

ARBITRARY ORDER APPROXIMATIONS AT CONSTANT COST FOR TIMOSHENKO BEAM NETWORK MODELS

MORITZ HAUCK^{1,*}, AXEL MÅLQVIST² AND ANDREAS RUPP³

Abstract. This paper considers the numerical solution of Timoshenko beam network models, comprised of Timoshenko beam equations on each edge of the network, which are coupled at the nodes of the network using rigid joint conditions. Through hybridization, we can equivalently reformulate the problem as a symmetric positive definite system of linear equations posed on the network nodes. This is possible since the nodes, where the beam equations are coupled, are zero-dimensional objects. To discretize the beam network model, we propose a hybridizable discontinuous Galerkin method that can achieve arbitrary orders of convergence under mesh refinement without increasing the size of the global system matrix. As a preconditioner for the typically very poorly conditioned global system matrix, we employ a two-level overlapping additive Schwarz method. We prove uniform convergence of the corresponding preconditioned conjugate gradient method under appropriate connectivity assumptions on the network. Numerical experiments support the theoretical findings of this work.

Mathematics Subject Classification. 05C50, 65F10, 65N15, 65N30, 65N55.

Received August 6, 2024. Accepted September 30, 2025.

1. INTRODUCTION

Many applications in science and engineering involve geometrically complex structures composed of slender, effectively one-dimensional objects. Examples include blood vessels [20], porous materials [10], or fiber-based materials [31]. For such problems, resolving all microscopic details in a three-dimensional computer simulation can be computationally very demanding. Therefore, in many cases, it seems appropriate to describe the geometry by a spatial network represented by a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ of nodes and edges which is embedded into a bounded domain $\Omega \subset \mathbb{R}^3$. The resulting spatial network model then involves one-dimensional differential equations on each edge, coupled by algebraic constraints at the nodes of the graph. In the following, we consider the elastic deformation of fiber-based materials, such as paper or cardboard, as a model problem. The spatial network underlying this problem is constructed as follows: Nodes are placed at the intersections of fibers, and an edge connects two nodes if a fiber is connecting them. Depending on the intersection area of the two fibers, additional

Keywords and phrases. Timoshenko beam network, elastic graph, hybridizable discontinuous Galerkin, arbitrary order approximation, *a priori* error analysis, additive Schwarz preconditioner.

¹ Institute for Applied and Numerical Mathematics, Karlsruhe Institute of Technology, Englerstr. 2, 76131 Karlsruhe, Germany.

² Department of Mathematical Sciences, University of Gothenburg and Chalmers University of Technology, 41296 Göteborg, Sweden.

³ Department of Mathematics, Saarland University, 66123 Saarbrücken, Germany.

*Corresponding author: moritz.hauck@kit.edu

nodes and edges may be added to strengthen the connection between the fibers. To obtain a spatial network model, we equip each edge with a Timoshenko beam model, cf. [5, 51], describing the elastic deformation of the corresponding fiber. A well-posed problem is then obtained by enforcing the continuity of displacements and rotations and the balance of forces and moments at the nodes of the spatial network, cf. [34]. For more details on this model, particularly the construction of the graph representing the geometric structure of cardboard, we refer to [21, 23, 31].

For practically simulating spatial network models, one needs to perform a discretization of the one-dimensional Timoshenko beam equations posed on the edges. A popular approach for this are beam finite elements, which differ mainly in the number and placement of the degrees of freedom; see, *e.g.*, [16, 30, 35, 50]. When discretizing the Timoshenko beam equations using beam elements, one often observes a shear-locking effect, which leads to an underestimation of the displacements. This effect is caused by underresolution and occurs mainly for beam elements with few degrees of freedom; see, *e.g.*, the theoretical study in [43]. Shear locking can be avoided by increasing the local degrees of freedom by subdividing the edges and using multiple beam elements, or by using higher-order beam elements. Note that shear locking can also be reduced by using numerical tricks such as under-integration, cf. [46, 56]. An alternative to beam elements are so-called analytical ansatz functions, which are derived by analytically solving the Timoshenko beam equations on each edge; see, *e.g.*, [29, 33, 47]. However, to find an explicit expression for the analytical solution, it is usually necessary to impose additional modeling assumptions, such as that the material coefficients are constant or that the distributed loads and moments are homogeneous or constant.

In this paper, we apply a hybridizable discontinuous Galerkin (HDG) method to discretize the spatial network model under consideration. The use of such a discretization is motivated by the study in [49], where the authors rigorously define and discretize diffusion-type problems on networks of hypersurfaces. There, the authors claim that a hybrid formulation is natural for partial differential equations on hypergraphs and, by extension, they should be discretized using HDG methods. In the special case of graphs, the nodes at which the one-dimensional differential equations are coupled are zero-dimensional objects, whence the spatial network problem can be equivalently reformulated as a symmetric positive definite system of linear equations posed on the nodes of the network. For such problems, an HDG discretization can achieve arbitrary convergence orders without increasing the number of globally coupled degrees of freedom. Following the paradigm of HDG methods that local solves are essentially for free since the corresponding problems are small and they can be solved in parallel, cf. [13], the proposed method can achieve arbitrary orders of convergence at (almost) constant computational cost. We perform an *a priori* error analysis of the HDG method, where we prove optimal convergence orders under mesh refinement. Note that the shear-locking effect of classical beam elements can be easily avoided by using sufficiently high polynomial degrees.

Due to the complex geometry of the spatial network and possibly highly varying material coefficients, the linear system of equations obtained by the HDG method is typically very poorly conditioned. Numerical experiments demonstrate that standard black-box preconditioners, such as many algebraic multigrid variants (see, *e.g.*, the review article [55]), can typically not significantly speed up convergence. This is because they do not sufficiently consider the geometry of the underlying problem. Also, preconditioners for HDG methods such as [15, 18, 37–39, 53] are not suitable for the present application since they require a coarsening strategy that, in the spatial network setting, would change the geometry of the underlying graph. Therefore, the injection operators defined therein cannot be used directly. To overcome these difficulties, this paper employs a preconditioner which is based on the observation that the network can be treated essentially as a continuous object at sufficiently coarse scales. This allows one to place an artificial coarse mesh over the network and use finite element techniques with respect to this mesh to introduce, *e.g.*, a two-level overlapping Schwarz preconditioner similar to [22]. We prove that the global system matrix is spectrally equivalent to an edge length weighted graph Laplacian (in each component). Under certain network connectivity assumptions, this then allows us to prove the uniform convergence of the corresponding preconditioned conjugate gradient method.

Note that, alternatively, multiscale methods such as the (Super-)Localized Orthogonal Decomposition (cf. [19, 25, 28, 41]) could be used to tackle the problem of large and ill-conditioned linear systems of equations, see [17,



FIGURE 1. Fiber network model of paper at the millimeter scale.

[24, 27]. These methods feature a compression property that allow one to significantly reduce the size of the linear systems of equations to be solved, making them tractable again, *e.g.*, by classical sparse direct solvers. Also the use of other multiscale methods such as the Multiscale Spectral Generalized Finite Element Method [2, 40] or Gambles [44, 45] seems possible for spatial network models. For a comprehensive overview of multiscale methods, see also [1, 42].

The rest of this paper is organized as follows: In Section 2, we introduce a Timoshenko beam network model, which is then discretized using a HDG method in Section 3. An *a priori* error analysis of the method is given in Section 4, while an efficient preconditioner is presented in Section 5. The paper concludes with some numerical experiments in Section 6.

