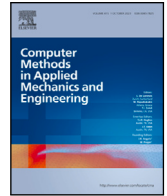




Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

Tree–cotree-based tearing and interconnecting for 3D magnetostatics: A dual–primal approach

Mario Mally ^a*, Bernard Kapidani ^b, Melina Merkel ^a, Sebastian Schöps ^a, Rafael Vázquez ^{c,d}

^a Computational Electromagnetics Group, Technische Universität Darmstadt, 64289 Darmstadt, Germany

^b Chair of Numerical Modelling and Simulation, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

^c Department of Applied Mathematics, Universidade de Santiago de Compostela, 15782, Santiago de Compostela, Spain

^d Galician Centre for Mathematical Research and Technology (CITMaga), 15782, Santiago de Compostela, Spain

ARTICLE INFO

Keywords:

IGA
IETI
Dual–primal
Tree–cotree gauging
Magnetostatics

ABSTRACT

The simulation of electromagnetic devices with complex geometries and large-scale discrete systems benefits from advanced computational methods like IsoGeometric Analysis and Domain Decomposition. In this paper, we employ both concepts in an Isogeometric Tearing and Interconnecting method to enable the use of parallel computations for magnetostatic problems. We address the underlying non-uniqueness by using a graph-theoretic approach, the tree–cotree decomposition. The classical tree–cotree gauging is adapted to be feasible for parallelization, which requires that all local subsystems are uniquely solvable. Our contribution consists of an explicit algorithm for constructing compatible trees and combining it with a dual–primal approach to enable parallelization. The correctness of the proposed approach is proved and verified by numerical experiments, showing its accuracy, scalability and optimal convergence.

1. Introduction

Simulation-based design workflows for engineering products are ubiquitous in academia and industry. Their success relies on accurate and efficient solvers. A suitable method to achieve this is IsoGeometric Analysis (IGA), due to its ability to accurately represent complex geometries and to provide higher continuity in the solution space. This is achieved by using NURBS (Non-Uniform Rational B-Splines) to represent the geometry and NURBS or B-Splines to approximate the solution [1]. These advantages are particularly attractive for the magnetostatic simulation of permanent magnet synchronous machines due to their cylindrical shape [2] and the computation of sensitive quantities, such as torque [3]. This is made possible by the introduction of spline spaces suitable for the representation of electromagnetic fields [4–6].

However, as the complexity and size of simulation models increase, it becomes essential to exploit parallel computing architectures to effectively handle the computational load. We follow the Domain Decomposition (DD) paradigm [7] and in particular the Tearing and Interconnecting (TI) methods originally proposed by Farhat and Roux in [8]. These methods are probably best known as Finite Element Tearing and Interconnecting (FETI), while the IGA counterpart is called Isogeometric Tearing and Interconnecting (IETI) [9,10]. Both decompose the computational domain into subdomains, couple them weakly at conformal interfaces using Lagrange multipliers, and thus promise scalability for large problems. However, solvability is a fundamental issue of decomposed

* Corresponding author.

E-mail addresses: mario.mally@tu-darmstadt.de (M. Mally), bernard.kapidani@epfl.ch (B. Kapidani), melina.merkel@tu-darmstadt.de (M. Merkel), sebastian.schoep@tu-darmstadt.de (S. Schöps), rafael.vazquez@usc.es (R. Vázquez).

<https://doi.org/10.1016/j.cma.2025.117737>

Received 31 July 2024; Received in revised form 23 December 2024; Accepted 7 January 2025

Available online 21 January 2025

0045-7825/© 2025 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Laplace-type problems due to the lack of essential boundary conditions for ‘floating’ subdomains. Two main approaches address this problem: The dual–primal approach, see e.g. [9,11,12] or the total/all-floating approach [13–15].

For high-frequency electromagnetic problems, FETI has been investigated for scattering and wave propagation, as discussed in [16] and references therein. In the case of magnetism, it was applied to the scalar potential formulation, i.e. a Laplace-type problem, in [17–19]. The vector potential formulation suffers from non-uniqueness issues due to the kernel of the curl operator. This formulation is considered in [20–23]. The first two contributions regularize the problem by perturbing the operator. We follow and extend the idea of Yao et al. in [22,23] and employ a tree–cotree decomposition [24–27].

Let us state our contributions. We provide a mathematical justification of the approach from Yao et al. Their work only considers problems with full Dirichlet boundary conditions and low-order finite elements. We give a new explicit algorithm for constructing a tree–cotree decomposition that also applies to problems with mixed boundary conditions and high-order isogeometric discretizations. Finally, we validate our approach with numerical experiments using the IGA-library GeoPDEs [28].

The paper is structured as follows: Section 2 introduces the relevant Maxwell equations and provides an overview of the weak FETI/IETI formulation for magnetostatics as well as its discretization. Section 3 explains the fundamental concepts of tree–cotree decompositions and presents an explicit algorithm for constructing appropriate trees. In Section 4, we examine and analyze the constructed tree within the context of dual–primal TI, detailing how to correctly select primal degrees of freedom to ensure local invertibility in general settings. Finally, in Section 5 the proposed approach is verified with numerical experiments and its scalability is examined with respect to relevant properties for large-scale problems.

2. Magnetostatics on decomposed domains

In the low-frequency regime, e.g. for the simulation of permanent magnet synchronous machines [29], the magnetostatic approximation of Maxwell’s equations [30]

$$\operatorname{curl}(\nu \mathbf{B}) = \mathbf{J} \quad (1)$$

$$\operatorname{div} \mathbf{B} = 0 \quad (2)$$

is most relevant. The system is formulated in terms of the magnetic flux density \mathbf{B} and the excitation current density \mathbf{J} . For the underlying material parameter ν (reluctivity), one typically assumes $0 < \nu_1 \leq \nu \leq \nu_2 < \infty$. We introduce a magnetic vector potential via $\mathbf{B} = \operatorname{curl} \mathbf{A}$, which automatically satisfies (2). Inserting it into (1) results in

$$\operatorname{curl}(\nu \operatorname{curl} \mathbf{A}) = \mathbf{J}. \quad (3)$$

We assume that these laws are prescribed on a bounded, simply-connected domain $\Omega \subset \mathbb{R}^3$ which we will relax only in Section 5.2 where a hollow cylinder is simulated. On the boundary $\partial\Omega$, conditions are prescribed to account for the behavior of the fields outside of the domain. The most common conditions are

$$\mathbf{A} \times \mathbf{n} = \mathbf{g}_D \quad \text{on } \Gamma_D, \quad (4)$$

$$(\nu \operatorname{curl} \mathbf{A}) \times \mathbf{n} = \mathbf{g}_N \quad \text{on } \Gamma_N, \quad (5)$$

which define (inhomogeneous) Dirichlet and Neumann boundary conditions on Γ_D and Γ_N , respectively. Here, \mathbf{n} denotes the outward normal vector of $\partial\Omega$. For simplicity of presentation, we focus on $\mathbf{g}_D = 0$ and $\mathbf{g}_N = 0$ which correspond to perfect electric and magnetic conducting boundary conditions, respectively. We assume that $\partial\Omega = \overline{\Gamma}_D \cup \overline{\Gamma}_N$ (where \overline{D} denotes the closure of open set D) and $\Gamma_D \cap \Gamma_N = \emptyset$. Furthermore we require that Γ_D is connected, which will be relaxed in the test configuration of Section 5.3.

The vector potential formulation (3)–(5) has a major intricacy originating from the non-trivial kernel of the curl-operator. To summarize the descriptions of [31], we can write

$$\ker(\operatorname{curl}) = \operatorname{im}(\operatorname{grad}) \cup \operatorname{span}(\mathbf{h}^{(k)})_{k=1}^{n_h} \quad (6)$$

and characterize the kernel with gradient fields $\operatorname{im}(\operatorname{grad})$ and a linear combination of n_h harmonic fields $\mathbf{h}^{(k)}$. The number of harmonic fields n_h depends on both the number of holes, i.e., the first Betti number, and the configuration of the boundary conditions. In the following, we assume $n_h = 0$, i.e., simply-connected domains Ω with connected Dirichlet boundary Γ_D . Nonetheless, we already introduce harmonic fields here, to be able to explain and comment on extensions to this paper. The identity (6) implies that only $\mathbf{A}_\perp \in \ker(\operatorname{curl})^\perp$ can be uniquely determined from (3)–(5). The remaining part $\mathbf{A}_0 \in \ker(\operatorname{curl})$ is undetermined and can be chosen freely, due to

$$\mathbf{B} = \operatorname{curl} \mathbf{A} = \operatorname{curl}(\mathbf{A}_\perp + \mathbf{A}_0) = \operatorname{curl} \mathbf{A}_\perp. \quad (7)$$

By fixing the kernel component \mathbf{A}_0 , which does not affect our quantity of interest \mathbf{B} , and reformulating (3), one can compute a unique \mathbf{A}_\perp if the current density is compatible, i.e. $\mathbf{J} \in \ker(\operatorname{curl})^\perp$, see e.g. [31]. This procedure is denoted as gauging in the literature, e.g. [32]. One particular example, for simply-connected domains, is the Coulomb gauge where $\operatorname{div} \mathbf{A} = \operatorname{div} \mathbf{A}_0 = 0$ is imposed as an additional constraint. In this paper, we focus on the tree–cotree gauge, which is applied in the discrete setting for the TI approach. Consequently, we need to discuss our weak formulation, its discretization, and the extension to domain decomposition before we can elaborate more on the gauging technique.

2.1. Weak formulation and its discretization

Let us state (3)–(5) in its weak form as: Find $\mathbf{A} \in H_{0,\Gamma_D}(\text{curl}; \Omega)$ such that

$$\int_{\Omega} (\nu \text{curl } \mathbf{A}) \cdot (\text{curl } \mathbf{v}) \, dV = \int_{\Omega} \mathbf{J} \cdot \mathbf{v} \, dV + \int_{\Gamma_N} \mathbf{g}_N \cdot \mathbf{v} \, dA \tag{8}$$

for all $\mathbf{v} \in H_{0,\Gamma_D}(\text{curl}; \Omega)$, where $H_{0,\Gamma_D}(\text{curl}; \Omega)$ denotes the adequate Hilbert space for vector potentials with vanishing tangential component on Γ_D [33]. We follow the typical Ritz–Galerkin procedure and introduce a finite-dimensional basis $\{\mathbf{w}_i\}_{i=1}^{n_g}$ such that $\mathbb{V} := \text{span}(\mathbf{w}_i)_{i=1}^{n_g} \subset H(\text{curl}; \Omega)$.

We approximate \mathbf{A} by a linear combination of finitely many basis functions, i.e.

$$\mathbf{A}_h = \sum_{i=1}^{n_g} a_i \mathbf{w}_i \in \mathbb{V}. \tag{9}$$

In this context, we call a_i a degree of freedom (DOF) and introduce $\tilde{\mathbf{a}}_g \in \mathbb{R}^{n_g}$ as the vector of all degrees of freedom (DOFs). Inserting the discrete field \mathbf{A}_h into (8) leads to

$$(\tilde{\mathbf{K}}_g)_{ij} := \int_{\Omega} \nu (\text{curl } \mathbf{w}_j) \cdot (\text{curl } \mathbf{w}_i) \, dV \tag{10}$$

for the entries of stiffness matrix $\tilde{\mathbf{K}}_g \in \mathbb{R}^{n_g \times n_g}$ and to

$$(\tilde{\mathbf{j}}_g)_i := \int_{\Omega} \mathbf{J} \cdot \mathbf{w}_i \, dV + \int_{\Gamma_N} \mathbf{g}_N \cdot \mathbf{w}_i \, dA \tag{11}$$

for the right hand side $\tilde{\mathbf{j}}_g \in \mathbb{R}^{n_g}$. Here, we added the subscript g to emphasize that this is the global problem. Under mild assumptions, it is possible to eliminate the Dirichlet DOFs as in [34, A.2] to obtain the linear system

$$\mathbf{K}_g \mathbf{a}_g = \mathbf{j}_g. \tag{12}$$

The entities \mathbf{K}_g , \mathbf{j}_g and \mathbf{a}_g are the appropriately modified versions of $\tilde{\mathbf{K}}_g$, $\tilde{\mathbf{j}}_g$ and $\tilde{\mathbf{a}}_g$, respectively.

Note that while we expect our findings to apply to any higher-order FEM discretization on, e.g. tetrahedral meshes [27,35], we focus on the multipatch IGA approach with tensor product, high-order, edge-element basis functions. In this framework, the domain Ω is divided into N patches $\Omega^{(i)}$. The decomposition satisfies $\bar{\Omega} = \bigcup_j \bar{\Omega}^{(i)}$ and the subdomains are mutually disjoint, i.e. $\Omega^{(j)} \cap \Omega^{(k)} = \emptyset$ for $j \neq k$. The discrete space is defined as $\mathbb{V} = \{\mathbf{u} \in H(\text{curl}; \Omega) : \mathbf{u}|_{\Omega^{(i)}} \in \mathbb{V}^{(i)}\}$. For each patch the subspace $\mathbb{V}^{(i)} = S_p^1(\Omega^{(i)})$ is a space of splines of mixed degree, and it has the form

$$S_p^1(\Omega^{(i)}) = \{\mathbf{u} \in H(\text{curl}; \Omega^{(i)}) : \mathbf{u} \circ \mathbf{F}^{(i)} = (D\mathbf{F}^{(i)})^{-T} \hat{\mathbf{u}}, \hat{\mathbf{u}} \in S_p^1(\hat{\Omega})\}, \tag{13}$$

$$S_p^1(\hat{\Omega}) = S_{p-1,p,p}(\Xi'_1, \Xi_2, \Xi_3) \times S_{p,p-1,p}(\Xi_1, \Xi'_2, \Xi_3) \times S_{p,p,p-1}(\Xi_1, \Xi_2, \Xi'_3), \tag{14}$$

where $\mathbf{F}^{(i)}$ is a regular mapping from the unit cube $\hat{\Omega}$ to the physical domain $\Omega^{(i)}$ and $D\mathbf{F}^{(i)}$ its non-singular Jacobian, p is the degree, and Ξ_k, Ξ'_k are the knot vectors in each parametric direction for degrees p and $p - 1$, respectively, see [36, Sec. 5] for details. As standard in the IGA setting, inserting knots in the knot vectors translates into mesh refinement, and we will denote the mesh size of the patches by h .

