that the restriction A_{ts} of A to $t \times s$ can be approximated by a matrix of low rank. It usually takes into account the geometry that is associated with the rows t and the columns s. The partition $\mathscr{P}$ is found as the leaves of $T_{I\times J}$. The set of hierarchical matrices on the partition $\mathscr{P}$ and blockwise rank k is then defined as

$$\mathscr{H}(\mathscr{P},k) = \{A \in \mathbb{R}^{I\times J} : \operatorname{rank} A_b \le k \text{ for all } b \in \mathscr{P}\}.$$

The elements of this set can be stored with logarithmic-linear complexity and hence provide data-sparse representations of fully populated matrices. Additionally, exploiting the hierarchical structure of the partition, an approximate algebra can be defined [15] which is based on divide-and-conquer versions of the usual arithmetic operations. At least for discretizations of elliptic operators, these approximate operations have logarithmic-linear complexity (see [7]) and can be used to define substitutes for higher level matrix operations such as inversion, LU factorization, and QR factorization.

Using hierarchical matrices, we are able to efficiently compute the local and non-local parts of the energy functional (1) with almost optimal complexity. Especially approximations to the fully

populated matrices (8), (9), (10), and (11) can be constructed with logarithmic-linear complexity via adaptive cross approximation (ACA) [6].

5 Iterative minimization of the energy

The numerical computation of local energy minima of the functional (1) is a challenging task due to the pointwise restriction $\|\mathbf{m}(x)\|_2 = 1$ for almost every $x \in \Omega$. In this section we will see how this constraint can be treated efficiently.

5.1 The minimization algorithm

We describe and analyze in this section our method to iteratively minimize the energy functional (1). To simplify the presentation, we consider the Dirichlet energy and a lower order term, i.e., the model functional

$$E(\mathbf{m}) = \Theta(\mathbf{m}) + \frac{1}{2}\int_\Omega \|D\mathbf{m}\|_F^2 \,\mathrm{d}x \tag{12}$$

with a smooth functional $\Theta : H^1(\Omega;\mathbb{R}^3) \to \mathbb{R}$. The following algorithm realizes a discrete H^1 flow for E with time-step size $\alpha > 0$ and employs ideas from [1, 2, 4].

Algorithm (minimization algorithm): Given $\mathbf{m}_h^0$ such that $\|\mathbf{m}_h^0(z)\|_2 = 1$ for all $z \in \mathcal{N}_h$ iterate for $n = 0, 1, 2, \dots$ the following steps:

(1) Compute $\mathbf{w}_h^n \in \mathcal{S}^1(\mathcal{T}_h)^3$ with $\int_\Omega \mathbf{w}_h^n \,\mathrm{d}x = 0$ such that $\mathbf{w}_h^n(z) \cdot \mathbf{m}_h^n(z) = 0$ for all $z \in \mathcal{N}_h$ and

$$\int_\Omega \operatorname{trace}\left[(D\mathbf{w}_h^n)^T (D\mathbf{v}_h)\right] \mathrm{d}x = -\Theta'(\mathbf{m}_h^n)[\mathbf{v}_h] - \int_\Omega \operatorname{trace}\left[(D\mathbf{m}_h^n)^T (D\mathbf{v}_h)\right] \mathrm{d}x$$

for all $\mathbf{v}_h \in \mathcal{S}^1(\mathcal{T}_h)^3$ with $\int_\Omega \mathbf{v}_h \,\mathrm{d}x = 0$.

(2) Define $\mathbf{m}_h \in \mathcal{S}^1(\mathcal{T}_h)^3$ through

$$\mathbf{m}_h^{n+1}(z) = \frac{\mathbf{m}_h^n(z) + \alpha \mathbf{w}_h^n(z)}{\|\mathbf{m}_h^n(z) + \alpha \mathbf{w}_h^n(z)\|_2}$$

for all $z \in \mathcal{N}_h$ with a suitable $\alpha > 0$.

To ensure that the iteration is energy decreasing, we assume that the underlying triangulation $\mathcal{T}_h$ is weakly acute, i.e., that the off-diagonal entries of the corresponding $P1$ stiffness matrix are non-positive, cf. [4] for details.

Lemma 1 ([4]). *Assume that $\mathcal{T}_h$ is weakly acute and suppose that $\mathbf{m}_h, \mathbf{w}_h \in \mathcal{S}^1(\mathcal{T}_h)^3$ are such that $\|\mathbf{m}_h(z)\|_2 = 1$ and $\mathbf{m}_h(z) \cdot \mathbf{w}_h(z) = 0$ for all $z \in \mathcal{N}_h$. Then*

$$\left\| D\left(\frac{\mathbf{m}_h + \alpha \mathbf{w}_h}{\|\mathbf{m}_h + \alpha \mathbf{w}_h\|_2} \right) \right\|_{L^2(\Omega)} \leq \|D(\mathbf{m}_h + \alpha \mathbf{w}_h)\|_{L^2(\Omega)}$$

for every $\alpha \in \mathbb{R}$.

To show convergence of the iterative algorithm, we argue as in [1, 2, 4]. For $\mathbf{m}_h^n$ and $\mathbf{w}_h^n$ as in the algorithm we have, upon choosing $\mathbf{v}_h = \mathbf{w}_h^n$ that

$$\int_\Omega \operatorname{trace}[(D\mathbf{m}_h^n)^T(D\mathbf{w}_h^n)]\,\mathrm{d}x = -\Theta'(\mathbf{m}_h^n)[\mathbf{w}_h^n] - \int_\Omega \|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x.$$

Lemma 1 implies that

$$\begin{aligned}\frac{1}{2}\int_\Omega \|D\mathbf{m}_h^{n+1}\|_F^2 - \|D\mathbf{m}_h^n\|_F^2\,\mathrm{d}x &\le \frac{1}{2}\int_\Omega \|D(\mathbf{m}_h^n+\alpha\mathbf{w}_h^n)\|_F^2 - \|D\mathbf{m}_h^n\|_F^2\,\mathrm{d}x\\ &= \alpha\int_\Omega \operatorname{trace}[(D\mathbf{m}_h^n)^T(D\mathbf{w}_h^n)] + \frac{\alpha^2}{2}\|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x\\ &= -\alpha\Theta'(\mathbf{m}_h^n)[\mathbf{w}_h^n] + (\alpha^2/2-\alpha)\int_\Omega \|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x.\end{aligned}$$

Hence, it follows that

$$\begin{aligned}E(\mathbf{m}_h^{n+1}) - E(\mathbf{m}_h^n) &= \Theta(\mathbf{m}_h^{n+1}) - \Theta(\mathbf{m}_h^n) + \frac{1}{2}\int_\Omega \|D\mathbf{m}_h^{n+1}\|_F^2 - \|D\mathbf{m}_h^n\|_F^2\,\mathrm{d}x\\ &= \Theta(\mathbf{m}_h^{n+1}) - \Theta(\mathbf{m}_h^n) - \alpha\Theta'(\mathbf{m}_h^n)[\mathbf{w}_h^n] +\\ &\quad + (\alpha^2/2-\alpha)\int_\Omega \|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x.\end{aligned}$$

