$\mathcal{H}$-Matrix Techniques for Stray-Field Computations in Computational Micromagnetics

Nikola Popović[1] and Dirk Praetorius[2]

[1] Department of Mathematics and Statistics,
Boston University,
111 Cummington Street,
Boston, MA 02215, U.S.A.,
Popovic@math.bu.edu

[2] Institute for Analysis and Scientific Computing,
Vienna University of Technology,
Wiedner Hauptstraße 8-10,
A-1040 Vienna, Austria,
Dirk.Praetorius@tuwien.ac.at

Abstract. A major task in the simulation of micromagnetic phenomena is the effective computation of the stray-field $\mathbf{H}$ and/or of the corresponding energy, where $\mathbf{H}$ solves the magnetostatic Maxwell equations in the entire space. For a given FE magnetization $\mathbf{m}_h$, the naive computation of $\mathbf{H}$ via a closed formula typically leads to dense matrices and quadratic complexity with respect to the number N of elements. To reduce the computational cost, it is proposed to apply $\mathcal{H}$-matrix techniques instead. This approach allows for the computation (and evaluation) of $\mathbf{H}$ in linear complexity even on adaptively generated (or unstructured) meshes.

1 Basic Micromagnetics

Let $\Omega \subset \mathbb{R}^d$ be the bounded spatial domain of a ferromagnet. Then, the magnetization $\mathbf{m} : \Omega \to \mathbb{R}^d$ induces the so-called stray-field [9] (or demagnetization field) $\mathbf{H} : \mathbb{R}^d \to \mathbb{R}^d$, which is the solution of the magnetostatic Maxwell equations

$$\operatorname{curl} \mathbf{H} = 0 \quad \text{and} \quad \operatorname{div} \mathbf{B} = 0 \quad \text{on } \mathbb{R}^d. \tag{1}$$

Here, $\mathbf{B} = \mathbf{H} + \mathbf{m}$ denotes the magnetic induction, with $\mathbf{m}$ extended by zero to $\mathbb{R}^d \backslash \Omega$. Stokes' Theorem implies $\mathbf{H} = -\nabla u$, with a potential u that solves

$$\operatorname{div}(-\nabla u + \mathbf{m}) = 0 \quad \text{in } \mathcal{D}'(\mathbb{R}^d). \tag{2}$$

Thus, there holds

$$u = \sum_{j=1}^{d} \frac{\partial G}{\partial x_j} * \mathbf{m}_j \quad \text{for any } \mathbf{m} \in L^2(\Omega)^d, \tag{3}$$

where G is the Newtonian kernel defined by $G(x) = (2\pi)^{-1} \log |x|$ for $d = 2$ and by $G(x) = -(4\pi)^{-1} |x|^{-1}$ for $d = 3$, respectively. Therefore, the components of the stray-field can be written as convolutions in the sense of Calderón-Zygmund. The operator $\mathcal{P} : L^2(\Omega)^d \to L^2(\Omega)^d$ mapping $\mathbf{m}$ onto the corresponding stray-field $\mathbf{H}$ is an orthogonal projection. Details and the precise mathematical setting can be found in [12].

The remainder of this paper is organized as follows: Section 2 introduces the stiffness matrix $\mathbf{A}$ arising from the FE discretization of (2). Section 3 recalls the definition of $\mathcal{H}^2$-matrices and indicates how $\mathbf{A}$ can be approximated by a $d \times d$ block matrix $\mathbf{A}_{\mathcal{H}}$ consisting of $\mathcal{H}^2$-matrix type blocks. Sections 4 and 5 contain our main results: In Section 4, we prove that $\mathbf{A}_{\mathcal{H}}$ can in fact be interpreted as a global $\mathcal{H}^2$-matrix approximation for $\mathbf{A}$. Section 5 provides an a priori analysis of the corresponding approximation error for a quite general class of FE discretizations. Some numerical experiments in Section 6 conclude the work.