We also assume, following [36, Assumption 3.2], that the Dirichlet boundary Γ_D is a union of patch-faces, i.e., images of sides of the unit cube mapped through $\mathbf{F}^{(i)}$. This simplifies the elimination of Dirichlet DOFs, see [28, Sec. 4.1.3].

With the choice of the basis as in [36], there is a one-to-one correspondence between the basis functions \mathbf{w}_i (or the DOFs a_i) and the edges of the control mesh [26,37], which we interpret as a graph and denote by

$$\mathcal{G} = (\mathcal{N}, \mathcal{E}).$$

We write \mathcal{N} and \mathcal{E} for the sets of nodes and edges of the graph, respectively. In the lowest-order case, i.e. $p = 1$, this control mesh corresponds to the mesh of a conventional FEM discretization with lowest-order Nédélec-type hexahedral elements.

Remark 1. We note that the Dirichlet boundary condition could be directly incorporated into the spaces \mathbb{V} and $\mathbb{V}^{(i)}$. However, we are not eliminating the corresponding DOFs, because it helps to identify the basis functions with edges of the control mesh. Maintaining the functions on the Dirichlet boundary also simplifies the explanation of how to construct an appropriate spanning tree.

2.2. Nonoverlapping domain decomposition

Although not the only choice, it is natural to consider each patch $\Omega^{(i)}$ of the multipatch decomposition (14) as a separate subdomain in the TI context, see e.g. [9]. Consequently, we have N subdomains for an IGA multipatch space consisting of N patches. This decomposition automatically fulfills all usual assumptions including simply-connected subdomains since each patch is given by a regular mapping from the unit cube, see (13). Consequently, we also consider

$$\mathbb{V}^{(i)} = S_p^1(\Omega^{(i)}) = \text{span}(\mathbf{w}_j^{(i)})_{j=1}^{n_i} \subset H(\text{curl}; \Omega^{(i)})$$

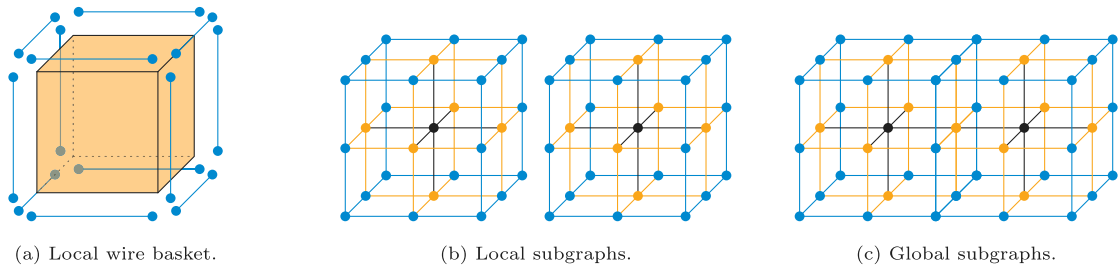


Fig. 1. Visualization of wire basket (blue) and $\partial\Omega^{(i)}$ (orange) for one hexahedral subdomain in Fig. 1(a). In Figs. 1(b) and 1(c), the corresponding subgraphs for two connected subdomains are shown. There, blue indicates $\mathcal{G}_\lambda^{(i)}$ and \mathcal{G}_λ . Furthermore, orange together with blue indicates $\mathcal{G}_\Gamma^{(i)}$ and \mathcal{G}_Γ . The inner black nodes and edges belong to neither wire basket nor boundary graphs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

for the discretization of a local magnetostatic problem on each subdomain $\Omega^{(i)}$, where $\{\mathbf{w}_j^{(i)}\}_{j=1}^{n_i}$ denote the basis functions of $\mathbb{V}^{(i)}$.

Let us introduce some more notation related to domain decomposition. Each boundary $\partial\Omega^{(i)}$ shall consist of open facets $\Gamma^{(i,k)}$ such that $\partial\Omega^{(i)} = \bigcup_k \overline{\Gamma^{(i,k)}}$ and $\Gamma^{(i,k)} \cap \Gamma^{(i,\ell)} = \emptyset$ for $k \neq \ell$. A facet is either a subset of the Dirichlet or Neumann boundary, i.e. $\Gamma^{(i,k)} \subset \Gamma_D$ or $\Gamma^{(i,k)} \subset \Gamma_N$, or an interface $\Gamma_{ij} := \Gamma^{(i,k)} = \Gamma^{(j,l)}$ for some $i < j$ and k, l . Finally, each facet boundary consists of edges $\overline{\lambda}^{(i,j,k)}$ such that $\partial\Gamma^{(i,j)} = \bigcup_{j,k} \overline{\lambda}^{(i,j,k)}$ and $\lambda^{(i,j,k)} \cap \lambda^{(i,j,\ell)} = \emptyset$ for $k \neq \ell$. The union of all edges $\overline{\lambda}^{(i,j,k)}$ (including their endpoints) defines the wire basket $\mathcal{W}^{(i)} := \bigcup_{j,k} \overline{\lambda}^{(i,j,k)}$, which is visualized in Fig. 1(a). For the local (control) meshes, we introduce graphs again. Analogous to the previous notation, $\mathcal{G}^{(i)} = (\mathcal{N}^{(i)}, \mathcal{E}^{(i)})$ denotes the (sub)graph formed by the nodes $\mathcal{N}^{(i)}$ and edges $\mathcal{E}^{(i)}$ related to the basis functions in $\mathbb{V}^{(i)}$. Furthermore, $\mathcal{G}_\Gamma^{(i)} = (\mathcal{N}_\Gamma^{(i)}, \mathcal{E}_\Gamma^{(i)})$ and $\mathcal{G}_\lambda^{(i)} = (\mathcal{N}_\lambda^{(i)}, \mathcal{E}_\lambda^{(i)})$ are the graphs for the basis functions on the boundaries and local wire baskets, respectively. Note that these subdomain-wise definitions can be extended to the global domain, i.e. \mathcal{G}_Γ and \mathcal{G}_λ , by creating unions due to the consistency of the decomposition. A visualization of these local and global subgraphs is provided in Figs. 1(b) and 1(c).

2.2.1. The local discrete problems and their coupling

Introducing discrete, local vector potentials $\mathbf{A}_h^{(i)} \in \mathbb{V}^{(i)}$ on every subdomain $\Omega^{(i)}$ with $\mathbf{A}_h = \sum_{j=1}^{n_i} a_j^{(i)} \mathbf{w}_j^{(i)}$ yields the uncoupled system of independent problems

$$\underbrace{\begin{bmatrix} \tilde{\mathbf{K}}^{(1)} & & \\ & \ddots & \\ & & \tilde{\mathbf{K}}^{(N)} \end{bmatrix}}_{=: \tilde{\mathbf{K}}} \underbrace{\begin{bmatrix} \tilde{\mathbf{a}}^{(1)} \\ \vdots \\ \tilde{\mathbf{a}}^{(N)} \end{bmatrix}}_{=: \tilde{\mathbf{a}}} = \underbrace{\begin{bmatrix} \tilde{\mathbf{j}}^{(1)} \\ \vdots \\ \tilde{\mathbf{j}}^{(N)} \end{bmatrix}}_{=: \tilde{\mathbf{j}}} \quad (15)$$

when applying the principles behind (8)–(12) to every subdomain. Note that the linear system (15) only contains local information. In order to compute solutions that also solve (12), it is essential to provide transmission conditions (also called coupling constraints), see [38], in addition to prescribing the Dirichlet conditions. Due to the conforming decomposition of Ω and of the space \mathbb{V} , the necessary coupling constraints are defined as

$$\tilde{\mathbf{B}}\tilde{\mathbf{a}} = \mathbf{0}, \quad (16)$$

where $\tilde{\mathbf{B}} \in \{-1, 0, 1\}^{m \times n}$ is a signed Boolean matrix. Here, n is given by $n = \sum_{i=1}^N n_i$ and each row of $\tilde{\mathbf{B}}$ encodes a constraint of the form

$$a_{l_j}^{(j)} - a_{i_k}^{(k)} = 0, \quad (17)$$

which enforces tangential continuity of the discrete solutions $\mathbf{A}_h^{(i)}$ on all interfaces. The arising m constraints are then incorporated by using an approach based on Lagrange multipliers which we denote here as $\tilde{\lambda} \in \mathbb{R}^m$. This results in

$$\begin{bmatrix} \mathbf{K} & \mathbf{B}^\top \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \tilde{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{j} \\ \mathbf{0} \end{bmatrix}, \quad (18)$$

where the Dirichlet boundary conditions are already imposed. The underlying elimination of Dirichlet DOFs, [34, A.2], affects the local systems again, i.e. \mathbf{K} , \mathbf{j} and \mathbf{a} are the modified versions of $\tilde{\mathbf{K}}$, $\tilde{\mathbf{j}}$ and $\tilde{\mathbf{a}}$ respectively. Additionally also the coupling constraints need to be modified. At the intersection between the Dirichlet boundary parts and interfaces, the coupling (17) between interface DOFs produces zero rows in the coupling matrix $\tilde{\mathbf{B}}$ when Dirichlet DOFs are eliminated from the system. After removing the corresponding constraints and multipliers, we write \mathbf{B} and λ instead of $\tilde{\mathbf{B}}$ and $\tilde{\lambda}$, respectively.

3. Tree–cotree gauging for tearing and interconnecting

When considering the global problem (12) typical gauging approaches, such as the tree–cotree gauging, can be straightforwardly applied to handle $\dim(\ker(\mathbf{K}_g)) > 0$, which originates from (6) and our choice of \mathbb{V} as explained in [26]. The difficulties arise if one wants to compute solutions to the local problems independently which requires gauging the matrix \mathbf{K} from (18). The reason is that

$$\dim(\ker(\mathbf{K}_g)) \leq \dim(\ker(\mathbf{K})) \quad (19)$$

holds. To gain insight into this relationship, one can draw an analogy with the Laplace problem. There, a constant c_g remains unspecified by the discrete problem with $\partial\Omega = \Gamma_N$, while every subdomain block leaves one constant c_i unspecified by itself. Due to the coupling constraints one implicitly enforces $c_g = c_1 = \dots = c_N$ which implies that every local c_i cannot be chosen freely but needs to be consistent with the global problem. Note that Dirichlet boundary conditions automatically fix $c_g = c_j$ with $\partial\Omega^{(j)} \cap \Gamma_D \neq \emptyset$ but that c_k remains unspecified in the local context for “floating” subdomains with $\partial\Omega^{(k)} \cap \Gamma_D = \emptyset$. In the curl–curl–context, we need to deal with local gradient fields (not local constants) in every $\Omega^{(i)}$ and their consistency to a global gradient field (not a global constant) instead. The underlying idea stays the same and we quantify that by relation (19).

Additional issues can arise due to edges which are shared by more than two subdomains. In analogy to the “cross-points” in nodal problems, we call them “cross-edges” and denote them as \mathcal{E}_{Π} in the global and as $\mathcal{E}_{\Pi}^{(i)}$ in the local context. These introduce redundancy in the coupling, i.e. it is not possible to uniquely determine λ from (18) because $\ker(\mathbf{B}^T) \neq \{\mathbf{0}\}$. As stated in [7, Sec. 6.3.1] or [39, Thm. 3.2] and the appended remark therein, this is not a problem if one introduces the restriction $\lambda \in \text{im}(\mathbf{B})$ and if $\ker(\mathbf{K}) \cap \ker(\mathbf{B}) = \{\mathbf{0}\}$ holds. Some configurations of cross-edges do not satisfy the latter condition while other remain solvable, but in most literature the corresponding coupling constraints are modified in some way to obtain appropriate solutions [7]. One elegant approach is the dual–primal TI variant in which selected constraints are eliminated by reducing each set of coupled DOFs in the selection to one DOF, [11,12]. In the nodal context, this approach has the additional advantage that the unspecified local constants in floating subdomains are fixed if the primal DOFs are selected appropriately.

This last additional advantage of the dual–primal ansatz for nodal problems motivates the following procedure in the curl–curl–context. First, construct an appropriate tree on \mathcal{G} that can be used to gauge the global problem. Then, tear the global tree into its local components on $\mathcal{G}^{(i)}$ and show that combining these with the dual–primal idea enables local gauging, i.e. invertibility of the local operator for each subdomain. A more detailed description of tree–cotree gauging and our choice for the global tree is given in the following part of this section. Considerations and details on the dual–primal approach with tree–cotree gauging are provided in Section 4.

3.1. Global tree–cotree decomposition for dual–primal approach

For the following considerations, we need to classify the edges and nodes of \mathcal{G}_T according to which boundary parts their respective basis functions contribute to. Starting in the local context, we write $\mathcal{N}_D^{(i)}$ and $\mathcal{E}_D^{(i)}$ for nodes and edges whose corresponding basis functions do not vanish on the Dirichlet boundary. Similarly, $\mathcal{N}_N^{(i)}$, $\mathcal{E}_N^{(i)}$ for the Neumann parts and $\mathcal{N}_I^{(i)}$, $\mathcal{E}_I^{(i)}$ for nodes and edges with relation to interfaces. Note that their intersections are not necessarily empty. The global counterparts \mathcal{N}_\star and \mathcal{E}_\star for $\star \in \{D, I, N\}$ are defined via the union of the respective local ones.

In tree–cotree gauging, one exploits the fact that it is possible to identify a decomposition of the DOFs into determined and undetermined ones from topological information of the mesh, i.e. \mathcal{G} , alone. Edges belonging to a spanning tree on \mathcal{G} correspond to undetermined DOFs while the remaining (determined) ones are called cotree DOFs, [40,41]. We write $\mathcal{E}_t \subset \mathcal{E}$ for the tree edges and $\mathcal{E}_c = \mathcal{E} \setminus \mathcal{E}_t$ for the cotree ones in the global context. The gauge then consists of prescribing the tree DOFs and eliminating them from the system. We employ Kruskal’s algorithm [42,43] with appropriate edge weights to grow such a spanning tree. The idea behind this algorithm works as follows: A not yet visited edge with the lowest weight is added to the tree if it does not close a loop. This is repeated until a stopping criterion, e.g. $\text{card}(\mathcal{E}_t) = \text{card}(\mathcal{N}) - 1$ for a connected graph \mathcal{G} , is reached. Following [43, Sec. 24.2], one can expect $\mathcal{O}(n_g \log(n_g))$ with $n_g = \text{card}(\mathcal{E})$ for the duration of Kruskal’s algorithm in an efficient implementation.