We assume here and will show below for the specification of Θ that corresponds to the model problem that we have

$$\left|\Theta(\mathbf{m}_h^{n+1}) - \Theta(\mathbf{m}_h^n) - \alpha\Theta'(\mathbf{m}_h^n)[\mathbf{w}_h^n]\right| \le C_\Theta\alpha^2\|\mathbf{w}_h^n\|_{H^1(\Omega)}^2. \tag{13}$$

This estimate may be regarded as a Taylor expansion of Θ but its proof also requires to bound the difference between $\mathbf{m}_h^{n+1}$ and $\mathbf{m}_h^n+\alpha\mathbf{w}_h^n$. With a Poincaré inequality in (13) we thus have

$$\begin{aligned}E(\mathbf{m}_h^{n+1}) - E(\mathbf{m}_h^n) &\le (-\alpha+\alpha^2/2+C_PC_\Theta\alpha^2)\int_\Omega \|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x\\ &\le -\alpha(1-\alpha/2-C_PC_\Theta\alpha)\int_\Omega \|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x.\end{aligned}$$

If α is sufficiently small so that $(1-\alpha/2-C_PC_\Theta\alpha)\ge 1/2$ then it follows that

$$E(\mathbf{m}_h^{n+1}) - E(\mathbf{m}_h^n) \le -\frac{\alpha}{2}\int_\Omega \|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x$$

and after summation over $n=0,1,\dots,N$

$$E(\mathbf{m}_h^{N+1}) + \frac{\alpha}{2}\sum_{n=0}^{N}\int_\Omega \|D\mathbf{w}_h^n\|_F^2\,\mathrm{d}x \le E(\mathbf{m}_h^0).$$

This proves the stability and convergence of our numerical method.

5.2 Application to the model problem

In our model (12), the functional $\Theta(\mathbf{m})$ is given by

$$\Theta(\mathbf{m}) = \int_\Omega 1-(\mathbf{e}\cdot\mathbf{m})^2 - \mathbf{f}\cdot\mathbf{m} + \frac{\mu_0}{2}\mathbf{m}\cdot\nabla u_{\mathbf{m}}\,\mathrm{d}x.$$

We claim that for this functional the estimate (13) is satisfied. For this we first show that

$$|\Theta(\mathbf{m})-\Theta(\tilde{\mathbf{m}})| \le C_1 \|\mathbf{m}-\tilde{\mathbf{m}}\|_{L^2(\Omega)} \tag{14}$$

provided that $\|\mathbf{m}\|_{L^2(\Omega)}$, $\|\tilde{\mathbf{m}}\|_{L^2(\Omega)} \le C$. With Hölder's inequality we verify that

$$\begin{aligned}\Theta(\mathbf{m})-\Theta(\tilde{\mathbf{m}}) \le \int_\Omega (\mathbf{e}\cdot\mathbf{m})^2-(\mathbf{e}\cdot\tilde{\mathbf{m}})^2\,\mathrm{d}x + \|\mathbf{f}\|_{L^2(\Omega)}\|\mathbf{m}-\tilde{\mathbf{m}}\|_{L^2(\Omega)} \\ + \frac{\mu_0}{2}\int_\Omega \nabla u_{\mathbf{m}}\cdot\mathbf{m}-\nabla u_{\tilde{\mathbf{m}}}\cdot\tilde{\mathbf{m}}\,\mathrm{d}x.\end{aligned}$$

Using Cauchy-Schwarz inequality and $\|\mathbf{e}\|_2 = 1$, the first term on the right-hand side can be bounded by

$$\int_\Omega (\mathbf{e}\cdot\mathbf{m})^2-(\mathbf{e}\cdot\tilde{\mathbf{m}})^2\,\mathrm{d}x \le \|\mathbf{m}-\tilde{\mathbf{m}}\|_{L^2(\Omega)}\|\mathbf{m}+\tilde{\mathbf{m}}\|_{L^2(\Omega)}.$$

Similarly, we have with (4) that

$$\begin{aligned}\int_\Omega \nabla u_{\mathbf{m}}\cdot\mathbf{m}-\nabla u_{\tilde{\mathbf{m}}}\cdot\tilde{\mathbf{m}}\,\mathrm{d}x &= \int_{\mathbb{R}^3} \|\nabla u_{\mathbf{m}}\|_2^2-\|\nabla u_{\tilde{\mathbf{m}}}\|_2^2\,\mathrm{d}x \\ &\le \|\nabla(u_{\mathbf{m}}-u_{\tilde{\mathbf{m}}})\|_{L^2(\mathbb{R}^3)}\|\nabla(u_{\mathbf{m}}+u_{\tilde{\mathbf{m}}})\|_{L^2(\mathbb{R}^3)} \\ &\le \|\mathbf{m}-\tilde{\mathbf{m}}\|_{L^2(\Omega)}(\|\mathbf{m}\|_{L^2(\Omega)}+\|\tilde{\mathbf{m}}\|_{L^2(\Omega)}).\end{aligned}$$

Hence, the property (14) follows from the assumed bounds on $\mathbf{m}$ and $\tilde{\mathbf{m}}$. We next show that

$$\left|\Theta(\mathbf{m}+\alpha\mathbf{w})-\Theta(\mathbf{m})-\alpha\Theta'(\mathbf{m})[\mathbf{w}]\right| \le C_2\alpha^2\|\mathbf{w}\|_{L^2(\Omega)}^2. \tag{15}$$

Straightforward calculations lead to

$$\begin{aligned}&\Theta(\mathbf{m}+\alpha\mathbf{w})-\Theta(\mathbf{m})-\alpha\Theta'(\mathbf{m})[\mathbf{w}] \\ &= \int_\Omega (\mathbf{e}\cdot(\mathbf{m}+\alpha\mathbf{w}))^2-(\mathbf{e}\cdot\mathbf{m})^2-2\alpha(\mathbf{e}\cdot\mathbf{m})(\mathbf{e}\cdot\mathbf{w})\,\mathrm{d}x \\ &\quad + \frac{\mu_0}{2}\int_\Omega \nabla u_{\mathbf{m}+\alpha\mathbf{w}}\cdot(\mathbf{m}+\alpha\mathbf{w})-\nabla u_{\mathbf{m}}\cdot\mathbf{m}-2\alpha\nabla u_{\mathbf{m}}\cdot\mathbf{w}\,\mathrm{d}x \\ &= \alpha^2\int_\Omega (\mathbf{e}\cdot\mathbf{w})^2\,\mathrm{d}x + \frac{\mu_0}{2}\int_\Omega \nabla u_{\mathbf{m}+\alpha\mathbf{w}}\cdot(\mathbf{m}+\alpha\mathbf{w})-\nabla u_{\mathbf{m}}\cdot\mathbf{m}-2\alpha\nabla u_{\mathbf{m}}\cdot\mathbf{w}\,\mathrm{d}x \\ &\le \alpha^2\|\mathbf{w}\|_{L^2(\Omega)}^2 + \frac{\mu_0}{2}\int_\Omega \nabla u_{\mathbf{m}+\alpha\mathbf{w}}\cdot(\mathbf{m}+\alpha\mathbf{w})-\nabla u_{\mathbf{m}}\cdot\mathbf{m}-2\alpha\nabla u_{\mathbf{m}}\cdot\mathbf{w}\,\mathrm{d}x.\end{aligned}$$