2. TIMOSHENKO BEAM NETWORK MODEL

In this section, we derive a Timoshenko beam network model that describes the elastic deformation of fiber-based materials such as paper or cardboard. The beam network model considers a finite number of initially straight beams whose cross sections are constant along the length of the beams. Note that beams are slender objects, *i.e.*, their axial dimension is much larger than their cross sectional diameter. This motivates to represent the network of beams by a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{\mathbf{n}_1, \mathbf{n}_2, \dots, \mathbf{n}_K\}$ is a set of zero-dimensional nodes and $\mathcal{E} = \{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_L\}$ is a set of locally one-dimensional edges. We assume that the graph \mathcal{G} is connected, which is natural, since otherwise its connected components could be considered separately. In the following, a three-dimensional setting is considered, *i.e.*, the nodes and edges of \mathcal{G} are contained in \mathbb{R}^3 . Each edge connects two different nodes, and we write $\mathbf{n} \sim \mathbf{e}$ if the node $\mathbf{n} \in \mathcal{N}$ is an endpoint of the edge $\mathbf{e} \in \mathcal{E}$. In the present paper application, the nodes $\mathbf{n} \in \mathcal{N}$ model the joints of the beams, whereas the edges $\mathbf{e} \in \mathcal{E}$ represent the fiber segments connecting two nodes; we refer to Figure 1 for an illustration of a fiber network model of paper. To describe the deformation of the beam network in response to applied forces and moments, we use the Timoshenko beam theory originally developed in [51]; see also [5]. In contrast to Euler–Bernoulli beam theory (see, *e.g.*, [3]), it can accurately capture shear effects.

2.1. Governing equations

The derivation of Timoshenko beam theory is based on the laws of linear elasticity. It relies on the assumption that the cross sections of a beam are infinitely rigid in their plane and remain plain after deformation. Furthermore, it is always assumed that the cross sections can rotate independently from the deformation of the centroid line. Under these assumptions, Timoshenko beam theory states for each beam one-dimensional differential equations describing its deformation. The deformation of the beam corresponding to edge $\mathbf{e} \in \mathcal{E}$ is described by the beam's centerline displacement $\mathbf{u}_{\mathbf{e}}: \mathbf{e} \rightarrow \mathbb{R}^3$ and its cross section rotation $\mathbf{r}_{\mathbf{e}}: \mathbf{e} \rightarrow \mathbb{R}^3$. These variables will serve as primal unknowns in the Timoshenko beam equations. The corresponding dual unknowns

are $\mathbf{n}_\epsilon: \epsilon \rightarrow \mathbb{R}^3$, the stress resultants from normal and shear forces, and $\mathbf{m}_\epsilon: \epsilon \rightarrow \mathbb{R}^3$, the resultant moment from torsion and bending moments. Denoting by \mathbf{n}_k and \mathbf{n}_ℓ with $k < \ell$ the endpoints of the edge ϵ , we can define the unit vector aligned with ϵ as $\mathbf{i}_\epsilon := (\mathbf{n}_\ell - \mathbf{n}_k)/h_\epsilon$, where $h_\epsilon := |\mathbf{n}_\ell - \mathbf{n}_k|$ denotes the length of ϵ and $|\cdot|$ is the Euclidean norm. We further define the (scalar) unit normals of the edge ϵ by $\nu_\epsilon(\mathbf{n}_k) := -1$ and $\nu_\epsilon(\mathbf{n}_\ell) := +1$.

The properties of the beams are determined by their material and shape parameters, which are encoded into the coefficients C_n and C_m ; see also Remark 2.1. In the following, we assume that C_n and C_m are symmetric $\mathbb{R}^{3 \times 3}$ -valued functions which satisfy uniform lower and upper spectral bounds, *i.e.*, there exist $0 < \alpha_n, \alpha_m, \beta_n, \beta_m < \infty$ such that

$$\alpha_n |\xi|^2 \leq (C_n(x)\xi) \cdot \xi \leq \beta_n |\xi|^2, \quad \alpha_m |\xi|^2 \leq (C_m(x)\xi) \cdot \xi \leq \beta_m |\xi|^2 \quad (2.1)$$

holds for all $x \in \epsilon$ and $\epsilon \in \mathcal{E}$.

Given the distributed force $\mathbf{f}_\epsilon: \epsilon \rightarrow \mathbb{R}^3$ and the distributed moment $\mathbf{g}_\epsilon: \epsilon \rightarrow \mathbb{R}^3$, the one-dimensional Timoshenko beam equations corresponding to edge $\epsilon \in \mathcal{E}$ read

$$-C_n(\partial_x \mathbf{u}_\epsilon + \mathbf{i}_\epsilon \times \mathbf{r}_\epsilon) = \mathbf{n}_\epsilon, \quad -C_m \partial_x \mathbf{r}_\epsilon = \mathbf{m}_\epsilon, \quad (2.2a)$$

$$\partial_x \mathbf{n}_\epsilon = \mathbf{f}_\epsilon, \quad \partial_x \mathbf{m}_\epsilon + \mathbf{i}_\epsilon \times \mathbf{n}_\epsilon = \mathbf{g}_\epsilon, \quad (2.2b)$$

where ∂_x denotes differentiation with respect to the spatial variable x that varies along ϵ with unit speed and \times denotes the cross product. We impose continuity and balance conditions at the nodes to couple the one-dimensional differential equations on each edge. The *continuity conditions* enforce that the displacements and rotations at each node coincide. Following the concept of hybridization, this can be reformulated by requiring that for each node $\mathbf{n} \in \mathcal{N}$ there exist vectors $\mathbf{u}_\mathbf{n}, \mathbf{r}_\mathbf{n} \in \mathbb{R}^3$ such that

$$\mathbf{u}_\epsilon(\mathbf{n}) = \mathbf{u}_\mathbf{n}, \quad \mathbf{r}_\epsilon(\mathbf{n}) = \mathbf{r}_\mathbf{n} \quad (2.2c)$$

holds for all edges $\epsilon \in \mathcal{E}$ adjacent to \mathbf{n} . The nodewise unknowns $\mathbf{u}_\mathbf{n}$ and $\mathbf{r}_\mathbf{n}$ will act as Lagrange multipliers in a hybrid formulation of the problem. We also prescribe Dirichlet boundary conditions at a given set of Dirichlet nodes $\emptyset \neq \mathcal{N}_D \subset \mathcal{N}$. More precisely, for each node $\mathbf{n} \in \mathcal{N}_D$ and given Dirichlet data $\mathbf{u}_\mathbf{n}^D, \mathbf{r}_\mathbf{n}^D \in \mathbb{R}^3$, we require that

$$\mathbf{u}_\mathbf{n} = \mathbf{u}_\mathbf{n}^D, \quad \mathbf{r}_\mathbf{n} = \mathbf{r}_\mathbf{n}^D. \quad (2.2d)$$

The *balance conditions* ensure an equilibrium of forces and moments at non-Dirichlet nodes. This means that for all $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D$ and given concentrated forces and moments $\mathbf{f}_\mathbf{n}, \mathbf{g}_\mathbf{n} \in \mathbb{R}^3$, which are typically set to zero, it should hold that

$$-[\![\mathbf{n}_\epsilon \nu_\epsilon]\!]_\mathbf{n} = -\mathbf{f}_\mathbf{n}, \quad -[\![\mathbf{m}_\epsilon \nu_\epsilon]\!]_\mathbf{n} = -\mathbf{g}_\mathbf{n}, \quad (2.2e)$$

where the summation operator $[\![\cdot]\!]_\mathbf{n}$ sums over all values that are attained at \mathbf{n} . This notation is a generalization of the jump operator in the context of discontinuous Galerkin methods; see, *e.g.*, equation (2.10) of [49]. The multiplication of equation (2.2e) with -1 is purely artificial at this point, but it will allow us later to deal with positive definite instead of negative definite operators. In the engineering literature, nodes subject to conditions (2.2c) and (2.2e) are typically called *rigid joints*, and elastic structures with rigid joints are sometimes called *frames*. Problem (2.2) defines a frame of Timoshenko beams.