2 Stray-Field Discretization

In FE simulations of micromagnetic phenomena, one usually restricts oneself to a finite dimensional subspace $\mathcal{S}_h$ of $L^2(\Omega)$. Fix a basis $\{\phi_j\}_{j=1}^N$ of $\mathcal{S}_h$. Then, the functions $\Phi_{[j,\alpha]} = \phi_j \mathbf{e}_\alpha$, with $\mathbf{e}_\alpha \in \mathbb{R}^d$ the α-th standard unit vector, define a basis of $\mathcal{S}_h^d \subset L^2(\Omega)^d$. To fix a numbering of these basis functions, we set $[j, \alpha] = j + (\alpha - 1)N$ for $1 \leq \alpha \leq d$. Now, for an FE discretization of $\mathcal{P}$, one has to compute the corresponding stiffness matrix $\mathbf{A} \in \mathbb{R}^{dN \times dN}$ defined by

$$\mathbf{A}_{jk} = \int_{\Omega} (\mathcal{P}\Phi_j)(x)\, \Phi_k(x)\, dx. \tag{4}$$

Actually, we consider the individual blocks $\mathbf{A}^{\alpha\beta} \in \mathbb{R}^{N\times N}$ of $\mathbf{A}$ separately, where

$$\mathbf{A}^{\alpha\beta}_{jk} = \int_\Omega (\mathcal{P}\varPhi_{[j,\alpha]})(x)\, \varPhi_{[k,\beta]}(x)\, dx \quad \text{for } 1 \leq \alpha, \beta \leq d. \tag{5}$$

Lemma 1. *The matrix* $\mathbf{A}$ *is a symmetric* $d \times d$ *block matrix. Furthermore, each of the* $N \times N$ *blocks* $\mathbf{A}^{\alpha\beta}$ *of* $\mathbf{A}$ *also is symmetric.*

Proof. The symmetry of $\mathbf{A}$ is a consequence of the L^2 orthogonality of $\mathcal{P}$. The symmetry of the blocks $\mathbf{A}^{\alpha\beta}$ follows from Calderón-Zygmund theory [12, Proposition 6.1]. □

3 Blockwise $\mathcal{H}^2$-Approximation of A

When applying $\mathcal{H}$-matrix techniques to approximate $\mathbf{A}$, and hence to reduce the cost of computing $\mathbf{H}$, one possibility is to treat each block $\mathbf{A}^{\alpha\beta}$ of $\mathbf{A}$ individually, as is done in [10]. To that end, one requires a classical integral representation of the associated far field.

Lemma 2. *Given a basis function* ϕ_j *of* $\mathcal{S}_h$*, let* $\operatorname{supp}(\phi_j)$ *denote its support. If* $\operatorname{supp}(\phi_j) \cap \operatorname{supp}(\phi_k) = \emptyset$ *for* $1 \leq j, k \leq N$*, then*

$$\mathbf{A}^{\alpha\beta}_{jk} = \int_\Omega \int_\Omega \partial_{\alpha\beta} G(x-y)\phi_j(y)\phi_k(x)\, dy\, dx, \tag{6}$$

with $\partial_{\alpha\beta}G$ *the second derivative of* G*.*

Proof. On $\operatorname{supp}(\phi_k)$, there holds $\mathcal{P}(\phi_j \mathbf{e}_\alpha) \cdot (\phi_k \mathbf{e}_\beta) = \partial_\beta(\partial_\alpha G * \phi_j)\phi_k$. By classical convolution results, we have $\partial_\alpha G * \phi_j \in \mathcal{C}^1(\mathbb{R}^d \backslash \operatorname{supp}(\phi_j))$ and $\partial_\beta(\partial_\alpha G * \phi_j) = \partial_{\alpha\beta} G * \phi_j$. This concludes the proof. □

The kernel functions $\kappa^{\alpha\beta}(x,y) := \partial_{\alpha\beta}G(x-y)$ appearing in (6) are asymptotically smooth. Therefore, each $\mathbf{A}^{\alpha\beta}$ can be approximated by an $\mathcal{H}^2$-matrix obtained from tensorial interpolation of $\kappa^{\alpha\beta}$ [3].

Let $\mathcal{I} = \{1, \dots, N\}$ denote the index set corresponding to the basis $\{\phi_j\}_{j=1}^N$. Given a cluster $\sigma \subseteq \mathcal{I}$, let $\cup\sigma := \bigcup\{\operatorname{supp}(\phi_j) : j \in \sigma\}$, and let $B_\sigma = \prod_{\ell=1}^d [a_\ell, b_\ell] \subset \mathbb{R}^d$ denote the box of minimal size containing $\cup\sigma$. For $j \in \sigma$, fix an element $x_j \in \operatorname{supp}(\phi_j)$ (e.g., the center of mass).