Using (3) and $\nabla u_{\mathbf{m}+\alpha\mathbf{w}} = \nabla u_{\mathbf{m}} + \alpha\nabla u_{\mathbf{w}}$, we find that the second integral on the right-hand side of the previous estimate satisfies

$$\begin{aligned}&\int_{\mathbb{R}^3} \|\nabla u_{\mathbf{m}+\alpha\mathbf{w}}\|_2^2-\|\nabla u_{\mathbf{m}}\|_2^2\,\mathrm{d}x - 2\alpha\int_\Omega \nabla u_{\mathbf{m}}\cdot\mathbf{w}\,\mathrm{d}x \\ &= \int_{\mathbb{R}^3} \|\nabla u_{\mathbf{m}}\|_2^2+2\alpha\nabla u_{\mathbf{m}}\cdot\nabla u_{\mathbf{w}}+\|\nabla u_{\alpha\mathbf{w}}\|_2^2-\|\nabla u_{\mathbf{m}}\|_2^2\,\mathrm{d}x - 2\alpha\int_\Omega \nabla u_{\mathbf{m}}\cdot\mathbf{w}\,\mathrm{d}x \\ &= \int_{\mathbb{R}^3} \|\nabla u_{\alpha\mathbf{w}}\|_2^2\,\mathrm{d}x \le \alpha^2\int_\Omega \|\mathbf{w}\|_2^2\,\mathrm{d}x.\end{aligned}$$

This leads to the estimate (15). The combination of (14) and (15) implies that for iterates $\mathbf{m}_h^n$, $\mathbf{w}_h^n$, and $\mathbf{m}_h^{n+1}$ we have, noting that $\|\mathbf{m}_h^n\|_{L^2(\Omega)}$, $\|\mathbf{w}_h^n\|_{L^2(\Omega)} \le C$,

$$\begin{aligned}\big|\Theta(\mathbf{m}_h^{n+1}) - \Theta(\mathbf{m}_h^n) - \alpha\Theta'(\mathbf{m}_h^n)[\mathbf{w}_h]\big| &\le \big|\Theta(\mathbf{m}_h^{n+1}) - \Theta(\mathbf{m}_h^n + \alpha\mathbf{w}_h^n)\big| + C_2\alpha^2\|\mathbf{w}_h^n\|_{L^2(\Omega)}^2\\ &\le C_1\|\mathbf{m}_h^{n+1} - (\mathbf{m}_h^n + \alpha\mathbf{w}_h^n)\|_{L^2(\Omega)} + C_2\alpha^2\|\mathbf{w}_h^n\|_{L^2(\Omega)}^2.\end{aligned}$$

For every node $z \in \mathcal{N}_h$ we have

$$\begin{aligned}\|\mathbf{m}_h^{n+1}(z) - \mathbf{m}_h^n(z) - \alpha\mathbf{w}_h^n(z)\|_2 &= \left\| \frac{\mathbf{m}_h^n(z) + \alpha\mathbf{w}_h^n(z)}{\|\mathbf{m}_h^n(z) + \alpha\mathbf{w}_h^n(z)\|_2} - \mathbf{m}_h^n(z) - \alpha\mathbf{w}_h^n(z)\right\|_2\\ &= \|\mathbf{m}_h^n(z) + \alpha\mathbf{w}_h^n(z)\|_2 - 1\\ &= \big(1 + \alpha^2\|\mathbf{w}_h^n(z)\|_2^2\big)^{1/2} - 1 \le \frac{\alpha^2}{2}\|\mathbf{w}_h^n(z)\|_2^2\end{aligned}$$

and the continuous embedding $H^1(\Omega) \hookrightarrow L^4(\Omega)$ yields

$$\|\mathbf{m}_h^{n+1} - (\mathbf{m}_h^n + \alpha\mathbf{w}_h^n)\|_{L^2(\Omega)} \le C\alpha^2\|\mathbf{w}_h^n\|_{L^4(\Omega)}^2 \le C'\alpha^2\|\mathbf{w}_h^n\|_{H^1(\Omega)}^2.$$

Hence, the model problem fulfills the desired property (13).

6 The string method

We aim at computing an unstable critical configuration whose energy is minimal among all maxima of curves connecting two given states, i.e., we compute a saddle point. For this, we adopt a method proposed in [13] that does not require an energy that is defined on a linear space as it is needed for classical mountain pass algorithms. The difficulty for the energy functional under consideration is the appropriate incorporation of the pointwise unit length constraint.

To describe the problem in a continuous setting, we assume that we are given two local minima $\mathbf{m}_0 \in \mathscr{A}$ and $\mathbf{m}_1 \in \mathscr{A}$ of the energy functional $E : \mathscr{A} \to \mathbb{R}$, where the space of admissible magnetic configurations $\mathscr{A}$ is defined by

$$\mathscr{A} = \{\mathbf{m} \in H^1(\Omega;\mathbb{R}^3) : \|\mathbf{m}(x)\|_2 = 1 \text{ for almost every } x \in \Omega\}.$$

We then consider a family of curves connecting $\mathbf{m}_0$ and $\mathbf{m}_1$ parametrized by $t \ge 0$, i.e., a continuous mapping

$$\varphi(t,\cdot) : [0,1] \to \mathscr{A}$$

such that $\varphi(t,0) = \mathbf{m}_0$ and $\varphi(t,1) = \mathbf{m}_1$.

Our aim is it to compute a curve connecting $\mathbf{m}_0, \mathbf{m}_1 \in \mathscr{A}$ such that the component of ∇E normal to φ vanishes, i.e.

$$(\nabla E)^\perp(\varphi) = 0, \tag{16}$$

where

$$(\nabla E)^\perp(\varphi) = \nabla E(\varphi) - (\nabla E(\varphi)\cdot\tau)\tau$$

and τ denotes the unit tangent vector of φ. A path φ satisfying (16) is called a minimum energy path (MEP).

To numerically evaluate such a path, we deploy the string method which was proposed in [11, 12] and modified in [13]. The modified string method stands out due its simplicity. Another known method to compute MEPs is the nudged elastic band (NEB) method; cf. [24].