Remembering that \mathcal{G} is connected, there is a unique solution to (2.2), since there is at least one Dirichlet node.

Remark 2.1 (Local formulation). In the literature, problem (2.2) is typically formulated in local coordinates, which requires a change of basis between local and global coordinates. As local basis corresponding to edge $\epsilon \in \mathcal{E}$ we consider $\{\mathbf{i}_\epsilon, \mathbf{j}_\epsilon, \mathbf{k}_\epsilon\}$, where \mathbf{i}_ϵ is as above, and \mathbf{j}_ϵ and \mathbf{k}_ϵ are chosen as the principal axes of inertia of the beam's cross section. As global basis the canonical basis of \mathbb{R}^3 is chosen, which we denote by $\{\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}\}$. We assume that $\{\mathbf{i}_\epsilon, \mathbf{j}_\epsilon, \mathbf{k}_\epsilon\}$ is right-handed, *i.e.*, it holds that $\mathbf{i}_\epsilon = \mathbf{j}_\epsilon \times \mathbf{k}_\epsilon$. The change-of-basis matrix for the beam ϵ is then given by $\mathbf{T}_\epsilon := (\mathbf{i}_\epsilon, \mathbf{j}_\epsilon, \mathbf{k}_\epsilon) \in \mathbb{R}^{3 \times 3}$. Since the basis is right-handed and orthonormal, it holds that

$\det(\mathbf{T}_\epsilon) = 1$. Multiplying (2.2a) and (2.2b) from the left by \mathbf{T}_ϵ^\top and using that \mathbf{T}_ϵ is orthogonal, we arrive at the local problem

$$-\hat{C}_n(\partial_{\hat{x}}\hat{\mathbf{u}}_\epsilon + \hat{\mathbf{i}} \times \hat{\mathbf{r}}_\epsilon) = \hat{\mathbf{n}}_\epsilon, \quad -\hat{C}_m\partial_{\hat{x}}\hat{\mathbf{r}}_\epsilon = \hat{\mathbf{m}}_\epsilon, \quad (2.3a)$$

$$\partial_{\hat{x}}\hat{\mathbf{n}}_\epsilon = \hat{\mathbf{f}}_\epsilon, \quad \partial_{\hat{x}}\hat{\mathbf{m}}_\epsilon + \hat{\mathbf{i}} \times \hat{\mathbf{n}}_\epsilon = \hat{\mathbf{g}}_\epsilon, \quad (2.3b)$$

where local variables are indicated by hats. This problem is posed on $[0, h_\epsilon] \times \{0\}^2$. Note that due to the choice of local basis, the coefficient matrices $\hat{C}_n(\hat{x}) = \mathbf{T}_\epsilon^\top C_n(x) \mathbf{T}_\epsilon$ and $\hat{C}_m(\hat{x}) = \mathbf{T}_\epsilon^\top C_m(x) \mathbf{T}_\epsilon$ are diagonal for all $\hat{x} \in [0, h_\epsilon] \times \{0\}^2$ and the components of $\hat{\mathbf{u}}_\epsilon$ and $\hat{\mathbf{r}}_\epsilon$ decouple. More precisely, we have the characterization

$$\hat{C}_n = \text{diag}(EA, kGA, kGA), \quad \hat{C}_m = \text{diag}(GI_t, EI_2, EI_3),$$

where E is the elastic modulus, G is the shear modulus, A is the cross section area, kA is the corrected shear area, I_2 and I_3 are the second moments of inertia of the cross section and I_t is the polar moment of the cross section. Note that the local formulation (2.3) is essential for implementing Timoshenko beam networks.

Remark 2.2 (Other HDG discretizations of Timoshenko beams). Our Timoshenko beam model is related to, but not identical with, the Timoshenko beam model whose HDG discretization is discussed in [6–9]. Those works consider a single beam (not a network) undergoing bending in one direction, neglecting other spatial dimensions. In our notation and omitting physical constants, the model reads:

$$-\partial_{\hat{x}}\hat{u} - \hat{r} = \hat{n}, \quad -\partial_{\hat{x}}\hat{r} = \hat{m}, \quad \partial_{\hat{x}}\hat{n} = \hat{f}, \quad \partial_{\hat{x}}\hat{m} = \hat{n}.$$

If compared to (2.3), these equations lack the cross products and a summand in the fourth equation. Nonetheless the discretization in [6–9] serves as an inspiration for our HDG scheme. However, we do not follow the post-processing approach in [6] (although it can enhance the accuracy of our simulations) as we claim that the polynomial degree does (almost) not influence the computational cost.

3. HDG DISCRETIZATION

This section derives an HDG method for the numerical solution of the Timoshenko beam network model introduced in the previous section. To simplify the presentation, the following remark introduces a notation to hide constants in estimates independent of the edge length h_ϵ and the stabilization parameter τ_ϵ (to be introduced in Sect. 3.2).

Remark 3.1 (Tilde notation). If $a \leq Cb$, where $C > 0$ is a constant that may depend on the coefficients C_n and C_m , the seminorms of the solutions \mathbf{u}_ϵ , \mathbf{r}_ϵ , \mathbf{n}_ϵ , and \mathbf{m}_ϵ , but is independent of the edge length h_ϵ and the HDG stabilization parameter τ_ϵ (defined in Sect. 3.2), we write $a \lesssim b$ to hide the constant. Similarly, we write $a \gtrsim b$ for $a \geq Cb$.

3.1. Hybrid dual mixed formulation

In the following, we present the hybrid dual-mixed formulation of the Timoshenko beam network model, which will serve as the starting point for deriving the HDG method. It is based on the function spaces $V_u^\epsilon := (L^2(\epsilon))^3$ and $V_n^\epsilon := (H^1(\epsilon))^3$ defined locally on each edge $\epsilon \in \mathcal{E}$. Let us also introduce the $L^2(\epsilon)$ -inner product and a lower-dimensional version of it acting on the nodes at the endpoints of ϵ for all functions $\mathbf{v}, \mathbf{w} \in V_u^\epsilon$ and $\mathbf{p}, \mathbf{q} \in V_n^\epsilon$ by

$$(\mathbf{v}, \mathbf{w})_\epsilon := \int_\epsilon \mathbf{v} \cdot \mathbf{w} \, d\sigma, \quad \langle \mathbf{p}, \mathbf{q} \rangle_\epsilon := \sum_{\mathbf{n} \sim \epsilon} \mathbf{p}(\mathbf{n}) \cdot \mathbf{q}(\mathbf{n}),$$

where the point evaluation is meaningful due to the continuous embedding $H^1(\mathfrak{e}) \hookrightarrow C^0(\bar{\mathfrak{e}})$ which holds in one spatial dimension. Note that in an abuse of notation, we will also allow functions defined only on the nodes of the networks to be plugged into $\langle \cdot, \cdot \rangle_{\mathfrak{e}}$, such as functions involving the unit normal $\nu_{\mathfrak{e}}$, but also $\mathbf{u}_{\mathbf{n}}$ and $\mathbf{r}_{\mathbf{n}}$.