From $\mathcal{I}$, we build a so-called cluster tree $\mathbb{T}$ by binary space partitioning [1], calling the function `clustertree` via `clustertree`$(\mathcal{I}, \emptyset)$:

function clustertree(σ, var $\mathbb{T}$)

```
if $|\sigma| \le 1$
   return
else
   split $B_\sigma$ along longest edge into boxes $B_\sigma^1, B_\sigma^2$ of equal volume
   if $\sigma_1 := \{j \in \mathcal{I} : x_j \in B_\sigma^1\} \notin \{\emptyset, \sigma\}$
      add $\sigma_1, \sigma_2 := \sigma \backslash \sigma_1$ to $\mathbb{T}$
      call clustertree($\sigma_1$, var $\mathbb{T}$)
      call clustertree($\sigma_2$, var $\mathbb{T}$)
   end
end
```

Having constructed $\mathbb{T}$, we generate a block partitioning $\mathbb{P}$ for $\mathcal{I} \times \mathcal{I}$ as follows: For a fixed parameter $\eta > 0$, we call $(\sigma, \tau) \in \mathbb{T} \times \mathbb{T}$ an *admissible (far field) block* if

$$\operatorname{diam}(B_\sigma \times B_\tau) \le \eta \operatorname{dist}(B_\sigma, B_\tau). \tag{7}$$

Otherwise, (σ, τ) is an *inadmissible (near field) block*. Now, the following recursive function, called by partition($\mathcal{I}, \mathcal{I}, \mathbb{T}$, var $\mathbb{P}$), partitions $\mathcal{I} \times \mathcal{I}$ into admissible blocks $(\sigma, \tau) \in \mathbb{P}_{\text{far}}$ and inadmissible blocks $(\sigma, \tau) \in \mathbb{P}_{\text{near}}$; clearly, $\mathbb{P} = \mathbb{P}_{\text{far}} \cup \mathbb{P}_{\text{near}}$. Here, $\operatorname{sons}(\sigma)$ denotes the set of all sons of $\sigma \in \mathbb{T}$ with respect to $\mathbb{T}$.

function partition($\sigma, \tau, \mathbb{T}$, var $\mathbb{P}$)

```
if $(\sigma, \tau)$ admissible
   add $(\sigma, \tau)$ to $\mathbb{P}_{\text{far}}$
elseif $\operatorname{sons}(\sigma) \neq \emptyset$
   if $\operatorname{sons}(\tau) \neq \emptyset$
      for all $(\sigma', \tau') \in \operatorname{sons}(\sigma) \times \operatorname{sons}(\tau)$ call partition($\sigma', \tau', \mathbb{T}$, var $\mathbb{P}$)
   else
      for all $\sigma' \in \operatorname{sons}(\sigma)$ call partition($\sigma', \tau, \mathbb{T}$, var $\mathbb{P}$)
   end
elseif $\operatorname{sons}(\tau) \neq \emptyset$
   for all $\tau' \in \operatorname{sons}(\tau)$ call partition($\sigma, \tau', \mathbb{T}$, var $\mathbb{P}$)
else
   add $(\sigma, \tau)$ to $\mathbb{P}_{\text{near}}$
end
```

Note that for $(\sigma, \tau) \in \mathbb{P}_{\text{far}}$, Lemma 2 applies. Since $\kappa^{\alpha\beta}$ is smooth on $B_\sigma \times B_\tau$, we may replace it by its tensorial Čebyšev interpolation,

$$\kappa^{\alpha\beta}(x, y) \approx \kappa_{\sigma\tau}^{\alpha\beta}(x, y) := \sum_{m,n=1}^{p^d} \kappa^{\alpha\beta}(x_m^\sigma, x_n^\tau) L_m^\sigma(x) L_n^\tau(y). \tag{8}$$

Here, L_m^σ and L_n^τ are tensorial Lagrange polynomials of overall degree p^d (i.e. of degree p in each of the coordinate directions), with corresponding interpolation nodes $x_m^\sigma \in B_\sigma$ and $x_n^\tau \in B_\tau$, respectively. This leads to an approximation of $\mathbf{A}^{\alpha\beta}$ via

$$\mathbf{A}^{\alpha\beta}|_{\sigma\times\tau} \approx \mathbf{A}_{\mathcal{H}}^{\alpha\beta}|_{\sigma\times\tau} := V_\sigma M_{\sigma\tau}^{\alpha\beta} V_\tau^T \quad \text{for } (\sigma,\tau) \in \mathbb{P}_{\text{far}}, \tag{9}$$