Algorithm (modified string method): Let two local minima $\mathbf{m}_0, \mathbf{m}_1 \in \mathscr{A}$ of the energy functional E be given. Define a path $\varphi^0 : \{0,\dots,N\} \to \mathscr{A}$ as a collection of $N+1$ points with $\varphi^0(0) = \mathbf{m}_0$

and $\varphi^0(N) = \mathbf{m}_1$. The points $\varphi^0(i)$, $i = 1,\dots,N-1$, are computed via interpolation. Iterate for $n = 0,1,2,\dots$ the following steps:

(1) Let $\varphi_*^n(0) = \varphi^n(0)$ and $\varphi_*^n(N) = \varphi^n(N)$. Compute for each configuration $\varphi^n(i)$ with $i = 1,\dots,N-1$ a single iteration step of the minimization algorithm as proposed in Sect. 5.1 and assign the result to $\varphi_*^n(i)$.
(2) Compute via interpolation the new path $\left(\varphi^{n+1}(i)\right)_{i=0,\dots,N}$ as a reparametrization of $\left(\varphi_*^n(i)\right)_{i=0,\dots,N}$.

The interpolation used in the modified string method can be done in arbitrary ways. We choose to interpolate geodesically on the sphere to obtain a reparametrization of the string.

The advantage of the modified string method is that the point-wise constraint $\|\mathbf{m}(x)\|_2 = 1$, $x \in \Omega$, is inherited from the above minimization algorithm. In Figure 2, an initial path and an MEP as the result of the modified string method are shown.

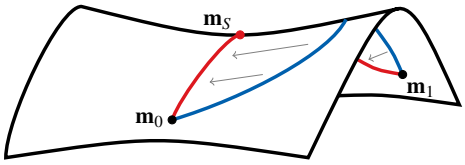


Fig. 2 A scheme for the modified string method showing the initial path (blue line) and the MEP (red line) connecting the two energy minima $\mathbf{m}_0$ and $\mathbf{m}_1$. The configuration $\mathbf{m}_S$ is the saddle point on the MEP.

7 Numerical examples

In this section, we first verify our implementation using a standard problem, afterwards we investigate the overall complexity of the presented numerical method and finally we compute an MEP for a cubic magnetic particle.

All tests were performed on a single core of an Intel Xeon X5482 processor with 3.2 GHz and 64 GB of RAM. The programming was done in C++ and is based on the hierarchical matrix library A$\mathscr{H}$MED[1]. Furthermore, we set the minimal block cluster size of the created $\mathscr{H}$-matrices to 50 and the relative blockwise approximation accuracy to $1e-3$.

7.1 Validation of implementation

A problem proposed by A. Hubert, University of Erlangen-Nuremberg, to check the computation of the different energies in (1) is to calculate the single domain limit of a cubic magnetic particle. This is the length of the cube for which the so-called flower and vortex state have equal energies. The test is also known as the μ-mag standard problem #3; see [26].

In our tests, the cube was discretized into 24576 tetrahedra and 3072 triangles. Fig. 3 shows the reduced energy, as proposed in [26], relative to the size of the cube. Our numerical tests show a single domain limit of 8.23, which represents the theoretically expected value of approximately 8.

[1] http://bebendorf.ins.uni-bonn.de/AHMED.html

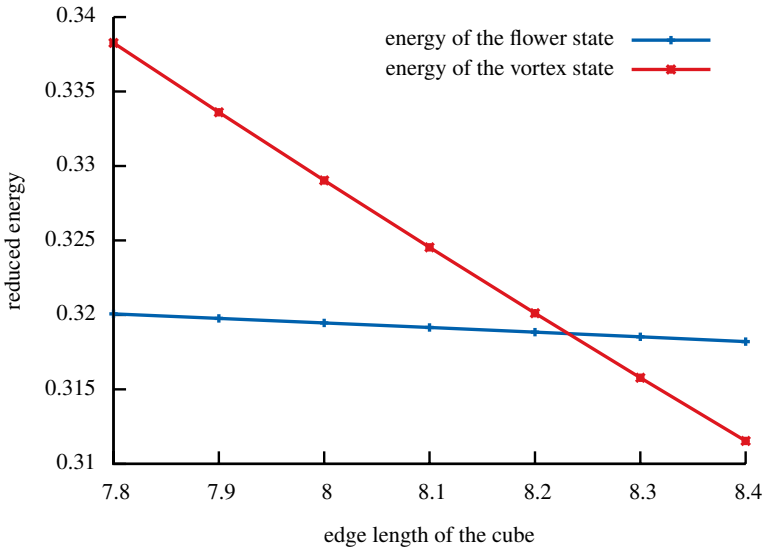


Fig. 3 The resulting energy minima of the μ-mag standard problem #3.

7.2 Time and memory consumption

To demonstrate the almost linear behavior of the presented numerical algorithms in terms of memory and time consumption, we chose different discretizations of the unit cube which were created using netgen[2].

As a first test, we look at the construction of the matrices (8), (9), (10), and (11). These matrices have to be set up only once for a certain geometry and approximation accuracy in an initialization step. Fig. 4 shows that the time and memory needed to construct the matrices (8) and (9) is linear up to logarithmic terms with respect to the number of tetrahedra. Furthermore, from Fig. 5 it can be seen that the construction of (10) and (11) is quasi-optimal with respect to the number of triangles.

For the string method, the minimization algorithm from Sect. 5.1 is the dominant part of the computational time. As can be seen from Fig. 6, these minimization steps of the algorithm presented in Section 5.1 are almost linear in terms of the number of tetrahedra. Hence, using $\mathscr{H}$-matrices to compute the different energies results in a numerical scheme which has logarithmic-linear complexity.

7.3 Minimum energy paths

In the following test, we use the simplified string method from Sect. 6 to compute an MEP. The geometry and the parameter configuration are chosen as in the μ-mag standard problem #3 with a cube edge length of 8.2. Hence, the two energy minima are the flower and the vortex state. For the

[2] http://www.hpfem.jku.at/netgen/

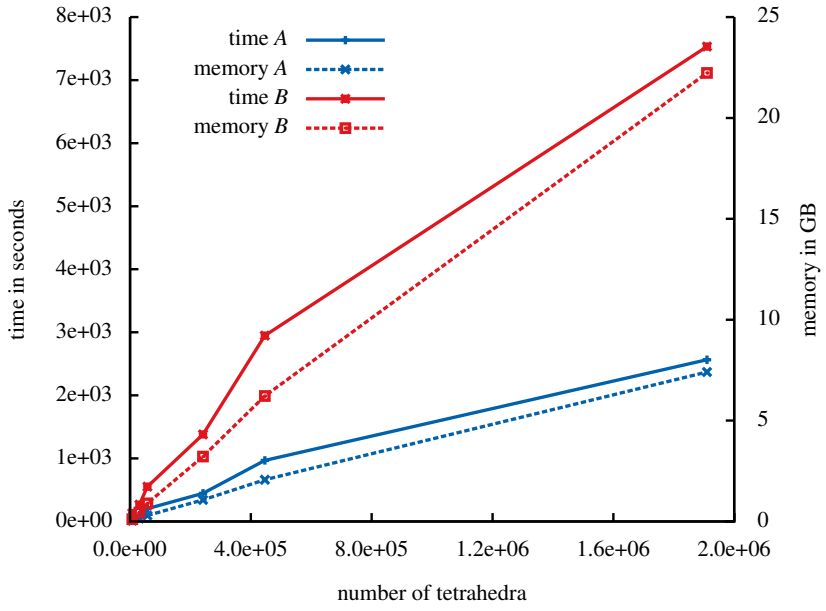


Fig. 4 The time and memory consumption of the matrices A and B; cf. (8) and (9).