The elimination of Dirichlet boundary conditions conflicts with the elimination of tree DOFs because more DOFs than necessary are prescribed with a value. This leads to restrictions for the choice of the spanning tree. E.g. in [44], it is described that the tree subset $\mathcal{E}_{Dt} := \mathcal{E}_D \cap \mathcal{E}_t$ needs to form a spanning tree on the subgraph $\mathcal{G}_D := (\mathcal{N}_D, \mathcal{E}_D)$ to consistently gauge (12). Such a tree can be obtained by constructing it on Γ_D first, before extending it into the remaining parts of Ω , [44]. Kruskal’s algorithm preserves this hierarchy by assigning the lowest weight to edges with relation to the Dirichlet boundary and assigning higher values to the remaining edges.

Taking care of the Dirichlet contribution correctly enables an appropriate gauge of (12), but increasing the number of hierarchy levels, i.e. weights in Kruskal’s algorithm, provides good properties which we can exploit in the local context. We propose to grow the tree on the wire basket graph \mathcal{G}_λ first, before extending it into the facets, represented by \mathcal{G}_F , and after that into the volume, i.e. covering the full \mathcal{G} . This reduces complexity and allows for parallelism since the treatment of facets and volumes is automatically independent. More importantly, we can state and prove **Theorem 1** for the cotree edges on the wire basket $\mathcal{E}_{\lambda c} := \mathcal{E}_\lambda \setminus \mathcal{E}_t$.

Theorem 1. *Let \mathcal{E}_t be a set of edges which form a spanning tree on the subgraph \mathcal{G}_λ of the wire basket. Then, $\text{card}(\mathcal{E}_{\lambda c})$ is independent of the mesh size h and degree p , i.e. of $\text{card}(\mathcal{E})$, for a given decomposition of Ω .*

Proof. We can consider the decomposition of Ω into patches as a coarse mesh, or more precisely a CW-complex, where each patch corresponds to one hexahedral cell. We respectively denote by \mathcal{N}_{dec} , \mathcal{E}_{dec} , \mathcal{F}_{dec} and C_{dec} the sets of nodes, edges, faces and cells for this mesh. The corresponding graph is denoted as $\mathcal{G}_{\text{dec}} := (\mathcal{N}_{\text{dec}}, \mathcal{E}_{\text{dec}})$. Using Euler's characteristic formula [45, Thm. 2.44], we have

$$\text{card}(\mathcal{N}_{\text{dec}}) - \text{card}(\mathcal{E}_{\text{dec}}) + \text{card}(\mathcal{F}_{\text{dec}}) - \text{card}(C_{\text{dec}}) = c_1 \in \mathbb{Z}, \tag{20}$$

where the constant c_1 depends on the topology of the domain, but is independent of the decomposition into subdomains. Note that we can obtain \mathcal{G}_λ by subdividing every edge of \mathcal{G}_{dec} appropriately. Consequently, we can interpret \mathcal{G}_λ as the edges and vertices of a CW-complex of the same domain, with the same faces \mathcal{F}_{dec} and cells C_{dec} , and the same topological invariant c_1 . Furthermore, we know that \mathcal{G}_λ is a connected graph, implying $\text{card}(\mathcal{E}_{\lambda t}) = \text{card}(\mathcal{N}_\lambda) - 1$ for a spanning tree on the wire basket. Consequently, we obtain

$$\text{card}(\mathcal{E}_{\lambda c}) = \text{card}(\mathcal{E}_\lambda) - \text{card}(\mathcal{E}_{\lambda t}) = \text{card}(\mathcal{E}_\lambda) - \text{card}(\mathcal{N}_\lambda) + 1 = \text{card}(\mathcal{F}_{\text{dec}}) - \text{card}(C_{\text{dec}}) - c_1 + 1, \tag{21}$$

due to (20). This proves Theorem 1 because $\text{card}(\mathcal{E}_{\lambda c})$ only depends on the topology and the given decomposition of Ω but not on $\text{card}(\mathcal{E})$, i.e. mesh size h and degree p . \square

Theorem 1 can be applied to more general settings than the IGA multipatch decomposition of Ω but we neglect a detailed discussion. As a consequence of Theorem 1 and its proof, we can state Corollary 2 as an extension.

Corollary 2. *If the decomposition of Ω satisfies $\text{card}(\mathcal{F}_{\text{dec}}) = \mathcal{O}(N)$, then $\text{card}(\mathcal{E}_{\lambda c}) = \mathcal{O}(N)$ for the number of subdomains N .*

Proof. Clearly, we have that $\text{card}(C_{\text{dec}}) = N$ and that c_1 is independent of N . Hence, $\text{card}(\mathcal{E}_{\lambda c}) = \mathcal{O}(N)$ follows from (21) if $\text{card}(\mathcal{F}_{\text{dec}}) = \mathcal{O}(N)$. \square

In the IGA multipatch setting, we know that every patch, i.e. every subdomain, has six faces. The sum of all patches counts every interface twice such that $0 < \text{card}(\mathcal{F}_{\text{dec}}) = 6N - n_{\text{intf}} = \mathcal{O}(N)$ because $n_{\text{intf}} < 6N$ holds for the number of interfaces n_{intf} . Therefore, we know that

$$\text{card}(\mathcal{E}_{\lambda c}) = \mathcal{O}(N) \tag{22}$$

holds for the IGA multipatch decomposition of Ω due to Corollary 2.

Let us address the question whether a spanning tree \mathcal{E}_t which provides also a spanning tree $\mathcal{E}_{\lambda t}$ when restricted to the wire basket \mathcal{G}_λ can be constructed to also provide a spanning tree \mathcal{E}_{DI} when restricted to the Dirichlet boundary \mathcal{G}_{D} . To answer this it is necessary to introduce a classification of different wire basket subsets depending on which facet type they contribute to. First, we denote the intersections of different facet types as $\mathcal{E}_{\text{DI}} := \mathcal{E}_{\text{D}} \cap \mathcal{E}_t$, $\mathcal{E}_{\text{DN}} := \mathcal{E}_{\text{D}} \cap \mathcal{E}_{\text{N}}$ and $\mathcal{E}_{\text{NI}} := \mathcal{E}_{\text{N}} \cap \mathcal{E}_t$ respectively. Then, if edges are shared by two facets of the same type, we write $\mathcal{E}_{\text{DD}} := (\mathcal{E}_\lambda \cap \mathcal{E}_{\text{D}}) \setminus (\mathcal{E}_{\text{DI}} \cup \mathcal{E}_{\text{DN}})$, $\mathcal{E}_{\text{NN}} := (\mathcal{E}_\lambda \cap \mathcal{E}_{\text{N}}) \setminus (\mathcal{E}_{\text{DN}} \cup \mathcal{E}_{\text{NI}})$ and $\mathcal{E}_{\text{II}} := (\mathcal{E}_\lambda \cap \mathcal{E}_t) \setminus (\mathcal{E}_{\text{DI}} \cup \mathcal{E}_{\text{NI}})$, respectively. Altogether, the relations

$$\mathcal{E}_\lambda = \mathcal{E}_{\text{DI}} \cup \mathcal{E}_{\text{DN}} \cup \mathcal{E}_{\text{NI}} \cup \mathcal{E}_{\text{II}} \cup \mathcal{E}_{\text{DD}} \cup \mathcal{E}_{\text{NN}}, \quad \mathcal{E}_t = \mathcal{E}_{\text{D}} \cup \mathcal{E}_{\text{N}} \cup \mathcal{E}_t \quad \text{and} \quad \mathcal{E}_\lambda \subseteq \mathcal{E}_t$$

hold. Note that the corresponding node subsets, e.g. \mathcal{N}_{NN} , are considered to be defined similarly. Through introducing a hierarchy on growing the tree first on \mathcal{E}_{DI} , then \mathcal{E}_{DN} and at last to \mathcal{E}_{NI} , we preserve the ‘‘Dirichlet first’’ condition on the wire basket and the remaining tree can be grown into the facets independently. After that, we extend the tree to \mathcal{E}_{DD} , \mathcal{E}_{NN} and \mathcal{E}_{II} to obtain that $\mathcal{E}_{\lambda t}$ is a spanning tree on \mathcal{G}_λ such that Theorem 1 can be employed. Note that \mathcal{E}_{II} , \mathcal{E}_{DD} and \mathcal{E}_{NN} can be handled independently of each other and that we assign different weights only for convenience. The remaining steps consist of extending the tree to \mathcal{G}_t and then to \mathcal{G} . This hierarchy can be obtained by employing Kruskal’s algorithm with the presented weights in Fig. 2. Note that it is possible to parallelize the extension into certain subgraphs such that the general complexity $\mathcal{O}(n_g \log(n_g))$ can be improved. We have visualized the weights and the resulting tree for an example configuration in Fig. 3.

4. Dual-primal formulation for 3D magnetostatics

In this section, we derive and explain the typical dual-primal approach in Section 4.1. This method requires selecting appropriate, constrained DOFs for which continuity is enforced by reducing them to only one DOF. These are denoted as ‘primal’ and we present such a choice in Section 4.2. Additionally, we prove that our choice satisfies all requirements to obtain a scalable TI method.

4.1. Dual-primal tearing and interconnecting

Starting with (18), where the Dirichlet boundary is already eliminated, we further split \mathbf{a} into tree DOFs \mathbf{a}_t , primal DOFs \mathbf{a}_p and the remaining ones \mathbf{a}_r . We write n_p for the number of the primal DOFs which depends on the, for now unspecified, selection. Incorporating $\mathbf{a}_t = \mathbf{0}$ and reordering the equations leads to

$$\begin{bmatrix} \mathbf{K}_{\text{rr}} & \mathbf{K}_{\text{rp}} & \mathbf{B}_r^T \\ \mathbf{K}_{\text{pr}} & \mathbf{K}_{\text{pp}} & \mathbf{B}_p^T \\ \mathbf{B}_r & \mathbf{B}_p & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{a}_r \\ \mathbf{a}_p \\ \tilde{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{j}_r \\ \mathbf{j}_p \\ \mathbf{0} \end{bmatrix}. \tag{23}$$

Note that removing tree-DOFs is similar to eliminating Dirichlet DOFs in the sense that zero rows are removed again to obtain the reduced coupling matrix $\tilde{\mathbf{B}} = [\tilde{\mathbf{B}}_r \quad \tilde{\mathbf{B}}_p]$. Hence, we write $\tilde{\mathbf{B}}_p$ and $\tilde{\mathbf{B}}_r$ for the respective primal and remaining modified coupling

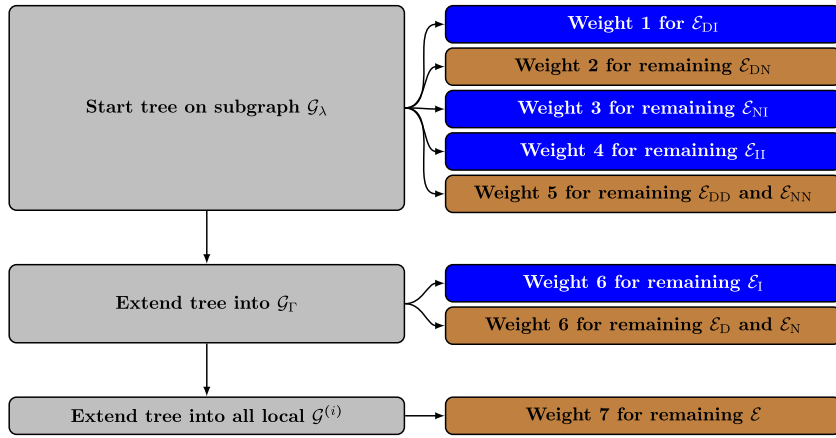


Fig. 2. Weight distribution for Kruskal's algorithm to generate a consistent tree. Tree generation for brown boxes can be parallelized for every subdomain. The blue boxes need global context and no parallelization is possible. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

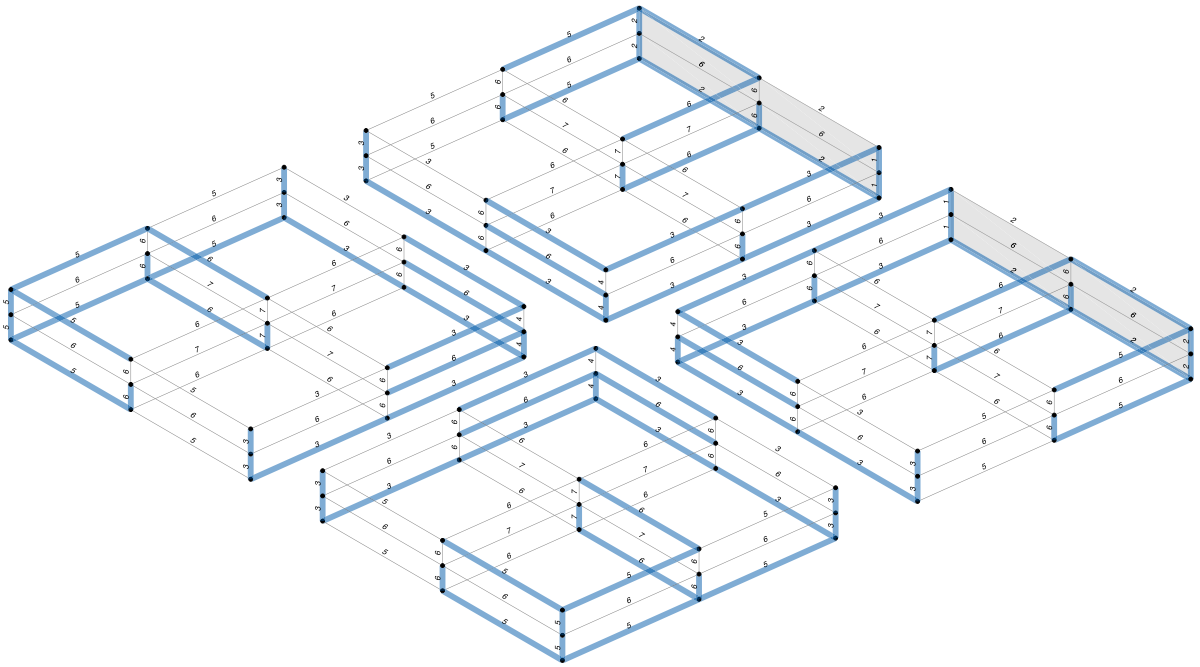


Fig. 3. Local graphs with prioritization, tree edges (blue) and Dirichlet sides marked in gray. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

constraints as well as $\bar{\lambda}$ for the multipliers. We define a matrix $C_p \in \{0,1\}^{n_p \times n_{gp}}$ which is the kernel representation of \bar{B}_p , i.e. $\ker(\bar{B}_p) = \text{im}(C_p)$ with $n_{gp} = \dim(\ker(\bar{B}_p))$. Substituting $a_p = C_p p$ in (23), removing the constraints associated with the zero rows in \bar{B}_r and multiplying the second line of (23) with C_p yields

$$\begin{bmatrix} K_{rr} & K_{rp}C_p & B_r^T \\ C_p^T K_{pr} & C_p^T K_{pp}C_p & 0 \\ B_r & 0 & 0 \end{bmatrix} \begin{bmatrix} a_r \\ p \\ \lambda_r \end{bmatrix} = \begin{bmatrix} j_r \\ C_p^T j_p \\ 0 \end{bmatrix} \quad (24)$$

as the global dual-primal system with remaining multipliers $\lambda_r \in \mathbb{R}^{m_r}$. These steps effectively reduce each set of coupled DOFs in a_p to one DOF in p .