where $(V_\sigma)_{jm} = \int_\Omega \phi_j L_m^\sigma \, dx$ and $(M_{\sigma\tau}^{\alpha\beta})_{mn} = \kappa^{\alpha\beta}(x_m^\sigma, x_n^\tau)$. Moreover, there holds the additional hierarchy

$$V_\sigma|_{\sigma'} = V_{\sigma'} T_{\sigma'\sigma} \quad \text{for } \sigma' \in \text{sons}(\sigma),$$

with a transfer matrix T given by $(T_{\sigma'\sigma})_{mn} = L_n^\sigma(x_m^{\sigma'})$. The following complexity estimate is a standard result from $\mathcal{H}^2$-matrix theory [1].

Theorem 1. *Given $\mathbb{T}$ and $\mathbb{P}$ as constructed above, define the sparsity constant $\mathrm{C}_{\text{sp}} = \max_{\sigma\in\mathbb{T}} \#\{\tau \in \mathbb{T} : (\sigma,\tau) \in \mathbb{P}\}$. Assume that for $(\sigma,\tau) \in \mathbb{P}_{\text{near}}$ and $(j,k) \in \sigma \times \tau$, each entry $\mathbf{A}_{jk}^{\alpha\beta}$ can be computed with complexity $\mathcal{O}(1)$. Then, for $\mathbf{A}_{\mathcal{H}}^{\alpha\beta}$, the assembly, storage, and matrix-vector multiplication can be performed with complexity $\mathcal{O}(\mathrm{C}_{\text{sp}} p^{2d} N)$.* □

4 Global $\mathcal{H}^2$-Approximation of A

We now define an approximation $\mathbf{A}_{\mathcal{H}}$ for the stiffness matrix $\mathbf{A}$ by replacing all blocks $\mathbf{A}^{\alpha\beta}$ by their $\mathcal{H}^2$-matrix approximants $\mathbf{A}_{\mathcal{H}}^{\alpha\beta}$. In fact, we show that $\mathbf{A}_{\mathcal{H}}$ can be interpreted as a global $\mathcal{H}^2$-approximation for $\mathbf{A}$ if one takes into account the following considerations:

- In contrast to the previous section, we now consider the index set $\widehat{\mathcal{I}} = \{1,\ldots,dN\}$. Given the cluster tree $\mathbb{T}$ built from $\mathcal{I} = \{1,\ldots,N\}$, we make d copies $\mathbb{T}_\alpha$ of $\mathbb{T}$ which correspond to the indices $\mathcal{I}_\alpha = \{[1,\alpha],\ldots,[N,\alpha]\}$. This gives us a cluster tree $\widehat{\mathbb{T}}$ for $\widehat{\mathcal{I}}$. The root of $\widehat{\mathbb{T}}$ has precisely the d son branches $\mathbb{T}_\alpha$.
- We use the same admissibility condition (7) as before, but replace ϕ_j in the definition of $\cup\sigma$ by Φ_j.
- Finally, when $\sigma = \widehat{\mathcal{I}}$ and $\sigma' = \mathcal{I}_\alpha$, the transfer matrices $T_{\sigma'\sigma}$ are just the identities.

Theorem 2. *The partitioning $\widehat{\mathbb{P}}$ induced by $\widehat{\mathbb{T}}$ and the modified admissibility condition coincides blockwise with the partitioning $\mathbb{P}$ for $\mathbf{A}_{\mathcal{H}}^{\alpha\beta}$. Therefore, and with the above definition of the additional transfer matrices needed, $\mathbf{A}_{\mathcal{H}}$ is an $\mathcal{H}^2$-matrix.* □

This important result allows us to apply any algorithm from $\mathcal{H}^2$-matrix theory to $\mathbf{A}$ as a whole. In particular, this concerns algorithms for the preconditioning or recompression of $\mathcal{H}^2$-matrices or the assembly of $\mathbf{A}$ by use of adaptive cross approximation [11]. So far, we were only able to apply the respective algorithms blockwise, i.e. to each $\mathbf{A}^{\alpha\beta}$ individually [10].