Given the data

- $\mathbf{f}_{\mathfrak{e}}, \mathbf{g}_{\mathfrak{e}} \in V_{\mathbf{u}}^{\mathfrak{e}}$ for all edges $\mathfrak{e} \in \mathcal{E}$,
- $\mathbf{f}_{\mathbf{n}}, \mathbf{g}_{\mathbf{n}} \in \mathbb{R}^3$ for all free nodes $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{D}}$,
- $\mathbf{u}_{\mathbf{n}}^{\mathcal{D}}, \mathbf{r}_{\mathbf{n}}^{\mathcal{D}} \in \mathbb{R}^3$ for all Dirichlet nodes $\mathbf{n} \in \mathcal{N}_{\mathcal{D}}$,

the hybrid dual mixed formulation of problem (2.2) seeks

- $\mathbf{u}_{\mathfrak{e}}, \mathbf{r}_{\mathfrak{e}} \in V_{\mathbf{u}}^{\mathfrak{e}}$ for all edges $\mathfrak{e} \in \mathcal{E}$,
- $\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}} \in V_{\mathbf{n}}^{\mathfrak{e}}$ for all edges $\mathfrak{e} \in \mathcal{E}$,
- $\mathbf{u}_{\mathbf{n}}, \mathbf{r}_{\mathbf{n}} \in \mathbb{R}^3$ for all nodes $\mathbf{n} \in \mathcal{N}$,

we refer to as primal, dual, and hybrid unknowns, respectively. The hybrid unknowns should satisfy for all $\mathbf{n} \in \mathcal{N}_{\mathcal{D}}$ the Dirichlet boundary conditions (2.2d), while the dual unknowns should satisfy for all $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{D}}$ the balance conditions (2.2e). Moreover, for all edges $\mathfrak{e} \in \mathcal{E}$ the following equations should hold for all $\mathbf{p}, \mathbf{q} \in V_{\mathbf{n}}^{\mathfrak{e}}$ and $\mathbf{v}, \mathbf{w} \in V_{\mathbf{u}}^{\mathfrak{e}}$:

$$\begin{aligned} -(C_{\mathbf{n}}^{-1} \mathbf{n}_{\mathfrak{e}}, \mathbf{p})_{\mathfrak{e}} + (\mathbf{u}_{\mathfrak{e}}, \partial_x \mathbf{p})_{\mathfrak{e}} - (\mathbf{i}_{\mathfrak{e}} \times \mathbf{r}_{\mathfrak{e}}, \mathbf{p})_{\mathfrak{e}} &= \langle \mathbf{u}_{\mathbf{n}}, \mathbf{p} \nu_{\mathfrak{e}} \rangle_{\mathfrak{e}}, \\ -(C_{\mathbf{m}}^{-1} \mathbf{m}_{\mathfrak{e}}, \mathbf{q})_{\mathfrak{e}} + (\mathbf{r}_{\mathfrak{e}}, \partial_x \mathbf{q})_{\mathfrak{e}} &= \langle \mathbf{r}_{\mathbf{n}}, \mathbf{q} \nu_{\mathfrak{e}} \rangle_{\mathfrak{e}}, \\ (\partial_x \mathbf{n}_{\mathfrak{e}}, \mathbf{v})_{\mathfrak{e}} &= (\mathbf{f}_{\mathfrak{e}}, \mathbf{v})_{\mathfrak{e}}, \\ (\mathbf{i}_{\mathfrak{e}} \times \mathbf{n}_{\mathfrak{e}}, \mathbf{w})_{\mathfrak{e}} + (\partial_x \mathbf{m}_{\mathfrak{e}}, \mathbf{w})_{\mathfrak{e}} &= (\mathbf{g}_{\mathfrak{e}}, \mathbf{w})_{\mathfrak{e}}. \end{aligned} \quad (3.1)$$

Note that from the first and second equations in (3.1), one can immediately conclude that $\mathbf{u}_{\mathfrak{e}}, \mathbf{r}_{\mathfrak{e}} \in (H^1(\mathfrak{e}))^3$, even though we only seeked functions in $(L^2(\mathfrak{e}))^3$.

In the following, we will regularly view problem (3.1) as a local solver that maps the Dirichlet data and distributed source terms corresponding to an edge to the solution on that edge. The following lemma proves the well-posedness of this local solver.

Lemma 3.2 (Well-posedness of local solver). *Consider the edge $\mathfrak{e} \in \mathcal{E}$ and let the boundary data $\mathbf{u}_{\mathbf{n}}, \mathbf{r}_{\mathbf{n}} \in \mathbb{R}^3$ for the two nodes \mathbf{n} at the endpoints of \mathfrak{e} as well as the source terms $\mathbf{f}_{\mathfrak{e}}, \mathbf{g}_{\mathfrak{e}} \in V_{\mathbf{u}}^{\mathfrak{e}}$ be given. Consider the bilinear forms*

$$\begin{aligned} a((\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}}), (\mathbf{p}, \mathbf{q})) &:= (C_{\mathbf{n}}^{-1} \mathbf{n}_{\mathfrak{e}}, \mathbf{p})_{\mathfrak{e}} + (C_{\mathbf{m}}^{-1} \mathbf{m}_{\mathfrak{e}}, \mathbf{q})_{\mathfrak{e}}, \\ b((\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}}), (\mathbf{v}, \mathbf{w})) &:= -(\partial_x \mathbf{n}_{\mathfrak{e}}, \mathbf{v})_{\mathfrak{e}} - (\mathbf{i}_{\mathfrak{e}} \times \mathbf{n}_{\mathfrak{e}}, \mathbf{w})_{\mathfrak{e}} - (\partial_x \mathbf{m}_{\mathfrak{e}}, \mathbf{w})_{\mathfrak{e}}, \end{aligned}$$

and norms

$$\|(\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}})\|_{V_{\mathbf{n}}^{\mathfrak{e}} \times V_{\mathbf{m}}^{\mathfrak{e}}}^2 := \|\mathbf{n}_{\mathfrak{e}}\|_{\mathfrak{e}}^2 + \|\partial_x \mathbf{n}_{\mathfrak{e}}\|_{\mathfrak{e}}^2 + \|\mathbf{m}_{\mathfrak{e}}\|_{\mathfrak{e}}^2 + \|\partial_x \mathbf{m}_{\mathfrak{e}}\|_{\mathfrak{e}}^2, \quad (3.2a)$$

$$\|(\mathbf{v}, \mathbf{w})\|_{V_{\mathbf{u}}^{\mathfrak{e}} \times V_{\mathbf{r}}^{\mathfrak{e}}}^2 := \|\mathbf{v}\|_{\mathfrak{e}}^2 + \|\mathbf{w}\|_{\mathfrak{e}}^2. \quad (3.2b)$$

Then, the local solver defined by problem (3.1) can be written as

$$\begin{aligned} a((\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}}), (\mathbf{p}, \mathbf{q})) + b((\mathbf{p}, \mathbf{q}), (\mathbf{u}_{\mathfrak{e}}, \mathbf{r}_{\mathfrak{e}})) &= -\langle \mathbf{u}_{\mathbf{n}}, \mathbf{p} \nu_{\mathfrak{e}} \rangle_{\mathfrak{e}} - \langle \mathbf{r}_{\mathbf{n}}, \mathbf{q} \nu_{\mathfrak{e}} \rangle_{\mathfrak{e}}, \\ b((\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}}), (\mathbf{v}, \mathbf{w})) &= -(\mathbf{f}_{\mathfrak{e}}, \mathbf{v})_{\mathfrak{e}} - (\mathbf{g}_{\mathfrak{e}}, \mathbf{w})_{\mathfrak{e}}, \end{aligned} \quad (3.3)$$

and has a unique solution comprised of $\mathbf{u}_{\mathfrak{e}}, \mathbf{r}_{\mathfrak{e}} \in V_{\mathbf{u}}^{\mathfrak{e}}$ and $\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}} \in V_{\mathbf{n}}^{\mathfrak{e}}$. In particular, the following inf-sup condition holds with a constant $\beta > 0$:

$$\inf_{(\mathbf{v}, \mathbf{w}) \in V_{\mathbf{u}}^{\mathfrak{e}} \times V_{\mathbf{r}}^{\mathfrak{e}}} \sup_{0 \neq (\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}}) \in V_{\mathbf{n}}^{\mathfrak{e}} \times V_{\mathbf{m}}^{\mathfrak{e}}} \frac{b((\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}}), (\mathbf{v}, \mathbf{w}))}{\|(\mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}})\|_{V_{\mathbf{n}}^{\mathfrak{e}} \times V_{\mathbf{m}}^{\mathfrak{e}}} \|(\mathbf{v}, \mathbf{w})\|_{V_{\mathbf{u}}^{\mathfrak{e}} \times V_{\mathbf{r}}^{\mathfrak{e}}}} \geq \beta. \quad (3.4)$$