To enable parallelism by Schur complements in a typical dual-primal TI method, one may require that

$$K_{rr} = \text{diag}(K_{rr}^{(1)}, \dots, K_{rr}^{(N)}),$$

where every local matrix $\mathbf{K}_{rr}^{(i)}$ is invertible. In other words, fixing the tree DOFs and selecting appropriate primal DOFs needs to remove all local kernels. Assuming that this holds, one can eliminate

$$\mathbf{a}_r = \mathbf{K}_{rr}^{-1} (\mathbf{j}_r - \mathbf{K}_{rp} \mathbf{C}_p \mathbf{p} - \mathbf{B}_r^\top \lambda_r) \quad (25)$$

in the first line of (24). Inserting (25) in the second and third line of (24) results in

$$\begin{bmatrix} \mathbf{C}_p^\top \mathbf{K}_{pr} \mathbf{K}_{rr}^{-1} \mathbf{K}_{rp} \mathbf{C}_p - \mathbf{C}_p^\top \mathbf{K}_{pp} \mathbf{C}_p & \mathbf{C}_p^\top \mathbf{K}_{pr} \mathbf{K}_{rr}^{-1} \mathbf{B}_r^\top \\ \mathbf{B}_r \mathbf{K}_{rr}^{-1} \mathbf{K}_{rp} \mathbf{C}_p & \mathbf{B}_r \mathbf{K}_{rr}^{-1} \mathbf{B}_r^\top \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \lambda_r \end{bmatrix} = \begin{bmatrix} \mathbf{C}_p^\top \mathbf{K}_{pr} \mathbf{K}_{rr}^{-1} \mathbf{j}_r - \mathbf{C}_p^\top \mathbf{j}_p \\ \mathbf{B}_r \mathbf{K}_{rr}^{-1} \mathbf{j}_r \end{bmatrix}. \quad (26)$$

We shorten the notation by introducing

$$\mathbf{F} := \mathbf{C}_p^\top \mathbf{K}_{pr} \mathbf{K}_{rr}^{-1} \mathbf{K}_{rp} \mathbf{C}_p - \mathbf{C}_p^\top \mathbf{K}_{pp} \mathbf{C}_p,$$

$$\mathbf{G} := \mathbf{B}_r \mathbf{K}_{rr}^{-1} \mathbf{K}_{rp} \mathbf{C}_p,$$

$$\mathbf{W} := \mathbf{B}_r \mathbf{K}_{rr}^{-1} \mathbf{B}_r^\top,$$

$$\mathbf{d} := \mathbf{C}_p^\top \mathbf{K}_{pr} \mathbf{K}_{rr}^{-1} \mathbf{j}_r - \mathbf{C}_p^\top \mathbf{j}_p,$$

$$\mathbf{e} := \mathbf{B}_r \mathbf{K}_{rr}^{-1} \mathbf{j}_r,$$

which leads to

$$\begin{bmatrix} \mathbf{F} & \mathbf{G}^\top \\ \mathbf{G} & \mathbf{W} \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \lambda_r \end{bmatrix} = \begin{bmatrix} \mathbf{d} \\ \mathbf{e} \end{bmatrix}. \quad (27)$$

For this, we apply yet again a Schur complement to eliminate \mathbf{p} and obtain

$$\mathbf{S} \lambda_r = (\mathbf{G} \mathbf{F}^{-1} \mathbf{G}^\top - \mathbf{W}) \lambda_r = \mathbf{G} \mathbf{F}^{-1} \mathbf{d} - \mathbf{e} \quad (28)$$

$$\mathbf{a}_p = \mathbf{C}_p \mathbf{F}^{-1} (\mathbf{d} - \mathbf{G}^\top \lambda_r) \quad (29)$$

$$\mathbf{a}_r = \mathbf{K}_{rr}^{-1} (\mathbf{j}_r - \mathbf{K}_{rp} \mathbf{a}_p - \mathbf{B}_r^\top \lambda_r). \quad (30)$$

Note that the inherent parallelism becomes evident when all matrices are split into local contributions. Details are omitted but an interested reader is referred to [11].

Typically, (28) is solved using iterative methods such as the preconditioned conjugate gradient (PCG) method as described in, e.g. [11,46]. The condition number of the interface problem has a major influence on the scalability of the method which motivates looking at appropriate preconditioners in more detail. We follow the approach of Farhat et al. in [46] and investigate the lumped

$$\mathbf{M}_L^{-1} := \mathbf{D}_s \mathbf{B}_r \mathbf{K}_{rr}^{-1} \mathbf{B}_r^\top \mathbf{D}_s \quad (31)$$

as well as the Dirichlet preconditioner

$$\mathbf{M}_D^{-1} := \mathbf{D}_s \mathbf{B}_r \mathbf{S}_{r|_r} \mathbf{B}_r^\top \mathbf{D}_s \quad \text{with} \quad \mathbf{S}_{r|_r} := \mathbf{K}_{r|_r} - \mathbf{K}_{r|_r \mathcal{V}} \mathbf{K}_{r \mathcal{V}}^{-1} \mathbf{K}_{r \mathcal{V} |_r} \quad (32)$$

in Section 5.3. For both approaches, it is necessary to further split the remaining DOFs \mathbf{a}_r into \mathbf{a}_{r_i} , those which belong to an interface, and \mathbf{a}_{r_v} , those which are not constrained in any way. The diagonal scaling matrix \mathbf{D}_s is necessary to treat material jumps between subdomains, [12,46]. In our numerical examples, we focus on problems with homogeneous material and use the identity matrix $\mathbf{D}_s = \mathbf{I}$. More general scaling matrices can be constructed by using “weighted counting functions” as explained in [12].

4.2. Selection of primal DOFs and invertibility

In this subsection, we discuss the choice of appropriate primal DOFs. We know that the global problem (12) is gauged appropriately because our tree is constructed to provide that. One can interpret (12) as a special variant of (24) where all interface DOFs are selected as primal. In other words, after eliminating a globally consistent tree as constructed in Section 3, the system in (24) is solvable because no issues with cross-edges remain and global gradients are taken care of. This implies that global invertibility of the system matrix in (24) only depends on our choice of primal DOFs \mathbf{a}_p . The local contribution $\mathbf{a}_p^{(i)}$ of subdomain $\Omega^{(i)}$ corresponds to the set of local edges $\mathcal{E}_p^{(i)}$. Its global counterpart (computed with union operation) is called \mathcal{E}_p and its edges correspond to the DOFs in \mathbf{p} . We have that $n_p = \sum_{i=1}^N \text{card}(\mathcal{E}_p^{(i)})$ and $n_{ep} = \text{card}(\mathcal{E}_p)$ hold for the respective number of DOFs/edges.

Before defining $\mathcal{E}_p^{(i)}$ and \mathcal{E}_p properly, we analyze which remaining issues our selection needs to fix. First, to obtain global invertibility of (24), it is necessary to remove issues related to the cross-edges $\mathcal{E}_{II}^{(i)}$. Due to setting all local tree DOFs to zero and the matching of tree edges on interfaces (originating from tearing the global tree apart), all issues from the cross-edges $\mathcal{E}_{II}^{(i)} \cap \mathcal{E}_t^{(i)}$ are already fixed. The remaining ones $\mathcal{E}_{II}^{(i)} \setminus \mathcal{E}_t^{(i)}$ need to be selected as primal edges to obtain global invertibility. However, this does not suffice to obtain local invertibility, i.e. that every local matrix $\mathbf{K}_{rr}^{(i)}$ is invertible. An example is provided in Fig. 4 where the matrices corresponding to subdomains in the middle and the lower left are not invertible. It can be seen easily because the underlying graph $\mathcal{G}_e^{(i)} := (\mathcal{N}^{(i)}, \mathcal{E}_e^{(i)})$, formed by all the nodes from $\mathcal{G}^{(i)}$ and the locally eliminated edges

$$\mathcal{E}_e^{(i)} := \mathcal{E}_D^{(i)} \cup \mathcal{E}_t^{(i)} \cup \mathcal{E}_p^{(i)}, \quad (33)$$

is not connected. In other words, there is no path from every node in $\mathcal{N}^{(i)}$ to all other nodes in $\mathcal{N}^{(i)}$ which only consists of edges in $\mathcal{E}_e^{(i)}$ for the affected subdomains. We formalize this in Lemma 3.

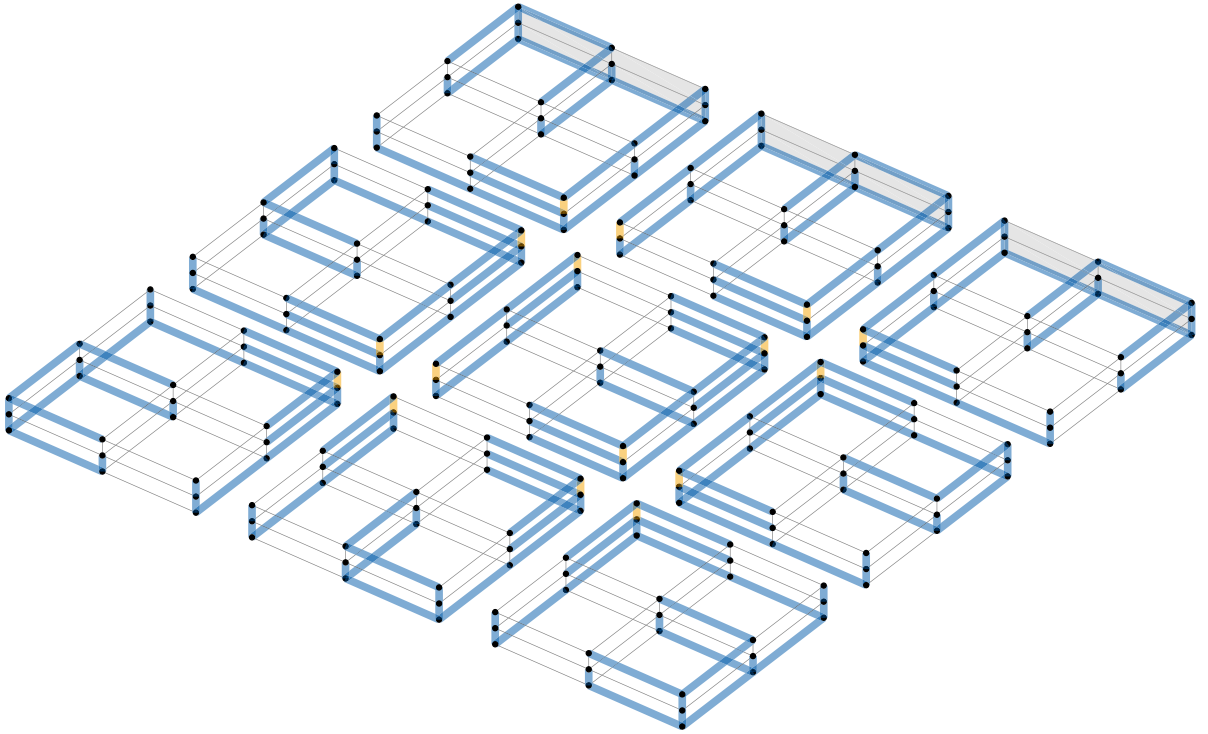


Fig. 4. Local graphs for a problematic configuration with tree edges (blue), cross-edges (orange) and Dirichlet facets marked in gray. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Lemma 3. *The matrix $\mathbf{K}_{\text{tr}}^{(i)}$ is invertible if the graph of locally eliminated DOFs $\mathcal{G}_e^{(i)} = (\mathcal{N}^{(i)}, \mathcal{E}_e^{(i)})$ is connected.*

Proof. The proof is based on the arguments from [24,40]. If all the DOFs from a spanning tree are removed from a stiffness matrix belonging to the curl-curl-operator, it is invertible which implies that all its column vectors are linearly independent. A spanning tree has the fundamental property that it connects all nodes with a minimal amount of edges. If we eliminate the DOFs related to a spanning tree along with some additional edges, we know that the resulting matrix is still invertible because removing one column and one row of the stiffness matrix preserves the linear independence of its column vectors. If $\mathcal{G}_e^{(i)}$ is connected, and since it contains all the nodes in $\mathcal{N}^{(i)}$, we implicitly know that $\mathcal{E}_e^{(i)}$ contains a spanning tree on $\mathcal{G}^{(i)}$. Hence, the statement for invertibility can be applied to the local stiffness matrix $\mathbf{K}^{(i)}$ and its reduced form $\mathbf{K}_{\text{tr}}^{(i)}$ with graph $\mathcal{G}_e^{(i)}$, which concludes the proof. \square

Remark 2. Eliminating more DOFs than necessary for regularizing the corresponding stiffness matrix gives rise to issues regarding consistency. In other words, only a certain amount of DOFs can be chosen freely when solving a singular system. The ones which are additionally eliminated need to be prescribed consistently. In our context, these additional DOFs are either primal or belong to the Dirichlet boundary. The Dirichlet DOFs are automatically consistent if the boundary condition is consistent with the original problem. The primal DOFs are computed by solving the coarse problem (29) which preserves the consistent, global context.