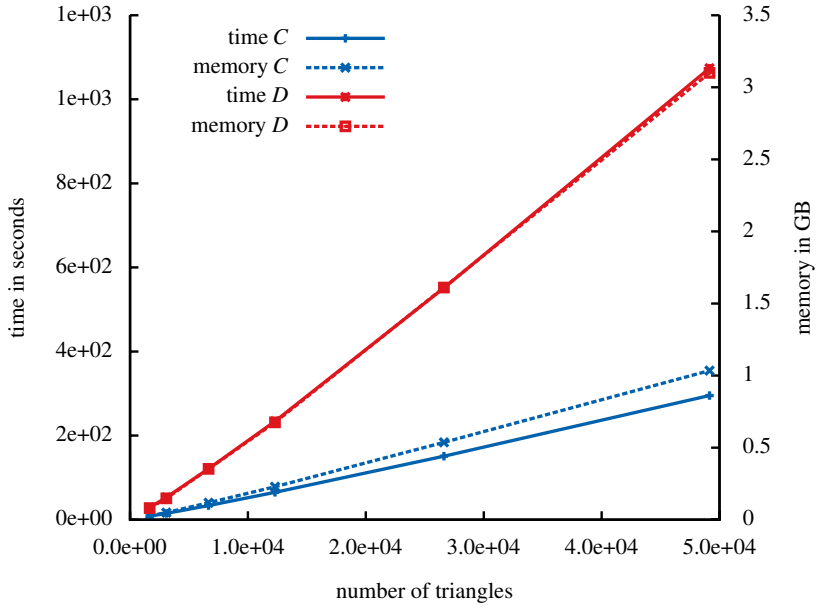


Fig. 5 The time and memory consumption of the matrices C and D; cf. (10) and (11).

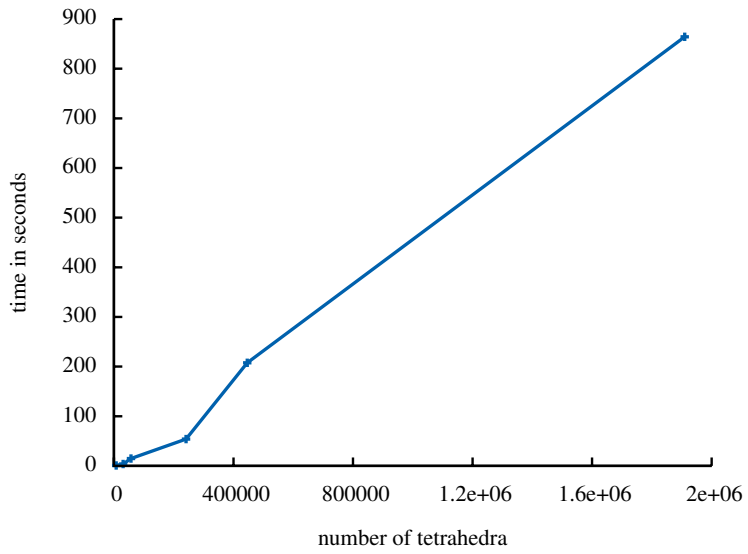


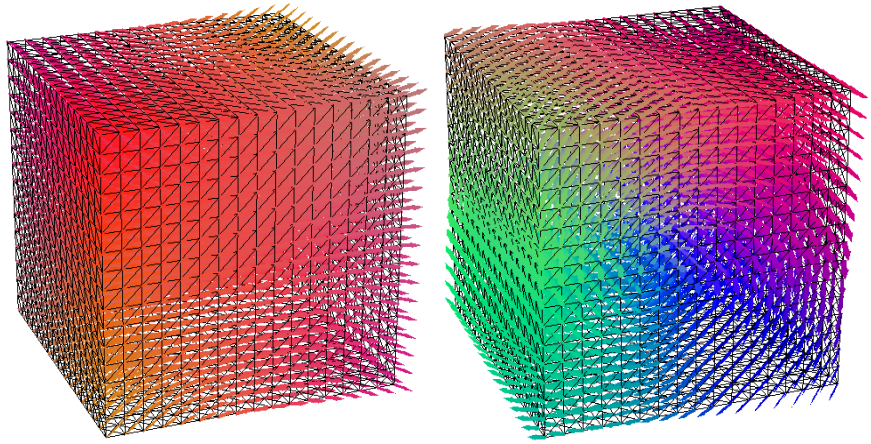
Fig. 6 The time needed for a single minimization step.

tests, the cube was discretized into 24576 tetrahedra and 3072 triangles. For each iteration step the discretized path consists of 41 single magnetization points.

In Fig. 7, the two energy minima (flower and vortex state) which are used to define our initial path are depicted. Different colors of the arrows representing the magnetization indicate different directions in space. The initial magnetization points in between the energy minima were computed using geodesical interpolation.

A comparison of the reduced energies of the initialized path and the MEP is shown in Fig. 8. Here, the magnetization point 25 maximizes the reduced energy of the MEP and is about 28.5% lower than the energy barrier of the initial path. Furthermore, in Table 1 slides of selected magnetization states of the MEP are shown. It is remarkable that in the magnetization point 40 the vortex is opening on the front side and closing on the back.

The limiting factor of the numerical experiments is the computational time since all the single magnetization points along the discretized path need to be minimized consecutively. Even with the employed fast methods approximately two days were needed to perform the computations.



(a) Flower state (b) Vortex state

Fig. 7 A model of the two energy minima used to compute the MEP.