Proof. Note that rewriting (3.1) as saddle-point problem (3.3) allows us to apply classical inf-sup theory (see, e.g., [4]) to establish well-posedness. Specifically, we need to verify the two BNB conditions, *i.e.*, the coercivity of a on the kernel of b and the inf-sup stability of b . These conditions involve the tuple spaces $V_{\mathbf{u}}^{\epsilon} \times V_{\mathbf{n}}^{\epsilon}$ and $V_{\mathbf{n}}^{\epsilon} \times V_{\mathbf{m}}^{\epsilon}$, whose canonical norms are given in (3.2). For proving the first BNB condition, which is the coercivity of a on $\ker(b)$, we note that $(\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon}) \in \ker(b)$ implies that $\partial_x \mathbf{n}_{\epsilon} = 0$ and $\partial_x \mathbf{m}_{\epsilon} = -\mathbf{i}_{\epsilon} \times \mathbf{n}_{\epsilon}$. This, together with the uniform spectral bounds for $C_{\mathbf{n}}$ and $C_{\mathbf{m}}$, cf. (2.1), yields for any $(\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon}) \in \ker(b)$ that

$$\begin{aligned} a((\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon}), (\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon})) &\gtrsim \|\mathbf{n}_{\epsilon}\|_{\epsilon}^2 + \|\partial_x \mathbf{n}_{\epsilon}\|_{\epsilon}^2 + \frac{1}{2} \|\mathbf{m}_{\epsilon}\|_{\epsilon}^2 + \frac{1}{2} \|\partial_x \mathbf{m}_{\epsilon}\|_{\epsilon}^2 - \frac{1}{2} \|\mathbf{i}_{\epsilon} \times \mathbf{n}_{\epsilon}\|_{\epsilon}^2 \\ &\geq \frac{1}{2} \|(\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon})\|_{V_{\mathbf{n}}^{\epsilon} \times V_{\mathbf{m}}^{\epsilon}}^2, \end{aligned}$$

which is the desired coercivity property.

To prove the second BNB condition, which is the inf-sup condition (3.4) of b , we will choose for any given tuple $(\mathbf{v}, \mathbf{w}) \in V_{\mathbf{u}}^{\epsilon} \times V_{\mathbf{u}}^{\epsilon}$ a tuple $(\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon}) \in V_{\mathbf{n}}^{\epsilon} \times V_{\mathbf{m}}^{\epsilon}$ whose components \mathbf{n}_{ϵ} and \mathbf{m}_{ϵ} are defined as (minus one times) the antiderivatives of \mathbf{v} and \mathbf{w} along the edge ϵ , respectively. The integration constant of the antiderivatives is set to zero. This yields

$$\sup_{(\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon}) \neq 0} \frac{b((\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon}), (\mathbf{v}, \mathbf{w}))}{\|(\mathbf{n}_{\epsilon}, \mathbf{m}_{\epsilon})\|_{V_{\mathbf{n}}^{\epsilon} \times V_{\mathbf{m}}^{\epsilon}}} \geq \left(1 + \frac{4h_{\epsilon}^2}{\pi^2}\right)^{-1/2} \frac{(\mathbf{v}, \mathbf{v})_{\epsilon} + (\mathbf{w}, \mathbf{w})_{\epsilon} + (\mathbf{i}_{\epsilon} \times \mathbf{n}_{\epsilon}, \mathbf{w})_{\epsilon}}{\|(\mathbf{v}, \mathbf{w})\|_{V_{\mathbf{u}}^{\epsilon} \times V_{\mathbf{u}}^{\epsilon}}}, \quad (3.5)$$

for any $(\mathbf{v}, \mathbf{w}) \in V_{\mathbf{u}}^{\epsilon} \times V_{\mathbf{u}}^{\epsilon}$. Here, we used that $\|\mathbf{n}_{\epsilon}\|_{\epsilon} \leq \frac{2h_{\epsilon}}{\pi} \|\mathbf{v}\|_{\epsilon}$ and $\|\mathbf{m}_{\epsilon}\|_{\epsilon} \leq \frac{2h_{\epsilon}}{\pi} \|\mathbf{w}\|_{\epsilon}$, where the constant can be derived by explicitly computing the first eigenvalue of the Laplacian eigenvalue problem on the edge ϵ subject to mixed boundary conditions. Note that these estimates also imply that $(\mathbf{i}_{\epsilon} \times \mathbf{n}_{\epsilon}, \mathbf{w})_{\epsilon} \leq \|\mathbf{n}_{\epsilon}\|_{\epsilon} \|\mathbf{w}\|_{\epsilon} \leq \frac{2h_{\epsilon}}{\pi} \|\mathbf{v}\|_{\epsilon} \|\mathbf{w}\|_{\epsilon}$, which can be used to estimate the numerator on the right-hand side of (3.5) as

$$(\mathbf{v}, \mathbf{v})_{\epsilon} + (\mathbf{w}, \mathbf{w})_{\epsilon} + (\mathbf{i}_{\epsilon} \times \mathbf{n}_{\epsilon}, \mathbf{w})_{\epsilon} \geq \left(1 - \frac{h_{\epsilon}}{\pi}\right) \|(\mathbf{v}, \mathbf{w})\|_{V_{\mathbf{u}}^{\epsilon} \times V_{\mathbf{u}}^{\epsilon}}^2.$$

Thus, the bilinear form b is inf-sup stable, cf. (3.4), if the edge length satisfies $h_{\epsilon} < \pi$. Note that this limitation can be easily overcome by rescaling the network, so it is not assumed in the following. The assertion can be concluded using the classical inf-sup theory. \square

In its above form, the hybrid dual mixed formulation involves primal, dual, and hybrid unknowns. To derive an equivalent condensed problem involving only hybrid unknowns, we use an alternative interpretation of the local solver defined in (3.1). More precisely, for each edge $\epsilon \in \mathcal{E}$, the local solver defines the following mappings:

$$\begin{aligned} \mathbf{U}_{\epsilon} : (\mathbf{u}_{\mathbf{n}}, \mathbf{r}_{\mathbf{n}}, \mathbf{f}_{\epsilon}, \mathbf{g}_{\epsilon}) &\mapsto \mathbf{u}_{\epsilon}, & \mathbf{R}_{\epsilon} : (\mathbf{u}_{\mathbf{n}}, \mathbf{r}_{\mathbf{n}}, \mathbf{f}_{\epsilon}, \mathbf{g}_{\epsilon}) &\mapsto \mathbf{r}_{\epsilon}, \\ \mathbf{N}_{\epsilon} : (\mathbf{u}_{\mathbf{n}}, \mathbf{r}_{\mathbf{n}}, \mathbf{f}_{\epsilon}, \mathbf{g}_{\epsilon}) &\mapsto \mathbf{n}_{\epsilon}, & \mathbf{M}_{\epsilon} : (\mathbf{u}_{\mathbf{n}}, \mathbf{r}_{\mathbf{n}}, \mathbf{f}_{\epsilon}, \mathbf{g}_{\epsilon}) &\mapsto \mathbf{m}_{\epsilon}. \end{aligned} \quad (3.6)$$

Note that we abuse the notation and require that each of the first two arguments contains the two values corresponding to the nodes at the endpoints of ϵ . In the case of homogeneous source terms, *i.e.*, $\mathbf{f}_{\epsilon} = \mathbf{g}_{\epsilon} = 0$, we will omit the last two arguments.