Now, all tools are ready to define our selection of primal edges $\mathcal{E}_p^{(i)}$, which we do in Theorem 4: we prove that global and local invertibility can be obtained with our algorithm.

Theorem 4. *Let the tree be generated with Kruskal's algorithm and weights as in Fig. 2, and let the local primal edges be chosen as*

$$\mathcal{E}_p^{(i)} = \left(\mathcal{E}_{\text{II}}^{(i)} \cup \mathcal{E}_{\text{NI}}^{(i)} \right) \setminus \mathcal{E}_t^{(i)}. \quad (34)$$

Then system (24) satisfies global and local invertibility.

Proof. Global invertibility originates from the fact that all edges $\mathcal{E}_{\text{II}}^{(i)} \setminus \mathcal{E}_t^{(i)}$ are contained in $\mathcal{E}_p^{(i)}$ and that the tree edges \mathcal{E}_t and correspondingly $\mathcal{E}_t^{(i)}$ are chosen such that the multipatch problem is gauged appropriately.

For local invertibility we use [Lemma 3](#), for which we need to show that the combination of the tree construction and our final selection of primal DOFs yields connected graphs $\mathcal{G}_e^{(i)}$ on every subdomain. The proof is carried out separately for every subdomain $\Omega^{(i)}$ in three steps. The three steps are respectively: First, showing that all nodes $\mathcal{N}_\lambda^{(i)}$ are connected by edges in $\mathcal{E}_{\lambda e}^{(i)} := \mathcal{E}_\lambda^{(i)} \cap \mathcal{E}_e^{(i)}$. Second, that the extension of Kruskal’s algorithm onto $\mathcal{N}_\Gamma^{(i)}$ and the elimination of all Dirichlet DOFs connects all nodes $\mathcal{N}_\Gamma^{(i)}$ with edges in $\mathcal{E}_{\Gamma e}^{(i)} := \mathcal{E}_\Gamma^{(i)} \cap \mathcal{E}_e^{(i)}$. Third, that the extension of the tree into the interior yields a connected graph $\mathcal{G}_e^{(i)}$ for every subdomain.

By reformulating $\mathcal{E}_{\lambda e}^{(i)}$, one obtains

$$\begin{aligned} \mathcal{E}_{\lambda e}^{(i)} &= \left(\mathcal{E}_D^{(i)} \cap \mathcal{E}_\lambda^{(i)} \right) \cup \left(\mathcal{E}_t^{(i)} \cap \mathcal{E}_\lambda^{(i)} \right) \cup \left(\mathcal{E}_p^{(i)} \cap \mathcal{E}_\lambda^{(i)} \right) \\ &= \left(\mathcal{E}_{DD}^{(i)} \cup \mathcal{E}_{DI}^{(i)} \cup \mathcal{E}_{DN}^{(i)} \right) \cup \left(\left(\mathcal{E}_{II}^{(i)} \cup \mathcal{E}_{NI}^{(i)} \cup \mathcal{E}_{NN}^{(i)} \right) \cap \mathcal{E}_t^{(i)} \right) \cup \left(\left(\mathcal{E}_{II}^{(i)} \cup \mathcal{E}_{NI}^{(i)} \right) \setminus \mathcal{E}_t^{(i)} \right) \\ &= \underbrace{\mathcal{E}_{DD}^{(i)} \cup \mathcal{E}_{DI}^{(i)} \cup \mathcal{E}_{DN}^{(i)} \cup \mathcal{E}_{NI}^{(i)} \cup \mathcal{E}_{II}^{(i)}}_{=\mathcal{E}_\lambda^{(i)} \setminus \mathcal{E}_{NN}^{(i)}} \cup \left(\mathcal{E}_{NN}^{(i)} \cap \mathcal{E}_t^{(i)} \right). \end{aligned}$$

This implies that all nodes in $\mathcal{N}_\lambda^{(i)} \setminus \mathcal{N}_{NN}^{(i)}$ are already connected because all available edges $\mathcal{E}_\lambda^{(i)} \setminus \mathcal{E}_{NN}^{(i)}$ are contained in $\mathcal{E}_{\lambda e}^{(i)}$. The remaining ones $\mathcal{N}_{NN}^{(i)}$ are also connected because Kruskal’s algorithm with given weights constructs a tree on \mathcal{G}_λ first. In detail, this is valid because all nodes in $\mathcal{N}_{NN}^{(i)}$ and all edges in $\mathcal{E}_{NN}^{(i)}$ stay the same in both the local as well as the global context (no components of the sets are removed in the union operation). In other words, all nodes in \mathcal{N}_{NN} are connected by edges in $\mathcal{E}_{NN} \cap \mathcal{E}_t$ which in turn implies the same for the respective local sets. Altogether, we therefore know that all nodes of $\mathcal{N}_\lambda^{(i)}$ are connected by edges in $\mathcal{E}_{\lambda e}^{(i)}$.

For the second step, we need to show that all nodes in $\mathcal{N}_\Gamma^{(i)}$ are connected by edges in $\mathcal{E}_{\Gamma e}^{(i)}$. We already know that this holds for $\mathcal{N}_\Gamma^{(i)} \subseteq \mathcal{N}_\Gamma^{(i)}$ with edges from $\mathcal{E}_{\lambda e}^{(i)} \subseteq \mathcal{E}_{\Gamma e}^{(i)}$. Next, Kruskal’s algorithm extends the tree into every facet independently which connects all nodes in $\mathcal{N}_\Gamma^{(i)}$ to the already available tree (whose edges are part of the eliminated edges). Furthermore, Dirichlet edges are eliminated which does not influence the connectedness. Consequently, all nodes in $\mathcal{N}_\Gamma^{(i)}$ are connected by edges in $\mathcal{E}_{\Gamma e}^{(i)}$.

The same arguments can be employed in the third step, where extending the tree into every subdomain implies that all nodes in $\mathcal{N}^{(i)}$ are connected by edges in $\mathcal{E}_e^{(i)}$. Hence, all graphs $\mathcal{G}_e^{(i)}$ are connected and every local block in \mathbf{K}_{rr} is invertible due to [Lemma 3](#). \square

Remark 3. Our choice of $\mathcal{E}_p^{(i)}$ is both straightforward to implement and efficient. It is simple because we do not need an algorithm to search for edges that are necessary to obtain local connectedness. At the same time, we may select more edges than needed for local connectedness but we know $\mathcal{E}_p \subseteq \mathcal{E}_{\lambda c}$. Due to this and [\(22\)](#) we know that

$$n_{gp} = \text{card}(\mathcal{E}_p) \leq \text{card}(\mathcal{E}_{\lambda c}) = \mathcal{O}(N) \quad \Rightarrow \quad n_{gp} = \mathcal{O}(N) \tag{35}$$

holds. Hence, the size of the coarse problem n_{gp} only grows linearly with the number of subdomains N . This is related to efficiency, since the size of the coarse problem is a major factor for scalability because of the sequential computations necessary to evaluate the action of \mathbf{F}^{-1} .

Remark 4. To the best of our understanding, our procedure is very similar to the approach from Yao et al. in [\[22,23\]](#) when considering a pure Dirichlet problem. Their construction is not explicitly described but both approaches shall eventually satisfy the same conditions. In contrast, we extend the underlying idea to problems with mixed boundary conditions by introducing additional edge weights and prove local invertibility for both the full Dirichlet and the more general case.

5. Numerical experiments

In this section, we verify that the overall algorithm correctly solves general 3D magnetostatic problems and investigate some performance indicators. We focus on supporting our theoretical findings rather than high-performance computing. We employ the IGA-library GeopDEs [\[28\]](#) in Matlab 2023a [\[47\]](#) for all tests. We obtain the actions of \mathbf{K}_{rr}^{-1} and \mathbf{F}^{-1} on vectors via appropriate factorizations (e.g. \mathbf{LDL}^T) provided by Matlab. The factorization of \mathbf{F} in combination with forward/backward substitution can be exploited to solve [\(28\)](#) iteratively using a Conjugate Gradient (CG) method without explicitly assembling \mathbf{S} . After obtaining λ_r from [\(28\)](#), one can recover \mathbf{a}_p and \mathbf{a}_r from [\(29\)](#) and [\(30\)](#), respectively.

For tests related to numerical accuracy we use the method of manufactured solutions [\[48, Sec. 6.3\]](#). We predefine the magnetic vector potential

$$\mathbf{A}_{\text{ana}}(x, y, z) = \begin{bmatrix} \cos(y) \cos(z) \sin(x) \\ -2 \cos(x) \cos(z) \sin(y) \\ \cos(x) \cos(y) \sin(z) \end{bmatrix} \quad \Rightarrow \quad \mathbf{B}_{\text{ana}}(x, y, z) = \text{curl } \mathbf{A}_{\text{ana}} = \begin{bmatrix} -3 \cos(x) \sin(y) \sin(z) \\ 0 \\ 3 \cos(z) \sin(x) \sin(y) \end{bmatrix}. \tag{36}$$

Then, we compute the source current density and the boundary conditions by applying the respective operators and traces in [\(3\)–\(5\)](#) to \mathbf{A}_{ana} for a given domain configuration. This allows us to study inhomogeneous Dirichlet and Neumann boundary conditions. For simplicity, we assume $\nu \equiv 1$.

Numerical errors will be measured by the subdomain-wise $H(\text{curl}; \Omega)$ seminorm of \mathbf{A}

$$\epsilon_B := \sqrt{\sum_{i=1}^N \|\mathbf{B}_{\text{ana}} - \mathbf{B}_{\text{num}}\|_{L^2(\Omega^{(i)})}^2}, \tag{37}$$

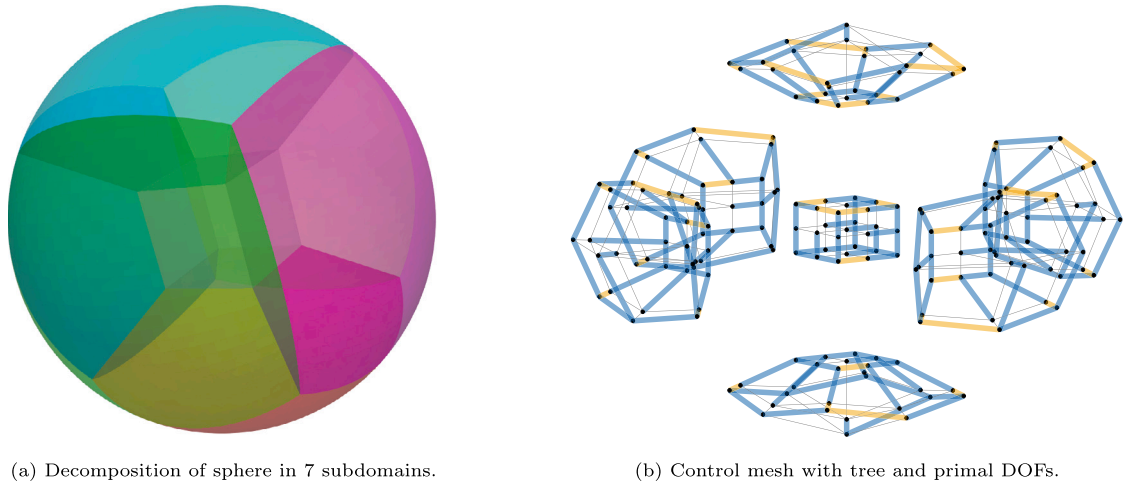


Fig. 5. Geometry and exemplary control mesh for spherical geometry including tree DOFs (blue), and primal DOFs (orange). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

which corresponds to the L^2 -error of the magnetic flux density \mathbf{B} . To measure the rate of convergence, we investigate the number of subdivisions in each direction in each subdomain s_h , for which we may expect

$$\epsilon_B \leq Ch^p = \tilde{C}s_h^{-p} \tag{38}$$

for suitable $C, \tilde{C} \in \mathbb{R}_{>0}$.

Finally, the condition numbers of several matrices are investigated. We use the following definition [49]

$$\kappa(\mathbf{X}) = \frac{\sigma_{\max}(\mathbf{X})}{\sigma_{\min}(\mathbf{X})},$$

where $\sigma_{\max}(\mathbf{X})$ and $\sigma_{\min}(\mathbf{X})$ are the maximal and minimal absolute eigenvalues of \mathbf{X} , respectively. In practice, these are estimated by an approach based on the Lanczos algorithm [50, Sec. 6.7.3].

5.1. Proof of concept on spherical geometry

In this subsection, we provide experimental proof that our spanning tree construction and the IETI-DP algorithm work as promised. We use a spherical geometry, with radius 2, as depicted in Fig. 5(a), and solve a manufactured problem with solution (36) for either Dirichlet $\Gamma_D = \partial\Omega$ or Neumann boundary conditions $\Gamma_N = \partial\Omega$. We visualize the control mesh with tree and primal edges in Fig. 5(b) and the numerical flux density in Fig. 6(a). The main experiment of this subsection is a convergence study of ϵ_B with respect to s_h . The results are visualized in Fig. 6(b) where one can observe the expected optimal behavior as given in (38). Furthermore, no issues due to singular matrices were observed implying that all $\mathbf{K}_{tr}^{(i)}$ are gauged correctly. Therefore, we conclude that the combination of our tree construction, the primal DOF selection and IETI-DP can be used to compute correct solutions for the flux density \mathbf{B} to (3)–(5).