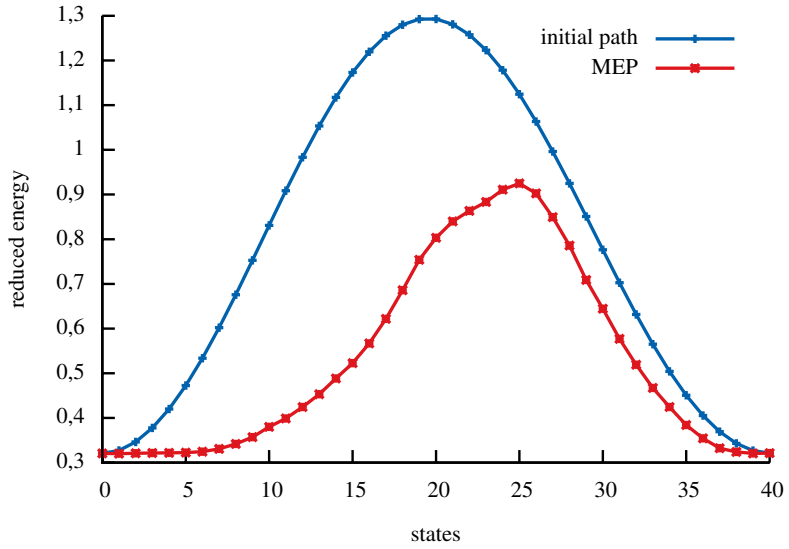
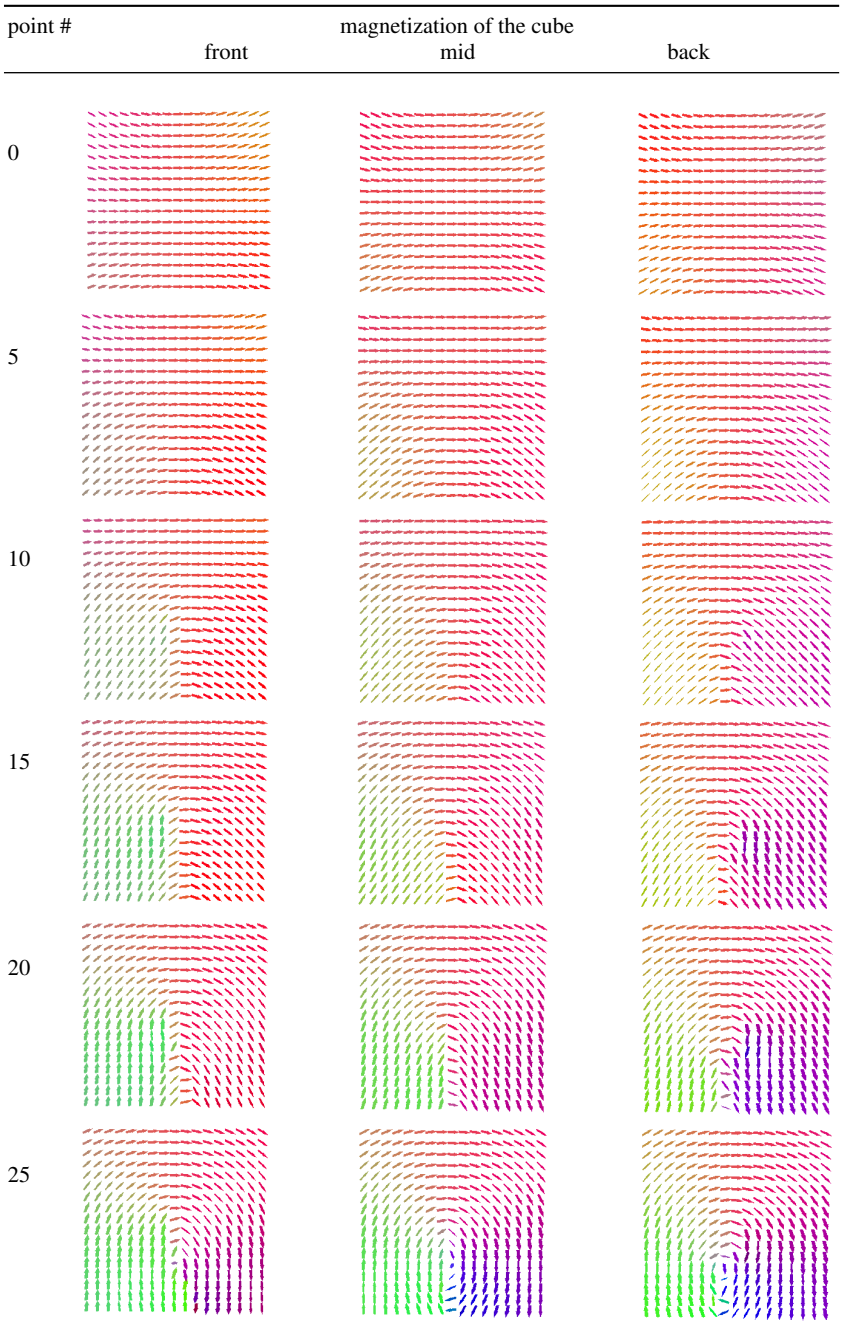


Fig. 8 The reduced energy of the magnetization points of the initial path and the MEP connecting the flower (left) and the vortex state (right).


point #
magnetization of the cube
front
mid
back
0
5
10
15
20
25

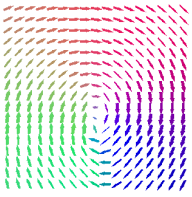
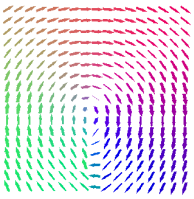
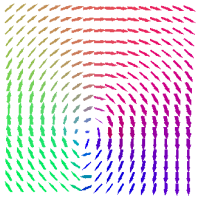
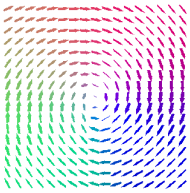
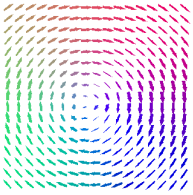
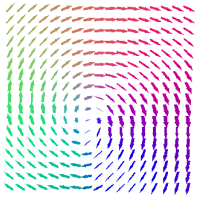
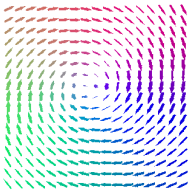
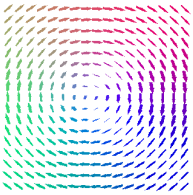
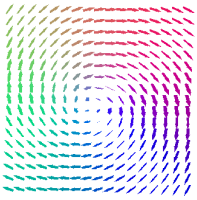
point #	magnetization of the cube		
	front	mid	back
30			
35			
40			

Table 1: The different magnetization states of the MEP connecting the vortex and the flower state.

Appendix

The efficient numerical evaluation of the singular integrals from Sect. 3; see (8), (9), (10) and (11), is a challenging task. One way to overcome difficulties is to use spherical coordinates, which has the disadvantage of awkward integration bounds.

Another way is to use the Duffy transformation as proposed in [10]. The general principle is to transform a triangle onto a square to get rid of the singularity. The following example demonstrates this approach. Using the transformation $x = \xi$ and $y = \xi\eta$, one obtains that

$$\int_0^1 \int_0^x \frac{1}{x+y}\,\mathrm{d}y\,\mathrm{d}x = \int_0^1 \int_0^1 \frac{1}{1+\eta}\,\mathrm{d}\eta\,\mathrm{d}\xi$$

and integration can be done by using standard methods.