Before we state the condensed problem, we need to introduce some notation. We denote the trial functions which are set to zero at Dirichlet nodes by $\boldsymbol{\lambda}, \phi : \mathcal{N} \rightarrow \mathbb{R}^3$. Writing $\boldsymbol{\lambda}_{\mathbf{n}}$ and $\phi_{\mathbf{n}}$ for the values of $\boldsymbol{\lambda}$ and ϕ at the node \mathbf{n} , respectively, this means that $\boldsymbol{\lambda}_{\mathbf{n}} = \phi_{\mathbf{n}} = 0$ for all $\mathbf{n} \in \mathcal{N}_{\mathbf{D}}$. Similarly, the test functions denoted by $\boldsymbol{\mu}, \psi : \mathcal{N} \rightarrow \mathbb{R}^3$ should also satisfy that $\boldsymbol{\mu}_{\mathbf{n}} = \psi_{\mathbf{n}} = 0$ for all $\mathbf{n} \in \mathcal{N}_{\mathbf{D}}$. Denoting the function space the trial and test functions are contained in by $V_{\boldsymbol{\lambda}}$, the condensed problem seeks $(\boldsymbol{\lambda}, \phi) \in V_{\boldsymbol{\lambda}} \times V_{\boldsymbol{\lambda}}$ such that

$$A((\boldsymbol{\lambda}, \phi), (\boldsymbol{\mu}, \psi)) = F((\boldsymbol{\mu}, \psi)) \quad (3.7)$$

holds for all $(\boldsymbol{\mu}, \psi) \in V_{\boldsymbol{\lambda}} \times V_{\boldsymbol{\lambda}}$. The bilinear form A of the condensed problem encodes the balance of forces and moments conditions for the operator-harmonic (homogeneous source terms) extension of the nodal data $(\boldsymbol{\lambda}, \phi)$, cf. (2.2e), and is given by

$$A((\boldsymbol{\lambda}, \phi), (\boldsymbol{\mu}, \psi)) := - \sum_{\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_{\mathbf{D}}} [\![\mathbf{N}_{\epsilon}(\boldsymbol{\lambda}_{\mathbf{n}}, \phi_{\mathbf{n}}) \nu_{\epsilon}]\!]_{\mathbf{n}} \cdot \boldsymbol{\mu}_{\mathbf{n}} + [\![\mathbf{M}_{\epsilon}(\boldsymbol{\lambda}_{\mathbf{n}}, \phi_{\mathbf{n}}) \nu_{\epsilon}]\!]_{\mathbf{n}} \cdot \psi_{\mathbf{n}}. \quad (3.8)$$

Inhomogeneous Dirichlet boundary data and source terms, as well as prescribed values for the balance conditions, are incorporated by the bilinear form F defined by

$$F((\boldsymbol{\mu}, \boldsymbol{\psi})) := \sum_{\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D} [(\llbracket \mathbf{N}_\epsilon(\mathbf{u}_\mathbf{n}^D, \mathbf{r}_\mathbf{n}^D, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon) \nu_\epsilon \rrbracket_\mathbf{n} - \mathbf{f}_\mathbf{n}) \cdot \boldsymbol{\mu}_\mathbf{n} + (\llbracket \mathbf{M}_\epsilon(\mathbf{u}_\mathbf{n}^D, \mathbf{r}_\mathbf{n}^D, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon) \nu_\epsilon \rrbracket_\mathbf{n} - \mathbf{g}_\mathbf{n}) \cdot \boldsymbol{\psi}_\mathbf{n}],$$

where we set $\mathbf{u}_\mathbf{n}^D = \mathbf{r}_\mathbf{n}^D = 0$ for all $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D$. After having computed $(\boldsymbol{\lambda}, \boldsymbol{\phi})$ as the solution to (3.7), the solution to the hybrid dual mixed formulation can be retrieved as follows: The displacements and rotations at the nodes can be obtained as $\mathbf{u}_\mathbf{n} = \boldsymbol{\lambda}_\mathbf{n} + \mathbf{u}_\mathbf{n}^D$ and $\mathbf{r}_\mathbf{n} = \boldsymbol{\phi}_\mathbf{n} + \mathbf{r}_\mathbf{n}^D$ for all $\mathbf{n} \in \mathcal{N}$ and the unknowns $\mathbf{u}_\epsilon, \mathbf{r}_\epsilon, \mathbf{n}_\epsilon, \mathbf{m}_\epsilon$ can for all $\epsilon \in \mathcal{E}$ be obtained from $\mathbf{u}_\mathbf{n}$ and $\mathbf{r}_\mathbf{n}$ using the local mappings defined in (3.6), *i.e.*,

$$\begin{aligned} \mathbf{u}_\epsilon &= \mathbf{U}_\epsilon(\mathbf{u}_\mathbf{n}, \mathbf{r}_\mathbf{n}, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon), & \mathbf{r}_\epsilon &= \mathbf{R}_\epsilon(\mathbf{u}_\mathbf{n}, \mathbf{r}_\mathbf{n}, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon), \\ \mathbf{n}_\epsilon &= \mathbf{N}_\epsilon(\mathbf{u}_\mathbf{n}, \mathbf{r}_\mathbf{n}, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon), & \mathbf{m}_\epsilon &= \mathbf{M}_\epsilon(\mathbf{u}_\mathbf{n}, \mathbf{r}_\mathbf{n}, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon). \end{aligned} \quad (3.9)$$

The following lemma shows that the bilinear form A is symmetric positive definite, proving the well-posedness of the condensed problem.

Lemma 3.3 (Properties of condensed problem). *The bilinear form A can be equivalently rewritten as the following symmetric expression*

$$A((\boldsymbol{\lambda}, \boldsymbol{\phi}), (\boldsymbol{\mu}, \boldsymbol{\psi})) = \sum_{\epsilon \in \mathcal{E}} [(C_n^{-1} \mathbf{N}_\epsilon(\boldsymbol{\lambda}_\mathbf{n}, \boldsymbol{\phi}_\mathbf{n}), \mathbf{N}_\epsilon(\boldsymbol{\mu}_\mathbf{n}, \boldsymbol{\psi}_\mathbf{n}))_\epsilon + (C_m^{-1} \mathbf{M}_\epsilon(\boldsymbol{\lambda}_\mathbf{n}, \boldsymbol{\phi}_\mathbf{n}), \mathbf{M}_\epsilon(\boldsymbol{\mu}_\mathbf{n}, \boldsymbol{\psi}_\mathbf{n}))_\epsilon]. \quad (3.10)$$

This implies that the bilinear form A is symmetric and positive definite, and hence the well-posedness of problem (3.7).