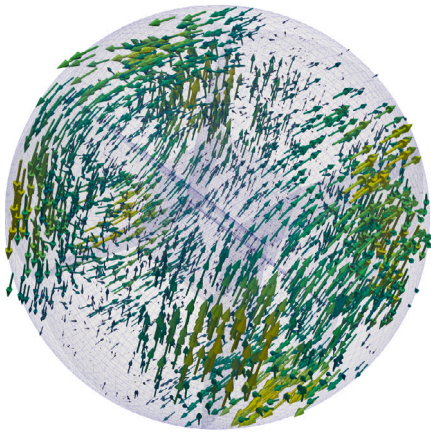
5.2. Experiments on toroidal geometry

In this subsection, we examine the toroidal geometry in Fig. 7(a) and the associated issues with the harmonic field h . This domain is not simply connected, i.e. its Betti number $b_1 = 1 > 0$ which means that there is a hole in Ω . We either employ full Neumann conditions, i.e. $\Gamma_N = \partial\Omega$, or the setup with mixed boundary

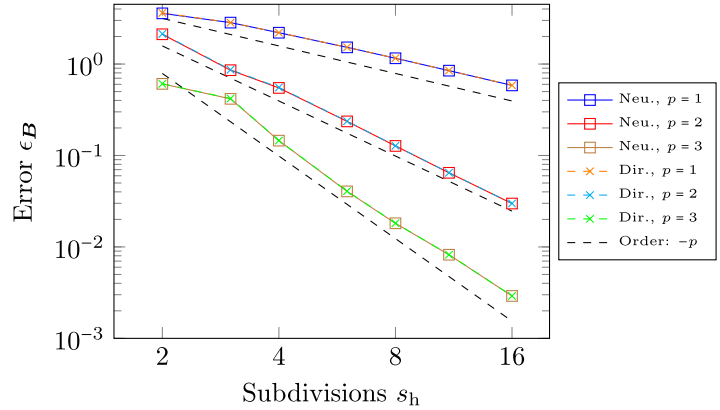
$$\Gamma_D := \{r = r_i, \theta \in (0, 2\pi], z \in (0, 2)\} \quad \text{and} \quad \Gamma_N := \partial\Omega \setminus \bar{\Gamma}_D.$$

We expect problems related to a non-trivial kernel to occur only for the full Neumann, but not in the mixed setting. When the Dirichlet constraints, which form a loop around the hole, are strongly imposed, the harmonic contribution to the kernel is automatically removed. However, in the full Neumann setting, we need to add an edge to the spanning tree that closes a loop around the hole [51].

In practice, the predicted issue is not immediately visible: In Fig. 7(b) optimal convergence behavior is observable in both the Neumann and the mixed setup. The reason is that `Matlab`'s `LDLT`-factorization can cope with the singular systems, but issues warnings. However, when examining the condition numbers, we see that $\kappa(\mathbf{F})$ cannot be computed with the employed method, see Table 1 for $N = 6$, in the full Neumann boundary case. This indicates a rank deficit. Nevertheless, $\kappa(\mathbf{S})$ and the error ϵ_B remain

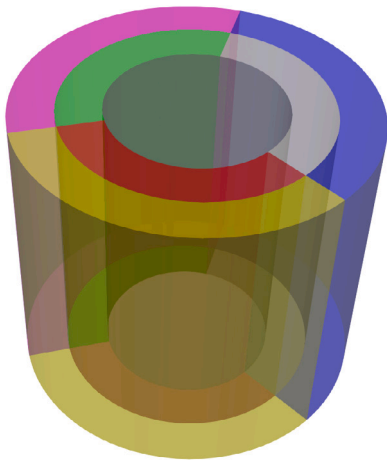


(a) Numerically computed flux density.

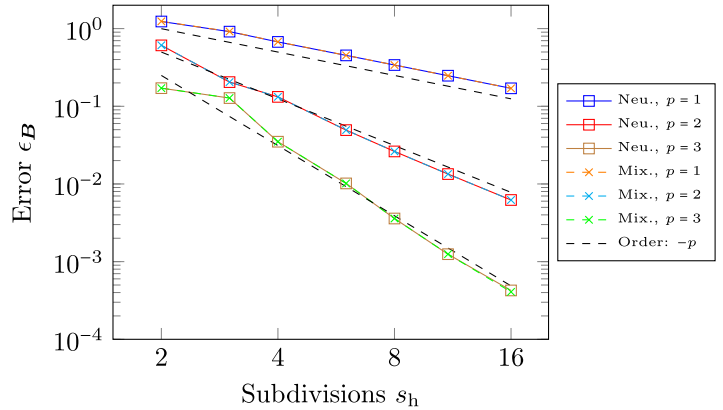


(b) Error of Neumann (square) and Dirichlet (cross) problem.

Fig. 6. Visualization of solution and convergence study for Dirichlet and Neumann problem (with $\epsilon_{tol} = 1 \cdot 10^{-6}$ for the interface problem).



(a) Decomposition of torus in 6 subdomains.



(b) Convergence for mixed BCs and full Neumann problem.

Fig. 7. Visualization of model problem with geometry (height: $h = 2$, inner radius: $r_i = 1$, outer radius: $r_o = 2$) and convergence study for a configuration with Dirichlet boundary conditions around the hole and a configuration with full Neumann boundary conditions (with $\epsilon_{tol} = 1 \cdot 10^{-6}$ for the interface problem).

comparable to the problem with mixed boundary conditions due to the implicit gauge of the applied factorizations. The configuration denoted with ‘Mod.’ in Table 1 refers to the Neumann problem with an additional tree edge added due to cohomology considerations. As expected, this repairs the rank deficit and $\kappa(\mathbf{F})$ behaves normally. A visualization is given in Fig. 8(a) where the violet edge is added to the blue tree. We include a second simpler decomposition with $N = 3$ where the rank deficit can be observed in the local context, i.e. $\kappa(\mathbf{K}_{rr})$ becomes very large for the Neumann setting. This may be surprising since we restrict harmonic fields to be only a global issue, but a closer look at Fig. 8(b) explains this behavior: without the additional edge, the graph $\mathcal{G}_e^{(i)}$ of eliminated DOFs is not connected in one subdomain. However, as before, the MatLab solvers are able to deal with the rank deficit such that the corresponding $\kappa(\mathbf{F})$, $\kappa(\mathbf{S})$ and ϵ_B stay appropriate. But again, adding an edge that closes a loop around the hole eliminates the additional kernel element and its associated difficulties. This verifies our assumptions and demonstrates that our algorithm can correctly handle both mixed boundary conditions and geometries with $b_1 \neq 0$ if a little additional care is taken. We recommend using a cohomology-aware tree construction, such as [51], to automatically create appropriate “belted” trees that directly include additional loop-closing edges.

5.3. Numerical scalability

To evaluate the behavior of our algorithm for large problems, we follow the tests and explanations of Farhat et al. in [11] and references therein. They determine scalability by looking at the sequential bottlenecks in the dual-primal algorithm. In the context

Table 1

Examination of condition numbers and error for different boundary configurations and a modified gauge for the Neumann problem ($p = 2$, $s_h = 8$ and $\epsilon_{\text{tol}} = 1 \cdot 10^{-6}$). The additional harmonic kernel introduces numerical difficulties highlighted in bold or with “—” which indicates that the condition number estimation terminated early due to failing internal checks.

Config.	N	$\kappa(\mathbf{K}_{\text{tr}})$	$\kappa(\mathbf{F})$	$\kappa(\mathbf{S})$	ϵ_B
Mix.	3	$1.95 \cdot 10^4$	$1.00 \cdot 10^0$	$6.42 \cdot 10^2$	$2.63 \cdot 10^{-2}$
Neu.	3	$4.12 \cdot 10^{14}$	$1.00 \cdot 10^0$	$8.16 \cdot 10^2$	$2.63 \cdot 10^{-2}$
Mod.	3	$6.32 \cdot 10^4$	$1.00 \cdot 10^0$	$8.15 \cdot 10^2$	$2.63 \cdot 10^{-2}$
Mix.	6	$1.40 \cdot 10^5$	$1.15 \cdot 10^1$	$1.78 \cdot 10^3$	$2.62 \cdot 10^{-2}$
Neu.	6	$2.82 \cdot 10^5$	—	$1.69 \cdot 10^3$	$2.62 \cdot 10^{-2}$
Mod.	6	$2.82 \cdot 10^5$	$1.28 \cdot 10^1$	$1.69 \cdot 10^3$	$2.62 \cdot 10^{-2}$

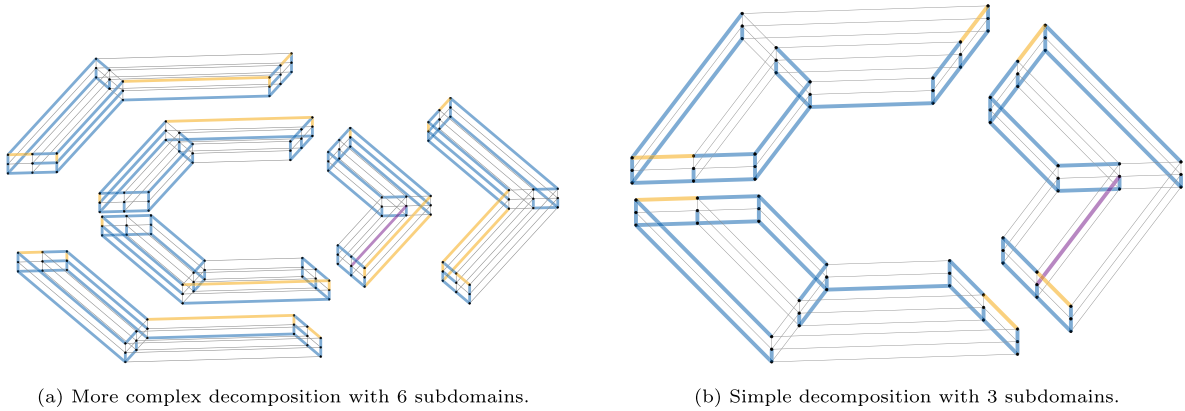


Fig. 8. Different decompositions of the torus, represented by their local meshes for $p = 1$ and $s_h = 2$. The meshes include the tree edges (blue), the primal edges (orange) and the additional tree edge used for removing the harmonic kernel (violet). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

of parallel performance, the sequential part is relevant in both Amdahl’s law (strong scalability) [52,53] and Gustafson’s law (weak scalability) [53,54].

For the following test, we use the unit cube geometry with side length 1, which is decomposed into N equally sized cubes. For the boundary, we employ mixed boundary conditions with a not connected Dirichlet boundary

$$\Gamma_D := \{[x_1, x_2, x_3]^T, \quad x_1, x_3 \in (0, 1), \quad x_2 \in \{0, 1\}\}, \quad \text{and} \quad \Gamma_N := \partial\Omega \setminus \overline{\Gamma_D}.$$

which is handled automatically by our algorithm. Due to this special decomposition, we can introduce the coarse problem size (body diagonal of each subdomain) as $H = \sqrt{3}N^{-1/3}$ and the corresponding number of coarse subdivisions $s_H = \sqrt{3}H^{-1}$. If the local subdivisions are uniform in each subdomain, the relation $h = \sqrt{3}(s_h s_H)^{-1}$ can be derived. Consequently, the well-known FETI-ratio, as discussed in [11] and references therein, can be expressed as $H/h = s_h$. The quantities s_h and s_H allow a concise discussion of the following three test configurations:

1. The first test consists of setting $H = \text{const.}$, i.e., $s_H = \text{const.}$ and varying s_h . In other words, we only change the size of the local problems.
2. The second test consists of setting $h = \text{const.}$, i.e., $s_h s_H = \text{const.}$ In other words, we keep the overall problem size constant while increasing the number of subdomains (i.e., processors). This is equivalent to testing for so-called strong scalability, which follows Amdahl’s law.
3. In the third test, we set $H/h = s_h = \text{const.}$ and vary s_H . In other words, we increase both, the problem size and the number of subdomains/processors, while keeping the workload per processor constant. This is a test of the so-called weak scalability based on Gustafson’s law.

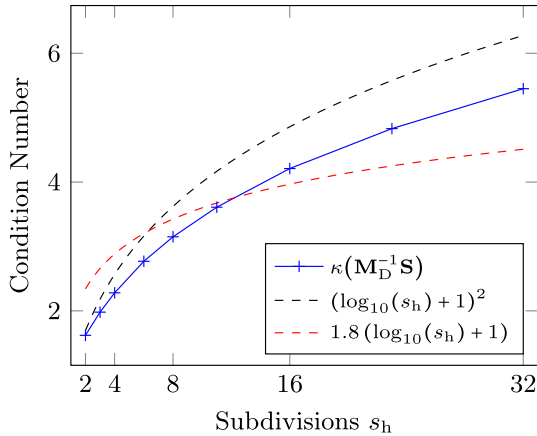
The results of the first scalability test are summarized in Table 2. We obtain the expected convergence orders w.r.t. s_h as in the other numerical tests of this section. We further observe that $\kappa(\mathbf{S})$ deteriorates fast with increasingly refined local discretizations. Applying the lumped preconditioner \mathbf{M}_L^{-1} improves the conditioning already but a better performance is shown by the Dirichlet preconditioner \mathbf{M}_D^{-1} . This numerical test even suggests that the relation

$$\kappa(\mathbf{M}_D^{-1}\mathbf{S}) \leq C (\log_{10}(s_h) + 1)^2 = C (\log_{10}(H/h) + 1)^2 \tag{39}$$

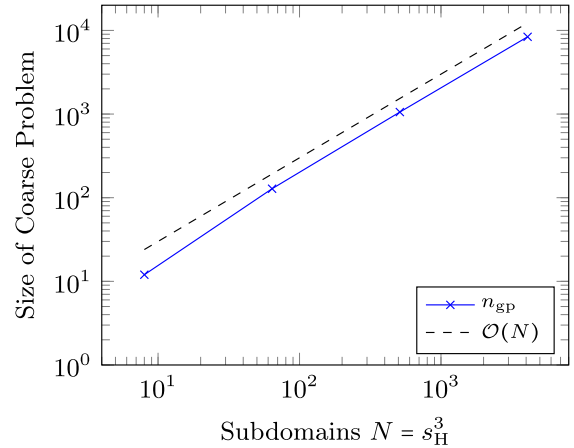
Table 2

First scalability experiment with $s_H = 2$, i.e., $N = 8$ on cube ($\rho = 1$ and $\epsilon_{tol} = 1 \cdot 10^{-6}$). The value $it(\bullet)$ refers to the number of PCG iterations with the respective (preconditioned) system matrix \bullet .

s_H	s_h	n_{gp}	$\kappa(\mathbf{S})$	$\kappa(\mathbf{M}_L^{-1}\mathbf{S})$	$\kappa(\mathbf{M}_D^{-1}\mathbf{S})$	$it(\mathbf{S})$	$it(\mathbf{M}_L^{-1}\mathbf{S})$	$it(\mathbf{M}_D^{-1}\mathbf{S})$	ϵ_B
2	2	12	$1.20 \cdot 10^1$	$3.78 \cdot 10^0$	$1.62 \cdot 10^0$	22	13	7	$2.62 \cdot 10^0$
2	4	12	$6.22 \cdot 10^1$	$2.13 \cdot 10^1$	$2.28 \cdot 10^0$	41	27	9	$1.33 \cdot 10^0$
2	8	12	$2.79 \cdot 10^2$	$1.12 \cdot 10^2$	$3.15 \cdot 10^0$	72	50	11	$6.69 \cdot 10^{-1}$
2	16	12	$1.08 \cdot 10^3$	$5.07 \cdot 10^2$	$4.21 \cdot 10^0$	124	93	14	$3.35 \cdot 10^{-1}$



(a) Condition number with bound.