This principle has already been applied to the integration on pairs of triangles in [28] and can be used to evaluate the integrals in (10) and (11). Similar ideas can be applied to the combination of a triangle and a tetrahedron. For the kernel function $\kappa : \mathbb{R}^3 \times \mathbb{R}^2 \to \mathbb{R}$ we need to evaluate the following integration on the reference triangle and tetrahedron

$$I := \int_0^1 \int_0^{1-x_1} \int_0^{1-x_1-x_2} \int_0^1 \int_0^{1-y_1} \kappa(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{y}\,\mathrm{d}\mathbf{x}$$

with $\mathbf{x} = (x_1, x_2, x_3)$ and $\mathbf{y} = (y_1, y_2)$. By introducing relative coordinates, the integral I can be transformed to

$$I = \int_0^1 \int_0^{\tilde{x}_1} \int_0^{\tilde{x}_1-\tilde{x}_2} \int_{-\tilde{x}_1}^{1-\tilde{x}_1} \int_{-\tilde{x}_2}^{\tilde{y}_1+\tilde{x}_1-\tilde{x}_2} \kappa\left(\begin{pmatrix}1-\tilde{x}_1\\ \tilde{x}_2\\ \tilde{x}_3\end{pmatrix}, \begin{pmatrix}1-\tilde{y}_1-\tilde{x}_1\\ \tilde{y}_2+\tilde{x}_2\\ 0\end{pmatrix}\right) \mathrm{d}\tilde{\mathbf{y}}\,\mathrm{d}\tilde{\mathbf{x}}.$$

The kernel is singular only for $\tilde{\mathbf{y}} = 0, \tilde{x}_3 = 0$, which we can be eliminated using a Duffy transformation. Similar to the approach in [28], we split the integral into six different domains.

$$\left\{\begin{matrix}-1 \le \tilde{y}_1 \le 0\\ -1 \le \tilde{y}_2 \le \tilde{y}_1\\ -\tilde{y}_2 \le \tilde{x}_1 \le 1\\ -\tilde{y}_2 \le \tilde{x}_2 \le \tilde{x}_1\\ 0 \le \tilde{x}_3 \le \tilde{x}_1 - \tilde{x}_2\end{matrix}\right\} \cup \left\{\begin{matrix}-1 \le \tilde{y}_1 \le 0\\ \tilde{y}_1 \le \tilde{y}_2 \le 0\\ -\tilde{y}_1 \le \tilde{x}_1 \le 1\\ -\tilde{y}_2 \le \tilde{x}_2 \le \tilde{x}_1 + \tilde{y}_1 - \tilde{y}_2\\ 0 \le \tilde{x}_3 \le \tilde{x}_1 - \tilde{x}_2\end{matrix}\right\} \cup \left\{\begin{matrix}-1 \le z_1 \le 0\\ 0 \le \tilde{y}_2 \le 1 + \tilde{y}_1\\ \tilde{y}_2 - \tilde{y}_1 \le \tilde{x}_1 \le 1\\ 0 \le \tilde{x}_2 \le \tilde{x}_1 + \tilde{y}_1 - \tilde{y}_2\\ 0 \le \tilde{x}_3 \le \tilde{x}_1 - \tilde{x}_2\end{matrix}\right\}$$
$$\cup \left\{\begin{matrix}0 \le \tilde{y}_1 \le 1\\ -1 + \tilde{y}_1 \le \tilde{y}_2 \le 0\\ -\tilde{y}_2 \le \tilde{x}_1 \le 1 - \tilde{y}_1\\ -\tilde{y}_2 \le \tilde{x}_2 \le \tilde{x}_1\\ 0 \le \tilde{x}_3 \le \tilde{x}_1 - \tilde{x}_2\end{matrix}\right\} \cup \left\{\begin{matrix}0 \le \tilde{y}_1 \le 1\\ 0 \le \tilde{y}_2 \le \tilde{y}_1\\ 0 \le \tilde{x}_1 \le 1 - \tilde{y}_1\\ 0 \le \tilde{x}_2 \le \tilde{x}_1\\ 0 \le \tilde{x}_3 \le \tilde{x}_1 - \tilde{x}_2\end{matrix}\right\} \cup \left\{\begin{matrix}0 \le \tilde{y}_1 \le 1\\ \tilde{y}_1 \le \tilde{y}_2 \le 1\\ \tilde{y}_2 - \tilde{y}_1 \le \tilde{x}_1 \le 1 - \tilde{y}_1\\ 0 \le \tilde{x}_2 \le \tilde{y}_1 - \tilde{y}_2 + \tilde{x}_1\\ 0 \le \tilde{x}_3 \le \tilde{x}_1 - \tilde{x}_2\end{matrix}\right\}.$$

The resulting integrals $I := I_1 + \ldots + I_6$, can be transformed onto a five-dimensional unit cube. The single terms are given in the following way.

- I_1:

$$I_1 = \int_{(0,1)^5} p_1 \kappa\left(\begin{pmatrix}1-\eta_5\\ \eta_5(1-\eta_1+\eta_1\eta_2)\\ \eta_1\eta_4\eta_5(1-\eta_2)\end{pmatrix}, \begin{pmatrix}1-\eta_5+\eta_1\eta_2\eta_3\eta_5\\ \eta_5(1-\eta_1)\end{pmatrix}\right) \mathrm{d}\eta,$$
$$p_1 := \eta_1^3\eta_2\eta_5^4(1-\eta_2)$$

- I_2:

$$I_2 = \int_{(0,1)^5} p_2 \kappa \left(\begin{pmatrix} 1-\eta_5 \\ \eta_1\eta_5(1-\eta_2+\eta_2\eta_3) \\ \eta_4\eta_5(1-\eta_1+\eta_1\eta_2-\eta_1\eta_2\eta_3) \end{pmatrix}, \begin{pmatrix} 1-\eta_5+\eta_1\eta_2\eta_5 \\ \eta_1\eta_5(1-\eta_2) \end{pmatrix} \right) \mathrm{d}\eta,$$
$$p_2 := \eta_1^2\eta_2\eta_5^4(1-\eta_1+\eta_1\eta_2-\eta_1\eta_2\eta_3)$$

- I_3:

$$I_3 = \int_{(0,1)^5} p_3 \kappa \left(\begin{pmatrix} 1-\eta_5 \\ \eta_1\eta_5(1-\eta_2) \\ \eta_4\eta_5(1-\eta_1+\eta_1\eta_2) \end{pmatrix}, \begin{pmatrix} 1-\eta_5+\eta_1\eta_2\eta_3\eta_5 \\ \eta_1\eta_5(1-\eta_2\eta_3) \end{pmatrix} \right) \mathrm{d}\eta,$$
$$p_3 := \eta_1^2\eta_2\eta_5^4(1-\eta_1+\eta_1\eta_2),$$

- I_4:

$$I_4 = \int_{(0,1)^5} p_4 \kappa \left(\begin{pmatrix} 1-\eta_5+\eta_1\eta_2\eta_3\eta_5 \\ \eta_1\eta_5(1-\eta_2\eta_3) \\ \eta_4\eta_5(1-\eta_1) \end{pmatrix}, \begin{pmatrix} 1-\eta_5 \\ \eta_1\eta_5(1-\eta_2) \end{pmatrix} \right) \mathrm{d}\eta,$$
$$p_4 := \eta_1^2\eta_2\eta_5^4(1-\eta_1)$$