5 Approximation Error Estimate

It remains to study the approximation error which results from replacing $\mathbf{A}^{\alpha\beta}$ by $\mathbf{A}_{\mathcal{H}}^{\alpha\beta}$. For $\kappa^{\alpha\beta}(x,y) = \partial_{\alpha\beta}G(x-y)$ and $\kappa_{\sigma\tau}^{\alpha\beta}$ as defined in (8), there holds

$$\|\kappa^{\alpha\beta} - \kappa_{\sigma\tau}^{\alpha\beta}\|_{\infty,B_\sigma\times B_\tau} \leq C\,\Lambda(p)\,(1+2/\eta)^{-p},$$

with $\eta > 0$ as in (7). Here, $\Lambda(p)$ grows logarithmically with p, and C is a numerical constant which depends only linearly on $\operatorname{dist}(B_\sigma, B_\tau)^{-d}$, cf. [2, 10]. In particular, the error decreases exponentially with the approximation order p. Now, given $C_{\mathcal{H}} > 0$, choose p large enough such that

$$\|\kappa^{\alpha\beta} - \kappa_{\sigma\tau}^{\alpha\beta}\|_{\infty,B_\sigma\times B_\tau} \leq C_{\mathcal{H}} \quad \text{for all } (\sigma,\tau) \in \mathbb{P}_{\text{far}}. \tag{10}$$

As a first direct consequence, we obtain

Theorem 3. *The matrix error in the Frobenius norm satisfies*

$$\|\mathbf{A}^{\alpha\beta} - \mathbf{A}_{\mathcal{H}}^{\alpha\beta}\|_F \leq C_{\mathcal{H}} N \max_{j=1,\dots,N} \|\phi_j\|_{L^1}^2. \qquad \square$$

More interesting than the matrix error, however, is the error for the corresponding bilinear forms: For a discrete magnetization $\mathbf{m}_h \in \mathcal{S}_h^d$, let $\widehat{\mathbf{m}_h} \in \mathbb{R}^{dN}$ denote the coefficient vector with respect to the basis functions $\Phi_{[j,\alpha]}$. Then, replacing $\mathbf{A}$ by $\mathbf{A}_{\mathcal{H}}$ corresponds to replacing the bilinear form $a(\mathbf{m}_h, \mathbf{n}_h) = \widehat{\mathbf{m}_h} \cdot \mathbf{A}\widehat{\mathbf{n}_h}$ by

$$a_{\mathcal{H}}(\mathbf{m}_h, \mathbf{n}_h) := \widehat{\mathbf{m}_h} \cdot \mathbf{A}_{\mathcal{H}}\widehat{\mathbf{n}_h} \quad \text{for } \mathbf{m}_h, \mathbf{n}_h \in \mathcal{S}_h^d.$$

The error analysis for these bilinear forms requires some additional assumptions on the basis functions ϕ_j: First, assume that

$$\sum_{j\in\sigma} |\mathrm{supp}\,(\phi_j)| \le C_{\mathrm{loc}} \Big| \bigcup_{j\in\sigma} \mathrm{supp}\,(\phi_j) \Big| \tag{11}$$

for any $\sigma \subseteq \{1,\dots,N\}$, with some $C_{\mathrm{loc}} > 0$. Moreover, let $C_{\mathrm{stab}} > 0$ be a constant such that for any coefficient vector $x \in \mathbb{R}^N$,

$$C_{\mathrm{stab}}^{-1} \Big\| \sum_{j=1}^N x_j\phi_j \Big\|_{L^2}^2 \le \sum_{j=1}^N \|x_j\phi_j\|_{L^2}^2 \le C_{\mathrm{stab}} \Big\| \sum_{j=1}^N x_j\phi_j \Big\|_{L^2}^2. \tag{12}$$

Note that assumptions (11) and (12) are quite natural. For a triangular mesh and the corresponding P^1 hat functions, (11) is essentially an assumption on the angles in the triangulation. Moreover, the usual FE bases satisfy (12) even for a quite general class of meshes [7]. In particular, both (11) and (12) are clearly satisfied for piecewise constant basis functions, with $C_{\mathrm{loc}} = 1 = C_{\mathrm{stab}}$.

The following theorem is our second main result:

Theorem 4. *Under the above assumptions, there holds*

$$|a(\mathbf{m}_h,\mathbf{n}_h) - a_{\mathcal{H}}(\mathbf{m}_h,\mathbf{n}_h)| \le C_{\mathcal{H}} C_{\mathrm{loc}} C_{\mathrm{stab}}^2 |\Omega| d^2 \, \|\mathbf{m}_h\|_{L^2} \|\mathbf{n}_h\|_{L^2} \tag{13}$$

for all $\mathbf{m}_h, \mathbf{n}_h \in \mathcal{S}_h^d$.