(b) Size of coarse problem with bound.

Fig. 9. Visualization of scalability indicators condition number of preconditioned interface problem and size of coarse problem. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3

Second scalability experiment with $s_h s_H = 32$ on cube ($\rho = 1$ and $\epsilon_{tol} = 1 \cdot 10^{-6}$). The value $it(\bullet)$ refers to the number of PCG iterations with the respective (preconditioned) system matrix \bullet .

s_H	s_h	n_{gp}	$\kappa(\mathbf{S})$	$\kappa(\mathbf{M}_L^{-1}\mathbf{S})$	$\kappa(\mathbf{M}_D^{-1}\mathbf{S})$	$it(\mathbf{S})$	$it(\mathbf{M}_L^{-1}\mathbf{S})$	$it(\mathbf{M}_D^{-1}\mathbf{S})$	ϵ_B
2	16	12	$1.08 \cdot 10^3$	$5.07 \cdot 10^2$	$4.21 \cdot 10^0$	124	93	14	$3.35 \cdot 10^{-1}$
4	8	128	$3.70 \cdot 10^2$	$1.35 \cdot 10^2$	$3.95 \cdot 10^0$	102	68	13	$3.35 \cdot 10^{-1}$
8	4	1056	$7.77 \cdot 10^1$	$2.57 \cdot 10^1$	$2.94 \cdot 10^0$	55	34	10	$3.35 \cdot 10^{-1}$
16	2	8384	$1.26 \cdot 10^1$	$4.65 \cdot 10^0$	$1.99 \cdot 10^0$	23	14	8	$3.35 \cdot 10^{-1}$

as in [12,55] holds. A visualization is provided in Fig. 9(a) where the dashed, red function was only added as a visual comparison. At last, we see that the size of \mathbf{p} , i.e. n_{gp} , remains constant when the subdomains are refined locally which agrees with our considerations in Remark 3.

The results of the second scalability experiment (strong scaling) are shown in Table 3. We keep the overall problem size constant while increasing the number of subdomains, i.e. we coarsen the local discretizations accordingly. Therefore, it is not surprising that we observe the same error in each simulation run which confirms that a local refinement can be exchanged with a corresponding coarse refinement. For the size of the coarse problem n_{gp} , we observe a linear increase with respect to the number of subdomains $N = s_H^3$ which is visualized in Fig. 9(b). This verifies our considerations in Remark 3 and $n_{gp} = \mathcal{O}(N)$. The size of the coarse problem is further explored in Appendix where we compute relevant numbers, e.g. $\text{card}(\mathcal{E}_{\lambda_c})$, analytically for the configuration of this subsection and compare them to the overall number of DOFs n . For the condition number and the amount of PCG iterations with different preconditioners, we observe a decrease as the number of local subdivisions decreases. This already suggests that the constant $C > 0$ in (39) does not depend on s_H .

The results of the third scalability experiment (weak scaling) shown in Table 4 support the findings of the previous scalability experiments. The same error occurs as in experiment 1, which implies again that a local refinement can be exchanged with a corresponding coarse refinement. The observations of the condition number and PCG iterations of the (preconditioned) interface problem indicate again that the number of subdomains has no or at most a minor influence. Following [11], this implies that our method is weakly (there called numerically) scalable. Only the solving of the coarse problem, although its size scales linearly with the number of subdomains, may remain as a bottleneck for large problems which motivates using inexact TI approaches [56] where (29) is solved iteratively.

Table 4

Third scalability experiment with $s_h = 2$ on cube ($p = 1$ and $\epsilon_{\text{tol}} = 1 \cdot 10^{-6}$). The value $\text{it}(\bullet)$ refers to the number of PCG iterations with the respective (preconditioned) system matrix \bullet .

s_H	s_h	n_{gp}	$\kappa(\mathbf{S})$	$\kappa(\mathbf{M}_L^{-1}\mathbf{S})$	$\kappa(\mathbf{M}_D^{-1}\mathbf{S})$	$\text{it}(\mathbf{S})$	$\text{it}(\mathbf{M}_L^{-1}\mathbf{S})$	$\text{it}(\mathbf{M}_D^{-1}\mathbf{S})$	ϵ_B
2	2	12	$1.20 \cdot 10^1$	$3.78 \cdot 10^0$	$1.62 \cdot 10^0$	22	13	7	$2.62 \cdot 10^0$
4	2	128	$1.39 \cdot 10^1$	$4.78 \cdot 10^0$	$1.90 \cdot 10^0$	25	15	8	$1.33 \cdot 10^0$
8	2	1056	$1.29 \cdot 10^1$	$4.76 \cdot 10^0$	$1.97 \cdot 10^0$	24	14	8	$6.69 \cdot 10^{-1}$
16	2	8384	$1.26 \cdot 10^1$	$4.65 \cdot 10^0$	$1.99 \cdot 10^0$	23	14	8	$3.35 \cdot 10^{-1}$

In summary, we observed the typical behavior for scalable TI methods w.r.t the preconditioned interface problem. Furthermore, we demonstrated that the size of the coarse problem only grows linearly with the number of subdomains and estimated its contribution to the overall system.

6. Conclusions and outlook

In this work, we discussed dual–primal Tearing and Interconnecting approach for magnetostatic problems using IGA edge-elements and mixed boundary conditions in combination with the tree–cotree gauge. In this context, we introduced an explicit way to construct appropriate trees and how to select primal DOFs to obtain local invertibility to enable parallel computations for each subdomain. Furthermore, we proved that the local subdomain contributions are invertible when using our proposed global tree in combination with our choice for the primal DOFs. Extensions to our work are for example considerations on and numerical experiments with inhomogeneous, possibly discontinuous material distributions as well as preconditioning in this context. Additionally, the application to industrial problems and an extension to eddy current problems are of interest.

CRedit authorship contribution statement

Mario Mally: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Bernard Kapidani:** Writing – review & editing, Methodology, Investigation, Formal analysis, Conceptualization. **Melina Merkel:** Writing – review & editing, Methodology, Investigation, Conceptualization. **Sebastian Schöps:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Funding acquisition, Conceptualization. **Rafael Vázquez:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used DeepL and ChatGPT in order to improve the readability and language of this paper. After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The work is supported by the joint DFG/FWF Collaborative Research Centre CREATOR (DFG: Project-ID 492661287/TRR 361; FWF: 10.55776/F90) at TU Darmstadt, TU Graz and JKU Linz. We acknowledge the funding of The “Ernst Ludwig Mobility Grant” of the Association of Friends of Technical University of Darmstadt e.V.

Appendix. Ratio between coarse and full problem

In this appendix, we examine the size of the coarse problem n_{gp} and how it relates to the full number of DOFs $n = \sum_{i=1}^N n_i$ for the configuration of Section 5.3. First, we look at a single subdomain $\Omega^{(i)}$ with s_h subdivisions in every direction for $p = 1$. Note that the following concepts can be extended to settings with $p > 1$ by adding a constant offset to s_h . The wire basket of $\Omega^{(i)}$ consists of $m_e^{(i)} = 12$ macroscopic edges and $m_n^{(i)} = 8$ vertices, each shared by three macroscopic edges. One can easily check that one of these edges contains s_h element edges and $s_h + 1$ nodes which yields

$$\text{card}\left(\mathcal{N}_\lambda^{(i)}\right) = 12s_h - 4 = 12(s_h + 1) - 16 = m_e^{(i)}(s_h + 1) - (3 - 1)m_n^{(i)}$$

$$\text{card}\left(\mathcal{E}_\lambda^{(i)}\right) = 12s_h = m_e^{(i)}s_h$$

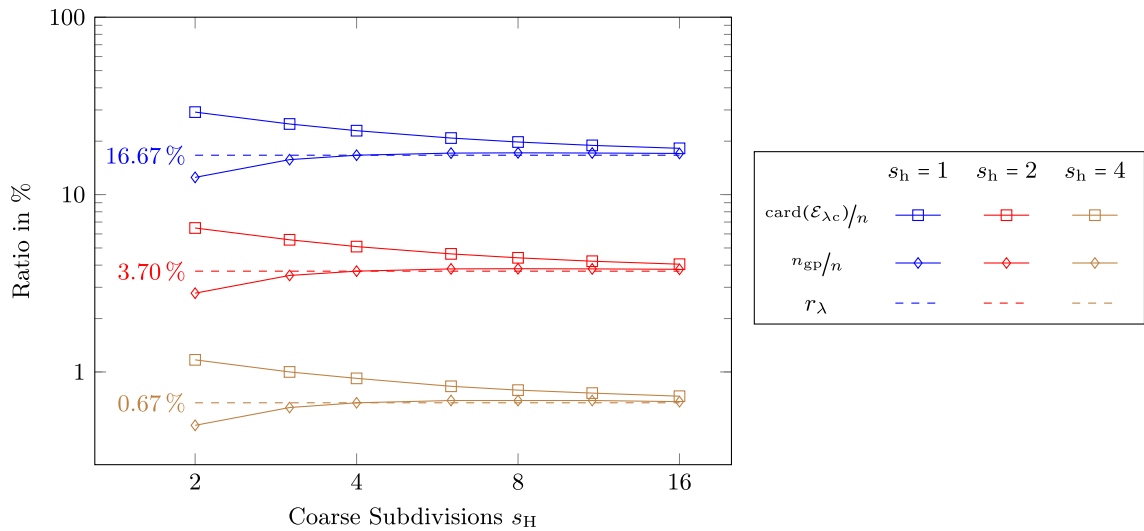


Fig. A.10. Ratio of cotree DOFs on wire basket card ($\mathcal{E}_{\lambda c}$) and primal DOFs n_{gp} with respect to n for varying s_h and s_H .

for the edges and nodes of the corresponding wire basket graph $\mathcal{G}_\lambda^{(i)}$. If we look at the broader picture with $N = s_H^3$ subdomains, we can compute the number of macroscopic edges by

$$m_e = \text{card}(\mathcal{E})|_{s_h=1} = 3s_H s_H (s_H s_H + 1)^2|_{s_h=1} = 3s_H (s_H + 1)^2,$$

which implies that

$$\text{card}(\mathcal{E}_\lambda) = m_e s_h = 3s_H s_H (s_H + 1)^2 \tag{A.1}$$

is the number of edges on \mathcal{G}_λ . Correspondingly, we can state

$$\text{card}(\mathcal{N}_\lambda) = m_e (s_h + 1) - \alpha m_n = m_e s_h + \underbrace{(m_e - \alpha)m_n}_{=c} = \text{card}(\mathcal{E}_\lambda) + c \tag{A.2}$$

for the number of nodes on the global wire basket where α and c are appropriate constants. For the global number of endpoints

$$m_n = \text{card}(\mathcal{N})|_{s_h=1} = (s_H s_H + 1)^3|_{s_h=1} = (s_H + 1)^3$$

can be calculated. The constant α only depends on how many edges are connected to each respective endpoint, which is quite technical to compute but clearly independent of the local discretization. To compute c , we can exploit that $\text{card}(\mathcal{N}_\lambda) = m_n$ needs to be valid for $s_h = 1$. This results in

$$m_n = \text{card}(\mathcal{N}_\lambda) = m_e - c \Leftrightarrow c = 3s_H (s_H + 1)^2 - (s_H + 1)^3 = 2s_H^3 + 3s_H^2 - 1$$

and

$$\text{card}(\mathcal{E}_{\lambda c}) = c + 1 = 2s_H^3 + 3s_H^2 \tag{A.3}$$

for the number of cotree elements on the wire basket. Then, the limit for the ratio n_λ/n is given by

$$r_\lambda := \lim_{s_H \rightarrow \infty} \frac{\text{card}(\mathcal{E}_{\lambda c})}{n} = \lim_{s_H \rightarrow \infty} \left(\frac{2s_H^3 + 3s_H^2}{3s_H^3 s_h (s_h + 1)^2} \right) = \frac{2}{3s_h (s_h + 1)^2}, \tag{A.4}$$

where

$$n = N \text{card}(\mathcal{E}^{(i)}) = 3s_H^3 s_h (s_h + 1)^2$$

is valid. Due to the relation $\text{card}(\mathcal{E}_{\lambda c}) \geq n_{gp}$, we know that n_{gp}/n is bounded by the respective ratio with $\text{card}(\mathcal{E}_{\lambda c})$. Hence, r_λ should be a good indicator for n_{gp}/n if s_H is large. The explained ratios are visualized in Fig. A.10 for different combinations of s_h and s_H where we can observe all expected properties. The estimated ratio r_λ is close to n_{gp}/n , therefore giving a good indication of the coarse problem size which can even be computed before assembling the discrete problem. We conclude that the coarse problem is only a fraction of the overall problem if the local systems are sufficiently refined.

Data availability

Code will be published as open source when paper is accepted or made available on request by a reviewer.