Proof. In this proof, we will abbreviate $\mathbf{N}_\epsilon(\boldsymbol{\lambda}_\mathbf{n}, \boldsymbol{\phi}_\mathbf{n})$ and $\mathbf{N}_\epsilon(\boldsymbol{\mu}_\mathbf{n}, \boldsymbol{\psi}_\mathbf{n})$ by \mathbf{N}_ϵ^1 and \mathbf{N}_ϵ^2 , respectively, and analogous abbreviations will be used for the other local mappings. Testing the first and second equations of system (3.1) with $\bar{\mathbf{N}}_\epsilon^{-1}$ and $\bar{\mathbf{M}}_\epsilon^{-1}$, respectively and using the symmetry of C_n and C_m , we obtain that

$$-\langle \boldsymbol{\mu}_\mathbf{n}, \mathbf{N}_\epsilon^1 \nu_\epsilon \rangle_\epsilon = (C_n^{-1} \mathbf{N}_\epsilon^1, \mathbf{N}_\epsilon^2)_\epsilon - (\mathbf{U}_\epsilon^2, \partial_x \mathbf{N}_\epsilon^1)_\epsilon + (\mathbf{i}_\epsilon \times \mathbf{R}_\epsilon^2, \mathbf{N}_\epsilon^1)_\epsilon, \quad (3.11)$$

$$-\langle \boldsymbol{\psi}_\mathbf{n}, \mathbf{M}_\epsilon^1 \nu_\epsilon \rangle_\epsilon = (C_m^{-1} \mathbf{M}_\epsilon^1, \mathbf{M}_\epsilon^2)_\epsilon - (\mathbf{R}_\epsilon^2, \partial_x \mathbf{M}_\epsilon^1)_\epsilon. \quad (3.12)$$

To rewrite the latter expressions, we test the third and fourth equations of system (3.1) with $\bar{\mathbf{U}}_\epsilon^2$ and $\bar{\mathbf{R}}_\epsilon^2$, respectively, and note that $\mathbf{f}_\epsilon = \mathbf{g}_\epsilon = 0$. Inserting the resulting equations into (3.11) and (3.12) and summing up yields the identity

$$-\langle \boldsymbol{\mu}_\mathbf{n}, \mathbf{N}_\epsilon^1 \nu_\epsilon \rangle_\epsilon - \langle \boldsymbol{\psi}_\mathbf{n}, \mathbf{M}_\epsilon^1 \nu_\epsilon \rangle_\epsilon = (C_n^{-1} \mathbf{N}_\epsilon^1, \mathbf{N}_\epsilon^2)_\epsilon + (C_m^{-1} \mathbf{M}_\epsilon^1, \mathbf{M}_\epsilon^2)_\epsilon.$$

Equation (3.10) then follows after summing over all edges and using the definition of $\llbracket \cdot \rrbracket_\mathbf{n}$.

To prove the positive definiteness of the bilinear form A , it only remains to show that $A((\boldsymbol{\lambda}, \boldsymbol{\phi}), (\boldsymbol{\lambda}, \boldsymbol{\phi})) = 0$ implies that $(\boldsymbol{\lambda}, \boldsymbol{\phi}) = 0$. However, this is a consequence of the well-posedness of the local solver (3.1) proved in Lemma 3.2. The positive definiteness implies the invertibility of the operator A and thus the well-posedness of problem (3.7). \square

3.2. Discretization

The HDG discretization for the numerical solution of the Timoshenko beam network model (2.2) is derived by discretizing the hybrid dual mixed formulation. Note that only the local solvers defined in (3.1) must be discretized since all other aspects are already finite-dimensional. For the discretization of the local solver corresponding to the edge $\epsilon \in \mathcal{E}$ we use the space of polynomials defined on ϵ of degree at most p , where $p \in \mathbb{N}$ is a fixed number. This space and the corresponding space of componentwise polynomials are denoted by $\mathbb{P}_p(\epsilon)$ and $V_p^\epsilon := (\mathbb{P}_p(\epsilon))^3$, respectively.

Given the data as in Section 3.1, the HDG method seeks

- $\bar{\mathbf{u}}_{\mathfrak{e}}, \bar{\mathbf{r}}_{\mathfrak{e}} \in V_p^{\mathfrak{e}}$ for all edges $\mathfrak{e} \in \mathcal{E}$,
- $\bar{\mathbf{n}}_{\mathfrak{e}}, \bar{\mathbf{m}}_{\mathfrak{e}} \in V_p^{\mathfrak{e}}$ for all edges $\mathfrak{e} \in \mathcal{E}$,
- $\bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{r}}_{\mathbf{n}} \in \mathbb{R}^3$ for all nodes $\mathbf{n} \in \mathcal{N}$,

where the hybrid HDG unknowns should satisfy for all $\mathbf{n} \in \mathcal{N}_D$ the Dirichlet boundary conditions (2.2d). We further require balance conditions for the numerical fluxes, which will be the discrete counterpart to the balance conditions (2.2e). Given a stabilization parameter $\tau_{\mathfrak{e}} > 0$, the numerical fluxes defined at the network nodes additionally incorporate the $\tau_{\mathfrak{e}}$ -weighted difference between the primal and hybrid HDG unknowns to increase the method's stability. The stabilization parameter should always satisfy $\tau_{\mathfrak{e}} h_{\mathfrak{e}} \lesssim 1$. The balance of numerical fluxes then reads for all $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D$ as

$$-\llbracket \bar{\mathbf{n}}_{\mathfrak{e}} \nu_{\mathfrak{e}} + \tau_{\mathfrak{e}} (\bar{\mathbf{u}}_{\mathfrak{e}} - \bar{\mathbf{u}}_{\mathbf{n}}) \rrbracket_{\mathbf{n}} = -f_{\mathbf{n}}, \quad -\llbracket \bar{\mathbf{m}}_{\mathfrak{e}} \nu_{\mathfrak{e}} + \tau_{\mathfrak{e}} (\bar{\mathbf{r}}_{\mathfrak{e}} - \bar{\mathbf{r}}_{\mathbf{n}}) \rrbracket_{\mathbf{n}} = -g_{\mathbf{n}}. \quad (3.13)$$

Moreover, for all edges $\mathfrak{e} \in \mathcal{E}$ we require that the HDG unknowns are connected by the following local equations, which should hold for all $\bar{\mathbf{p}}, \bar{\mathbf{q}}, \bar{\mathbf{v}}, \bar{\mathbf{w}} \in V_p^{\mathfrak{e}}$:

$$\begin{aligned} &-(C_n^{-1} \bar{\mathbf{n}}_{\mathfrak{e}}, \bar{\mathbf{p}})_{\mathfrak{e}} + (\bar{\mathbf{u}}_{\mathfrak{e}}, \partial_x \bar{\mathbf{p}})_{\mathfrak{e}} - (\mathbf{i}_{\mathfrak{e}} \times \bar{\mathbf{r}}_{\mathfrak{e}}, \bar{\mathbf{p}})_{\mathfrak{e}} = \langle \bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{p}} \nu_{\mathfrak{e}} \rangle_{\mathfrak{e}}, \\ &-(C_m^{-1} \bar{\mathbf{m}}_{\mathfrak{e}}, \bar{\mathbf{q}})_{\mathfrak{e}} + (\bar{\mathbf{r}}_{\mathfrak{e}}, \partial_x \bar{\mathbf{q}})_{\mathfrak{e}} = \langle \bar{\mathbf{r}}_{\mathbf{n}}, \bar{\mathbf{q}} \nu_{\mathfrak{e}} \rangle_{\mathfrak{e}}, \\ &(\partial_x \bar{\mathbf{n}}_{\mathfrak{e}}, \bar{\mathbf{v}})_{\mathfrak{e}} + \tau_{\mathfrak{e}} \langle \bar{\mathbf{u}}_{\mathfrak{e}}, \bar{\mathbf{v}} \rangle_{\mathfrak{e}} = (\mathbf{f}_{\mathfrak{e}}, \bar{\mathbf{v}})_{\mathfrak{e}} + \tau_{\mathfrak{e}} \langle \bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{v}} \rangle_{\mathfrak{e}}, \\ &(\mathbf{i}_{\mathfrak{e}} \times \bar{\mathbf{n}}_{\mathfrak{e}}, \bar{\mathbf{w}})_{\mathfrak{e}} + (\partial_x \bar{\mathbf{m}}_{\mathfrak{e}}, \bar{\mathbf{w}})_{\mathfrak{e}} + \tau_{\mathfrak{e}} \langle \bar{\mathbf{r}}_{\mathfrak{e}}, \bar{\mathbf{w}} \rangle_{\mathfrak{e}} = (\mathbf{g}_{\mathfrak{e}}, \bar{\mathbf{w}})_{\mathfrak{e}} + \tau_{\mathfrak{e}} \langle \bar{\mathbf{r}}_{\mathbf{n}}, \bar{\mathbf{w}} \rangle_{\mathfrak{e}}. \end{aligned} \quad (3.14)$$