Proof. For $\mathbf{m} \in \mathcal{S}_h^d$ and $\sigma \subseteq \mathcal{I}$, we write $\mathbf{m}_\sigma := \sum_{\ell\in\sigma} \widehat{\mathbf{m}}_\ell \Phi_\ell$. With this notation, the error $e = a - a_{\mathcal{H}}$ for the bilinear forms reads

$$e(\mathbf{m},\mathbf{n}) = \sum_{(\sigma,\tau)\in\mathbb{P}_{\mathrm{far}}} \int_{\cup\sigma}\int_{\cup\tau} \mathbf{m}_\sigma(x) \big(\kappa^{\alpha\beta} - \kappa^{\alpha\beta}_{\sigma\tau}\big)(x,y)\, \mathbf{n}_\tau(y)\, dy\, dx.$$

From (10) and the Hölder and Cauchy Inequalities, one obtains

$$\begin{aligned} |e(\mathbf{m},\mathbf{n})| &\le C_{\mathcal{H}} \sum_{(\sigma,\tau)\in\mathbb{P}_{\mathrm{far}}} |\cup\sigma|^{1/2} |\cup\tau|^{1/2} \|\mathbf{m}_\sigma\|_{L^2} \|\mathbf{n}_\tau\|_{L^2} \\ &\le C_{\mathcal{H}} \Big(\sum_{(\sigma,\tau)\in\mathbb{P}_{\mathrm{far}}} |\cup\sigma||\cup\tau| \Big)^{1/2} \Big(\sum_{(\sigma,\tau)\in\mathbb{P}_{\mathrm{far}}} \|\mathbf{m}_\sigma\|_{L^2}^2 \|\mathbf{n}_\tau\|_{L^2}^2 \Big)^{1/2}. \end{aligned}$$

Finally, a direct calculation shows that these sums can be dominated by $d^2 C_{\mathrm{loc}}^2 |\Omega|^2$ and $d^2 C_{\mathrm{stab}}^4 \|\mathbf{m}\|_{L^2}^2 \|\mathbf{n}\|_{L^2}^2$, respectively. □

N / p	2	3	4	5	6
320	2.1	2.1	2.1	2.7	2.7
1280	4.3	4.3	4.3	6.0	6.1
5120	5.8	5.8	6.1	11.2	11.2
20480	6.7	6.7	6.8	13.8	13.9
81920	7.0	7.0	7.0	15.2	15.3

N / p	2	3	4	5	6
320	0.1	0.1	0.1	0.1	0.2
1280	0.1	0.1	0.2	0.2	0.3
5120	0.1	0.2	0.3	0.4	0.6
20480	0.2	0.2	0.4	0.6	1.0
81920	0.2	0.2	0.5	0.7	1.2

Table 1. Assembly times (left) and recompression times (right) for $\mathbf{A}_{\mathcal{H}}$ ([ms/N], uniform mesh-refinement).

N / p	2	3	4	5	6	full
320	5.9	6.4	7.7	9.5	11.7	10.0
1280	12.3	13.5	16.8	20.9	25.4	40.0
5120	17.8	21.1	30.0	40.5	50.9	160.0
20480	21.0	26.0	39.5	56.9	76.4	640.0
81920	22.8	28.9	45.2	67.0	92.9	2560.0

N / p	2	3	4	5	6
320	5.8	5.9	6.0	7.4	7.5
1280	12.1	12.3	12.8	17.0	17.3
5120	17.4	18.4	20.7	32.4	33.1
20480	20.4	21.7	24.3	41.9	43.4
81920	22.1	23.6	27.4	47.3	49.3

Table 2. Storage requirements of $\mathbf{A}_{\mathcal{H}}$ (left) and of $\mathbf{A}_{\mathcal{H}}^{\mathrm{rcp}}$ (right) ([kB/N], uniform mesh-refinement). For comparison, we give the values for the full matrix $\mathbf{A}$.

6 Numerical Experiments

To underline our theoretical results, we performed numerical experiments for the Landau-Lifshitz minimization problem in the large-body limit [8]. We discretized the corresponding Euler-Lagrange equations by a Galerkin method with $\mathcal{T}$-piecewise constant ansatz and test functions, where $\mathcal{T}$ is a triangulation of $\Omega \subset \mathbb{R}^2$ by rectangular elements which admits hanging nodes. More specifically, we considered a ferromagnetic rod $\Omega = (-0.5, 0.5) \times (-2.5, 2.5)$, with the uniform initial mesh consisting of $N = 20$ squares. For the corresponding numerical analysis and an effective implementation, the reader is referred to [5, 6].