References

- [1] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Comput. Methods Appl. Mech. Engrg.* (ISSN: 0045-7825) 194 (2005) 4135–4195, <http://dx.doi.org/10.1016/j.cma.2004.10.008>.
- [2] Z. Bontinck, J. Corno, S. Schöps, H. De Gersem, Isogeometric analysis and harmonic stator-rotor coupling for simulating electric machines, *Comput. Methods Appl. Mech. Engrg.* (ISSN: 0045-7825) 334 (2018) 40–55, <http://dx.doi.org/10.1016/j.cma.2018.01.047>.
- [3] M. Merkel, B. Kapidani, S. Schöps, R. Vázquez, Torque computation with the isogeometric mortar method for the simulation of electric machines, *IEEE Trans. Magn.* (ISSN: 0018-9464) 58 (9) (2022) <http://dx.doi.org/10.1109/TMAG.2022.3186247>.
- [4] A. Buffa, G. Sangalli, R. Vázquez, Isogeometric analysis in electromagnetics: B-splines approximation, *Comput. Methods Appl. Mech. Engrg.* (ISSN: 0045-7825) 199 (2010) 1143–1152, <http://dx.doi.org/10.1016/j.cma.2009.12.002>.
- [5] A. Buffa, R. Vázquez, G. Sangalli, L. Beirão da Veiga, Approximation estimates for isogeometric spaces in multipatch geometries, *Numer. Methods Partial Differential Equations* 31 (2) (2015) 422–438, <http://dx.doi.org/10.1002/num.21943>.
- [6] A. Buffa, J. Dölz, S. Kurz, S. Schöps, R. Vázquez, F. Wolf, Multipatch approximation of the de Rham sequence and its traces in isogeometric analysis, *Numer. Math.* (ISSN: 0029-599X) 144 (1) (2019) 201–236, <http://dx.doi.org/10.1007/s00211-019-01079-x>.
- [7] A. Toselli, O.B. Widlund, *Domain Decomposition Methods – Algorithms and Theory*, in: *Springer Series in Computational Mathematics*, vol. 34, Springer, ISBN: 978-3-540-20696-5, 2005.
- [8] C. Farhat, F.-X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, *Internat. J. Numer. Methods Engrg.* (ISSN: 0029-5981) 32 (6) (1991) 1205–1227, <http://dx.doi.org/10.1002/nme.1620320604>.
- [9] S.K. Kleiss, C. Pechstein, B. Jüttler, S. Tomar, IETI – isogeometric tearing and interconnecting, *Comput. Methods Appl. Mech. Engrg.* (ISSN: 0045-7825) 247 (2012) 201–215, <http://dx.doi.org/10.1016/j.cma.2012.08.007>.
- [10] R. Bouclier, T. Hirschler, *IGA: Non-Conforming Coupling and Shape Optimization of Complex Multipatch Structures*, vol. 1, ISTE Ltd and John Wiley & Sons, 2022.
- [11] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, D. Rixen, FETI-DP: a dual-primal unified FETI method—part I: A faster alternative to the two-level FETI method, *Internat. J. Numer. Methods Engrg.* (ISSN: 0029-5981) 50 (7) (2001) 1523–1544, <http://dx.doi.org/10.1002/nme.76>.
- [12] A. Klawonn, O.B. Widlund, M. Dryja, Dual-primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients, *SIAM J. Numer. Anal.* 40 (1) (2002) 159–179, <http://dx.doi.org/10.1137/S0036142901388081>.
- [13] Z. Dostál, D. Horák, R. Kučera, Total FETI—an easier implementable variant of the FETI method for numerical solution of elliptic PDE, *Comm. Numer. Methods Engrg.* 22 (12) (2006) 1155–1162, <http://dx.doi.org/10.1002/cnm.881>.
- [14] G. Of, O. Steinbach, The all-floating boundary element tearing and interconnecting method, *J. Numer. Math.* 17 (4) (2009) <http://dx.doi.org/10.1515/jnum.2009.014>.
- [15] M. Bony, M. Montardini, G. Sangalli, M. Tani, A domain decomposition method for isogeometric multi-patch problems with inexact local solvers, *Comput. Math. Appl.* (ISSN: 0898-1221) 80 (11) (2020) 2604–2621, <http://dx.doi.org/10.1016/j.camwa.2020.08.024>.
- [16] Y.-J. Li, J.-M. Jin, Parallel implementation of the FETI-DPEM algorithm for general 3D EM simulations, *J. Comput. Phys.* (ISSN: 0021-9991) 228 (9) (2009) 3255–3267, <http://dx.doi.org/10.1016/j.jcp.2009.01.029>.
- [17] D. Marcsa, M. Kuczmann, Finite element tearing and interconnecting method and its algorithms for parallel solution of magnetic field problems, *Electr., Control. Commun. Eng.* 3 (1) (2013) 25–30, <http://dx.doi.org/10.2478/ecce-2013-0011>.
- [18] M.I. Ghenai, R. Perrussel, O. Chadebec, F. Vi, J.-M. Guichon, G. Meunier, J. Siau, Domain decomposition for 3-D nonlinear magnetostatic problems: Newton–Krylov–Schur versus Schur–Newton–Krylov methods, *IEEE Trans. Magn.* (ISSN: 0018-9464) 60 (3) (2024) 1–4, <http://dx.doi.org/10.1109/TMAG.2023.3299989>.
- [19] R. Schneckelmeier, S. Takacs, IETI-DP methods for discontinuous Galerkin multi-patch isogeometric analysis with T-junctions, *Comput. Methods Appl. Mech. Engrg.* (ISSN: 0045-7825) 393 (2022) 114694, <http://dx.doi.org/10.1016/j.cma.2022.114694>.
- [20] A. Toselli, Dual-primal FETI algorithms for edge finite-element approximations in 3D, *IMA J. Numer. Anal.* (ISSN: 0272-4979) 26 (1) (2006) 96–130, <http://dx.doi.org/10.1093/imanum/dri023>.
- [21] J. Keränen, J. Pippuri, M. Malinen, J. Ruokolainen, P. Råback, M. Lyly, K. Tammi, Efficient parallel 3-D computation of electrical machines with Elmer, *IEEE Trans. Magn.* (ISSN: 0018-9464) 51 (3) (2015) 1–4, <http://dx.doi.org/10.1109/TMAG.2014.2356256>.
- [22] W. Yao, J.-M. Jin, P.T. Krein, Analysis of electromechanical problems using the dual-primal finite element tearing and interconnecting method, in: 2012 IEEE Power and Energy Conference At Illinois, 2012, pp. 1–5, <http://dx.doi.org/10.1109/PECI.2012.6184595>.
- [23] W. Yao, J.-M. Jin, P.T. Krein, A highly efficient domain decomposition method applied to 3-D finite-element analysis of electromechanical and electric machine problems, *IEEE Trans. Energy Convers.* (ISSN: 0885-8969) 27 (4) (2012) 1078–1086, <http://dx.doi.org/10.1109/TEC.2012.2216528>.
- [24] R. Albanese, G. Rubinacci, Integral formulation for 3D eddy-current computation using edge elements, *IEE Proc. Sci. Meas. Tech.* (ISSN: 1350-2344) 135 (7) (1988) 457–462, <http://dx.doi.org/10.1049/ip-a-1:19880072>.
- [25] I. Munteanu, Tree-cotree condensation properties, *ICS Newsllett.* (ISSN: 1026-0854) 9 (2002) 10–14, URL <https://www.compumag.org/wp/newsletter/>.
- [26] B. Kapidani, M. Merkel, S. Schöps, R. Vázquez, Tree-cotree decomposition of isogeometric mortared spaces in $H(\text{curl})$ on multi-patch domains, *Comput. Methods Appl. Mech. Engrg.* (ISSN: 0045-7825) 395 (2022) 114949, <http://dx.doi.org/10.1016/j.cma.2022.114949>.
- [27] F. Rapetti, A. Alonso Rodríguez, E. De Los Santos, On the tree gauge in magnetostatics, *J.* (ISSN: 2571-8800) 5 (1) (2022) 52–63, <http://dx.doi.org/10.3390/j5010004>.
- [28] R. Vázquez, A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0, *Comput. Math. Appl.* (ISSN: 0898-1221) 72 (3) (2016) 523–554, <http://dx.doi.org/10.1016/j.camwa.2016.05.010>.
- [29] S.J. Salon, *Finite Element Analysis of Electrical Machines*, Kluwer, 1995.
- [30] J.D. Jackson, *Classical Electrodynamics*, third ed., Wiley & Sons, New York, NY, USA, ISBN: 978-0-471-30932-1, 1998, <http://dx.doi.org/10.1017/CBO9780511760396>.
- [31] D.N. Arnold, *Finite Element Exterior Calculus*, SIAM, ISBN: 978-1-61197-553-6, 2018, <http://dx.doi.org/10.1137/1.9781611975543>.
- [32] L. Kettunen, K. Forsman, A. Bossavit, Gauging in Whitney spaces, *IEEE Trans. Magn.* (ISSN: 0018-9464) 35 (3) (1999) 1466–1469, <http://dx.doi.org/10.1109/20.767243>.
- [33] P. Monk, *Finite Element Methods for Maxwell’s Equations*, Oxford University Press, Oxford, 2003.
- [34] L. Formaggia, F. Saleri, A. Veneziani, *Solving Numerical PDEs: Problems, Applications, Exercises*, Springer, ISBN: 978-88-470-2412-0, 2012, <http://dx.doi.org/10.1007/978-88-470-2412-0>.
- [35] A. Alonso Rodríguez, F. Rapetti, Small trees for high order Whitney elements, in: *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2018*, in: *Lecture Notes in Computational Science and Engineering*, Springer International Publishing, ISBN: 978-3-030-39647-3, 2020, pp. 587–597, <http://dx.doi.org/10.1007/978-3-030-39647-3>.
- [36] L. Beirão da Veiga, A. Buffa, G. Sangalli, R. Vázquez, Mathematical analysis of variational isogeometric methods, *Acta. Num.* (ISSN: 1474-0508) 23 (2014) 157–287, <http://dx.doi.org/10.1017/s096249291400004x>.
- [37] B. Kapidani, M. Merkel, S. Schöps, R. Vázquez, Arbitrary order spline representation of cohomology generators for isogeometric analysis of eddy current problems, *Adv. Comput. Math.* (ISSN: 1019-7168) 50 (92) (2024) <http://dx.doi.org/10.1007/s10444-024-10181-0>.

- [38] A. Quarteroni, A. Valli, *Domain Decomposition Methods for Partial Differential Equations*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 1999.
- [39] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, *Acta. Num.* (ISSN: 1474-0508) 14 (1) (2005) 1–137, <http://dx.doi.org/10.1017/S0962492904000212>.
- [40] J.B. Manges, Z.J. Cendes, A generalized tree-cotree gauge for magnetic field computation, *IEEE Trans. Magn.* (ISSN: 0018-9464) 31 (3) (1995) 1342–1347, <http://dx.doi.org/10.1109/20.376275>.
- [41] R. Albanese, G. Rubinacci, Solution of three dimensional eddy current problems by integral and differential methods, *IEEE Trans. Magn.* (ISSN: 0018-9464) 24 (1) (1988) 98–101, <http://dx.doi.org/10.1109/20.43865>.
- [42] J.B. Kruskal, On the shortest spanning subtree of a graph and the traveling salesman problem, *PROC* (ISSN: 1088-6826) 7 (1) (1956) 48–50, <http://dx.doi.org/10.2307/2033241>.
- [43] T.H. Cormen, C.E. Leiserson, R.L. Rivest, *Introduction to Algorithms*, MIT Press, ISBN: 978-0262032933, 2001.
- [44] P. Dular, A. Nicolet, A. Genon, W. Legros, A discrete sequence associated with mixed finite elements and its gauge condition for vector potentials, *IEEE Trans. Magn.* (ISSN: 0018-9464) 31 (3) (1995) 1356–1359, <http://dx.doi.org/10.1109/20.376278>.
- [45] A. Hatcher, *Algebraic Topology*, Cambridge University Press, New York, ISBN: 978-0-521-79540-1, 2001.
- [46] C. Farhat, M. Lesoinne, K. Pierson, A scalable dual-primal domain decomposition method, *Numer. Linear Algebra Appl.* 7 (7–8) (2000) 687–714, [http://dx.doi.org/10.1002/1099-1506\(200010/12\)7:7/8<687::AID-NLA219>3.0.CO;2-S](http://dx.doi.org/10.1002/1099-1506(200010/12)7:7/8<687::AID-NLA219>3.0.CO;2-S).
- [47] Mathworks, *MATLAB® release notes, 2023, 9.14 (R2023a)*.
- [48] W.L. Oberkampf, C.J. Roy, *Verification and Validation in Scientific Computing*, Cambridge University Press, New York, NY, USA, ISBN: 978-0-521-11360-1, 2010.
- [49] J. Todd, The condition of a certain matrix, *Math. Proc. Cambridge Philos. Soc.* 46 (1) (1950) 116–118, <http://dx.doi.org/10.1017/S0305004100025536>.
- [50] Y. Saad, *Iterative Methods for Sparse Linear Systems*, second ed., Society for Industrial and Applied Mathematics, Boston, MA, USA, ISBN: 978-0-89871-534-7, 2003, URL <http://www-users.cs.umn.edu/~saad/books.html>.
- [51] P. Dłotko, B. Kapidani, R. Specogna, Topoprocessor: An efficient computational topology toolbox for h-oriented eddy current formulations, *IEEE Trans. Magn.* (ISSN: 0018-9464) 53 (6) (2017) 1–4, <http://dx.doi.org/10.1109/tmag.2017.2661480>.
- [52] G.M. Amdahl, Validity of the single processor approach to achieving large scale computing capabilities, in: *Proceedings of the April 18-20, 1967, Spring Joint Computer Conference*, in: *AFIPS '67 (Spring)*, Association for Computing Machinery, Atlantic City, New Jersey, ISBN: 9781450378956, 1967, pp. 483–485, <http://dx.doi.org/10.1145/1465482.1465560>.
- [53] A. Klawonn, O. Rheinbach, Highly scalable parallel domain decomposition methods with an application to biomechanics, *ZAMM* (ISSN: 1521-4001) 90 (1) (2010) 5–32, <http://dx.doi.org/10.1002/zamm.200900329>.
- [54] J.L. Gustafson, Reevaluating Amdahl's law, *CACM* (ISSN: 0001-0782) 31 (5) (1988) 532–533, <http://dx.doi.org/10.1145/42411.42415>.
- [55] J. Mandel, R. Tezaur, On the convergence of a dual-primal substructuring method, *Numer. Math.* (ISSN: 0029-599X) 88 (3) (2001) 543–558, <http://dx.doi.org/10.1007/s211-001-8014-1>.
- [56] A. Klawonn, O. Rheinbach, Inexact FETI-DP methods, *Internat. J. Numer. Methods Engrg.* (ISSN: 0029-5981) 69 (2) (2007) 284–307, <http://dx.doi.org/10.1002/nme.1758>.