- I_5:

$$I_5 = \int_{(0,1)^5} p_5 \kappa \left(\begin{pmatrix} 1-\eta_5+\eta_1\eta_2\eta_5 \\ \eta_1\eta_5(1-\eta_2) \\ \eta_4\eta_5(1-\eta_1) \end{pmatrix}, \begin{pmatrix} 1-\eta_5 \\ \eta_1\eta_5(1-\eta_2+\eta_2\eta_3) \end{pmatrix} \right) \mathrm{d}\eta,$$
$$p_5 := \eta_1^2\eta_2\eta_5^4(1-\eta_1)$$

- I_6:

$$I_6 = \int_{(0,1)^5} p_6 \kappa \left(\begin{pmatrix} 1-\eta_5+\eta_1\eta_2\eta_3\eta_5 \\ \eta_5(1-\eta_1) \\ \eta_1\eta_4\eta_5(1-\eta_2\eta_3) \end{pmatrix}, \begin{pmatrix} 1-\eta_5 \\ \eta_5(1-\eta_1+\eta_1\eta_2) \end{pmatrix} \right) \mathrm{d}\eta,$$
$$p_6 := \eta_1^3\eta_2\eta_5^4(1-\eta_2\eta_3)$$

The integrals $I_1, \ldots, I_6$, can be evaluated efficiently using standard quadrature formulas, e.g., Gaussian quadrature.

References

1. F. Alouges. A new algorithm for computing liquid crystal stable configurations: the harmonic mapping case. *SIAM J. Numer. Anal.*, 34:1708–1726, 1997.
2. F. Alouges, S. Conti, A. DeSimone, and Y. Pokern. Energetics and switching of quasi-uniform states in small ferromagnetic particles. *Mathematical Modelling and Numerical Analysis*, 38:235–248, 2004.
3. J. Barnes and P. Hut. A hierarchical $\mathcal{O}(N \log N)$ force calculation algorithm. *Nature*, 324:446–449, 1986.
4. S. Bartels. Stability and convergence of finite-element approximation schemes for harmonic maps. *SIAM J. Numer. Anal.*, 43:220–238, 2005.
5. S. Bartels and A. Prohl. Convergence of an implicit finite element method for the Landau-Lifshitz-Gilbert equation. *SIAM J. Numer. Anal.*, 44(4):1405–1419 (electronic), 2006.

6. M. Bebendorf. Approximation of boundary element matrices. *Numer. Math.*, 86(4):565–589, 2000.
7. M. Bebendorf. *Hierarchical Matrices: A Means to Efficiently Solve Elliptic Boundary Value Problems*, volume 63 of *Lecture Notes in Computational Science and Engineering (LNCSE)*. Springer, 2008. ISBN 978-3-540-77146-3.
8. Y. S. Choi and P. J. McKenna. A mountain pass method for the numerical solution of semilinear elliptic problems. *Nonlinear Anal.*, 20(4):417–437, February 1993.
9. A. DeSimone, R. V. Kohn, S. Müller, and F. Otto. Magnetic microstructures—a paradigm of multiscale problems. In *ICIAM 99 (Edinburgh)*, pages 175–190. Oxford Univ. Press, Oxford, 2000.
10. M. Duffy. Quadrature over a pyramid or cube of integrands with a singularity at a vertex. *SIAM J. Numer. Anal.*, 19:1260–1262, 1982.
11. W. E, W. Ren, and E. Vanden-Eijnden. String method for the study of rare events. *Physical Review B*, 66:052301 1–4, 2002.
12. W. E, W. Ren, and E. Vanden-Eijnden. Energy landscape and thermally activated switching of submicron-sized ferromagnetic elements. *Journal of Applied Physics*, 93:2275–2282, 2003.
13. W. E, W. Ren, and E. Vanden-Eijnden. Simplified and improved string method for computing the minimum energy paths in barrier-crossing events. *The Journal of Chemical Physics*, 126:164103 1–8, 2007.
14. J. Miguel García-Cervera. Numerical micromagnetics: A review. *Bol. Soc. Esp. Matem. Apl.*, 39:103–135, 2007.
15. L. Grasedyck and W. Hackbusch. Construction and arithmetics of $\mathscr{H}$-matrices. *Computing*, 70:295–334, 2003.
16. L. F. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.
17. L. F. Greengard and V. Rokhlin. A new version of the fast multipole method for the Laplace equation in three dimensions. In *Acta numerica, 1997*, volume 6 of *Acta Numer.*, pages 229–269. Cambridge Univ. Press, Cambridge, 1997.
18. W. Hackbusch. A sparse matrix arithmetic based on $\mathscr{H}$-matrices. Part I: Introduction to $\mathscr{H}$-matrices. *Computing*, 62(2):89–108, 1999.
19. W. Hackbusch. *Hierarchische Matrizen*. Springer Verlag, 2009.
20. W. Hackbusch and B. N. Khoromskij. A sparse $\mathscr{H}$-matrix arithmetic. Part II: Application to multi-dimensional problems. *Computing*, 64(1):21–47, 2000.
21. W. Hackbusch and Z. P. Nowak. On the fast matrix multiplication in the boundary element method by panel clustering. *Numer. Math.*, 54(4):463–491, 1989.
22. R. Hertel and H. Kronmüller. Finite element calculations on the single-domain limit of a ferromagnetic cubea solution to mag standard problem no. 3. *Journal of Magnetism and Magnetic Materials*, 238(23):185 – 199, 2002.
23. A. Hubert and R. Schäfer. *Magnetic Domains – The Analysis of Magnetic Microstructures*. Springer-Verlag, 2009.
24. H. Jónsson, G. Mills, and K. W. Jacobsen. Nudged elastic band method for finding minimum energy paths of transitions. In *Classical and Quantum Dynamics in Condensed Phase Simulations*. World Scientific, Singapore, 1998.
25. M. Kruzík and A. Prohl. Recent developments in the modeling, analysis, and numerics of ferromagnetism. *SIAM Rev.*, 48(3):439–483, March 2006.
26. R.D. McMichael. Standard problem number 3 - problem specification and reported solutions. *http://www.ctcms.nist.gov/∼rdm/mumag.html*, 2008.
27. N. Popović and D. Praetorius. Applications of h-matrix techniques in micromagnetics. *Computing 74*, 3:177–204, 2005.
28. S. Sauter and C. Schwab. *Boundary element methods*. Springer-Verlag, 2011.
29. M. E. Schabes and H. N. Bertram. Magnetization processes in ferromagnetic cubes. *Journal of Applied Physics*, 64(3):1347 –1357, aug 1988.
30. E. E. Tyrtyshnikov. Mosaic-skeleton approximations. *Calcolo*, 33(1-2):47–57 (1998), 1996. Toeplitz matrices: structures, algorithms and applications (Cortona, 1996).