N / p	2	3	4	5	6
320	2.7_{-5}	2.9_{-6}	3.3_{-7}	3.5_{-8}	4.0_{-9}
1280	1.5_{-5}	1.6_{-6}	1.7_{-7}	1.8_{-8}	2.1_{-9}
5120	5.5_{-6}	5.8_{-7}	6.0_{-8}	6.1_{-9}	6.7_{-10}
20480	1.7_{-6}	1.8_{-7}	1.8_{-8}	1.8_{-9}	2.1_{-10}

N / p	2	3	4	5	6
320	3.7_{-5}	3.6_{-6}	2.9_{-7}	2.3_{-8}	2.0_{-9}
1280	2.5_{-5}	2.0_{-6}	2.6_{-7}	2.0_{-8}	1.9_{-9}
5120	8.4_{-6}	9.2_{-7}	1.3_{-7}	8.0_{-9}	6.7_{-10}
20480	2.4_{-6}	3.0_{-7}	4.2_{-8}	2.2_{-9}	2.7_{-10}

Table 3. Errors $\|\mathbf{A} - \mathbf{A}_{\mathcal{H}}\|_2$ (left) and $\|\mathbf{A} - \mathbf{A}_{\mathcal{H}}^{\mathrm{rcp}}\|_2$ (right) (uniform mesh-refinement).

N / p	**2**	**3**	**4**	**5**	**6**
308	2.0	2.0	2.1	2.3	2.4
1244	4.4	4.4	4.4	6.2	6.2
6548	5.0	5.1	8.5	9.2	13.5
26204	6.1	6.1	10.9	12.6	18.4
117524	7.0	7.2	13.3	13.4	22.5

N / p	**2**	**3**	**4**	**5**	**6**
308	0.1	0.1	0.1	0.1	0.2
1244	0.1	0.1	0.1	0.2	0.3
6548	0.2	0.2	0.3	0.5	0.6
26204	0.2	0.3	0.3	0.8	0.9
117524	0.2	0.3	0.5	0.9	1.3

Table 4. Assembly times (left) and recompression times (right) for $\mathbf{A}_{\mathcal{H}}$ ([ms/N], adaptive mesh-refinement).

N / p	**2**	**3**	**4**	**5**	**6**	**full**
308	5.7	6.1	7.3	9.0	11.5	9.6
1244	12.7	13.9	17.3	21.0	25.0	38.9
6548	18.0	23.8	34.2	46.5	54.9	204.6
26204	21.8	30.7	46.6	66.0	80.4	818.9
117524	24.9	33.9	54.0	76.8	101.3	3672.6

N / p	**2**	**3**	**4**	**5**	**6**
308	5.5	5.6	5.6	6.5	6.6
1244	12.3	12.6	12.8	17.3	17.5
6548	16.8	18.0	26.5	29.1	37.9
26204	20.8	30.7	35.8	40.0	53.1
117524	23.6	25.9	41.2	43.8	65.6

Table 5. Storage requirements of $\mathbf{A}_{\mathcal{H}}$ (left) and of $\mathbf{A}_{\mathcal{H}}^{\mathrm{rcp}}$ (right) ([kB/N], adaptive mesh-refinement). For comparison, we give the values for the full matrix $\mathbf{A}$.

The experiments were conducted using the HLib software package by S. Börm and L. Grasedyck of the Max-Planck-Institute for Mathematics in the Sciences (Leipzig, Germany). We utilized a Compaq/HP AlphaServer ES45 under Unix, with 32 GB of RAM and four Alpha EV68 CPUs running at 1 GHz each. Implementational details can be found in [10, 11]; in particular, the $\mathcal{H}^2$-factorization of a block $\mathbf{A}^{\alpha\beta}|_{\sigma\times\tau}$ was stored only if this was cheaper than storing the exact matrix block.

Tables 1–3 contain experimental results (assembly times, storage requirements, and the error $\|\mathbf{A} - \mathbf{A}_{\mathcal{H}}\|_2$ computed by a power iteration) for uniform mesh-refinement. Moreover, we compared $\mathbf{A}_{\mathcal{H}}$ to the matrix $\mathbf{A}_{\mathcal{H}}^{\mathrm{rcp}}$ obtained by adaptive $\mathcal{H}^2$-recompression [4] of $\mathbf{A}_{\mathcal{H}}$.

N / p	**2**	**3**	**4**	**5**	**6**
308	3.9_{-5}	3.9_{-6}	6.1_{-7}	3.9_{-8}	1.9_{-9}
1244	2.5_{-5}	2.8_{-6}	2.9_{-7}	2.7_{-8}	4.1_{-9}
6548	1.4_{-5}	1.6_{-6}	1.5_{-7}	1.6_{-8}	2.5_{-9}
26204	5.1_{-6}	5.8_{-7}	5.5_{-8}	6.8_{-9}	9.7_{-10}

N / p	**2**	**3**	**4**	**5**	**6**
308	1.6_{-4}	1.3_{-5}	2.4_{-6}	1.0_{-7}	4.6_{-9}
1244	1.6_{-4}	7.5_{-6}	1.1_{-6}	4.7_{-8}	8.6_{-9}
6548	3.1_{-5}	2.6_{-6}	2.1_{-7}	2.6_{-8}	3.2_{-9}
26204	6.8_{-6}	6.4_{-7}	5.7_{-8}	6.8_{-9}	9.7_{-10}

Table 6. Errors $\|\mathbf{A} - \mathbf{A}_{\mathcal{H}}\|_2$ (left) and $\|\mathbf{A} - \mathbf{A}_{\mathcal{H}}^{\mathrm{rcp}}\|_2$ (right) (adaptive mesh-refinement).

Note that $\mathbf{A}_{\mathcal{H}}^{\mathrm{rcp}}$ provides almost the same accuracy as $\mathbf{A}_{\mathcal{H}}$, but typically requires only 70% of the storage, and that the recompression times are negligible in comparison to the respective assembly times.

Finally, in Tables 4–6, we give the corresponding results for a sequence of adaptively generated meshes. Here, $\mathcal{T}^{(j+1)}$ is obtained from $\mathcal{T}^{(j)}$ as follows: First, $\mathcal{T}^{(j)}$ is refined uniformly. Then, in a second step, we additionally refine either the four corner elements of Ω (for j even) or all the elements along the edges of Ω (for j odd). Meshes of this type are observed in [6] for an adaptive mesh-refining strategy based on a residual a posteriori error estimate.

References

1. Börm, S., Grasedyck, L., Hackbusch, W.: Introduction to Hierarchical Matrices with Applications, Eng. Anal. with B. Elem. **27** (2003) 405–422.
2. Börm, S., Löhndorf, M., Melenk, M.: Approximation of Integral Operators by Variable-Order Interpolation, Numer. Math., in press (2004).
3. Börm, S., Hackbusch, W.: $\mathcal{H}^2$-Matrix Approximation of Integral Operators by Interpolation, Appl. Numer. Math. **43** (2002) 129–143.
4. Börm, S., Hackbusch, W.: Data-Sparse Approximation by Adaptive $\mathcal{H}^2$-Matrices, Computing **69** (2002) 1–35.
5. Carstensen, C., Praetorius, D.: Effective Simulation of a Macroscopic Model for Stationary Micromagnetics, Comput. Methods Appl. Mech. Engrg. **194** (2005) 531–548.
6. Carstensen, C., Praetorius, D.: Numerical Analysis for a Macroscopic Model in Micromagnetics, SIAM J. Numer. Anal., in press (2004).
7. Dahmen, W., Faermann, B., Graham, I., Hackbusch, W., Sauter, S.: Inverse Inequalities on Non-Quasi-Uniform Meshes and Application to the Mortar Finite Element Method, Math. Comp. **73** (2004) 1107–1138.
8. DeSimone, A.: Energy Minimizers for Large Ferromagnetic Bodies, Arch. Rational Mech. Anal. **125** (1993) 99–143.
9. Hubert, A., Schäfer, R.: Magnetic Domains, Springer-Verlag, Berlin (1998).
10. Popović, N., Praetorius, D.: Applications of $\mathcal{H}$-Matrix Techniques in Micromagnetics, Computing, in press (2004).
11. Popović, N., Praetorius, D.: The $\mathcal{H}^2$-ACA Algorithm for the Efficient Treatment of the Magnetostatic Potential, in preparation (2005).
12. Praetorius, D.: Analysis of the Operator $\Delta^{-1}\mathrm{div}$ Arising in Magnetic Models, Z. Anal. Anwendungen **23** (2004) 589–605.