Compared to (3.1), this system additionally incorporates $\tau_{\mathfrak{e}}$ -weighted stabilization terms, strengthening its diagonal. This is necessary because our choice of function spaces might render the proof of Lemma 3.2 invalid, which relied on an inf-sup condition between the spaces $V_{\mathbf{u}}^{\mathfrak{e}}$ and $V_{\mathbf{n}}^{\mathfrak{e}}$. Problem (3.14) defines a discrete version of the local solver from (3.1), which maps the Dirichlet data and distributed source terms corresponding to an edge to the discrete solution on that edge. The following lemma proves this discrete local solver is well-posed for any positive stabilization parameter.

Lemma 3.4 (Well-posedness of discretized local solver). *Consider the edge $\mathfrak{e} \in \mathcal{E}$ and let the boundary data $\bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{r}}_{\mathbf{n}} \in \mathbb{R}^3$ for the two nodes \mathbf{n} at the endpoints of \mathfrak{e} as well as the source terms $\mathbf{f}_{\mathfrak{e}}, \mathbf{g}_{\mathfrak{e}} \in V_{\mathbf{u}}^{\mathfrak{e}}$ be given. Then, for any stabilization parameters $\tau_{\mathfrak{e}} > 0$, the discretized local solver defined by (3.14) has a unique solution comprised of $\bar{\mathbf{u}}_{\mathfrak{e}}, \bar{\mathbf{r}}_{\mathfrak{e}}, \bar{\mathbf{n}}_{\mathfrak{e}}, \bar{\mathbf{m}}_{\mathfrak{e}} \in V_p^{\mathfrak{e}}$.*

Proof. Since (3.14) induces a square linear system of equations, it is sufficient to prove that $\bar{\mathbf{u}}_{\mathbf{n}} = \bar{\mathbf{r}}_{\mathbf{n}} = 0$ and $\mathbf{f}_{\mathfrak{e}} = \mathbf{g}_{\mathfrak{e}} = 0$ implies that $\bar{\mathbf{u}}_{\mathfrak{e}} = \bar{\mathbf{r}}_{\mathfrak{e}} = \bar{\mathbf{n}}_{\mathfrak{e}} = \bar{\mathbf{m}}_{\mathfrak{e}} = 0$. To this end, we test the individual equations of system (3.14) with the test functions $-\bar{\mathbf{n}}_{\mathfrak{e}}, -\bar{\mathbf{m}}_{\mathfrak{e}}, \bar{\mathbf{u}}_{\mathfrak{e}}$, and $\bar{\mathbf{r}}_{\mathfrak{e}}$, respectively. Summing up the resulting equations then yields that

$$(C_n^{-1} \bar{\mathbf{n}}_{\mathfrak{e}}, \bar{\mathbf{n}}_{\mathfrak{e}})_{\mathfrak{e}} + (C_m^{-1} \bar{\mathbf{m}}_{\mathfrak{e}}, \bar{\mathbf{m}}_{\mathfrak{e}})_{\mathfrak{e}} + \tau_{\mathfrak{e}} \langle \bar{\mathbf{u}}_{\mathfrak{e}}, \bar{\mathbf{u}}_{\mathfrak{e}} \rangle_{\mathfrak{e}} + \tau_{\mathfrak{e}} \langle \bar{\mathbf{r}}_{\mathfrak{e}}, \bar{\mathbf{r}}_{\mathfrak{e}} \rangle_{\mathfrak{e}} = 0. \quad (3.15)$$

By the positive definiteness of the matrices C_n and C_m , cf. (2.1), this implies that $\bar{\mathbf{n}}_{\mathfrak{e}} = \bar{\mathbf{m}}_{\mathfrak{e}} = 0$. Integrating the second equation in (3.14) by parts and exploiting that $\bar{\mathbf{r}}_{\mathfrak{e}}$ satisfies zero boundary conditions, cf. (3.15), we observe that $\bar{\mathbf{r}}_{\mathfrak{e}} = 0$. Finally, by the very same argument, the first equation of (3.14) reveals that $\bar{\mathbf{u}}_{\mathfrak{e}} = 0$, which concludes the proof. \square

To establish a condensed formulation of the HDG method similar to (3.7), we introduce for all edges $\mathfrak{e} \in \mathcal{E}$ discrete analogs to the local mappings from (3.6). These are defined in terms of the discrete local solver (3.14) and are denoted by

$$\begin{aligned} \bar{U}_{\mathfrak{e}}: (\bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{r}}_{\mathbf{n}}, \mathbf{f}_{\mathfrak{e}}, \mathbf{g}_{\mathfrak{e}}) &\mapsto \bar{\mathbf{u}}_{\mathfrak{e}}, & \bar{R}_{\mathfrak{e}}: (\bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{r}}_{\mathbf{n}}, \mathbf{f}_{\mathfrak{e}}, \mathbf{g}_{\mathfrak{e}}) &\mapsto \bar{\mathbf{r}}_{\mathfrak{e}}, \\ \bar{N}_{\mathfrak{e}}: (\bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{r}}_{\mathbf{n}}, \mathbf{f}_{\mathfrak{e}}, \mathbf{g}_{\mathfrak{e}}) &\mapsto \bar{\mathbf{n}}_{\mathfrak{e}}, & \bar{M}_{\mathfrak{e}}: (\bar{\mathbf{u}}_{\mathbf{n}}, \bar{\mathbf{r}}_{\mathbf{n}}, \mathbf{f}_{\mathfrak{e}}, \mathbf{g}_{\mathfrak{e}}) &\mapsto \bar{\mathbf{m}}_{\mathfrak{e}}. \end{aligned} \quad (3.16)$$

Recalling that V_λ denotes the space of \mathbb{R}^3 -valued functions defined on \mathcal{N} with homogeneous Dirichlet boundary conditions at the nodes in the set \mathcal{N}_D , the condensed formulation of the HDG method seeks the tuple $(\bar{\lambda}, \bar{\phi}) \in V_\lambda \times V_\lambda$ such that

$$\bar{A}((\bar{\lambda}, \bar{\phi}), (\bar{\mu}, \bar{\psi})) = \bar{F}((\bar{\mu}, \bar{\psi})) \quad (3.17)$$

holds for all $(\bar{\mu}, \bar{\psi}) \in V_\lambda \times V_\lambda$. The bilinear form \bar{A} encodes the balance conditions of the numerical fluxes at the free nodes of the network and is defined as

$$\begin{aligned} \bar{A}((\bar{\lambda}, \bar{\phi}), (\bar{\mu}, \bar{\psi})) := & - \sum_{\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D} \left[\left(\llbracket \bar{N}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}) \nu_\epsilon + \tau_\epsilon (\bar{U}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}) - \bar{\lambda}_\mathbf{n}) \rrbracket_\mathbf{n} \cdot \bar{\mu}_\mathbf{n} \right. \right. \\ & \left. \left. + \llbracket \bar{M}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}) \nu_\epsilon + \tau_\epsilon (\bar{R}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}) - \bar{\phi}_\mathbf{n}) \rrbracket_\mathbf{n} \cdot \bar{\psi}_\mathbf{n} \right) \right]. \end{aligned} \quad (3.18)$$

The linear form \bar{F} , which is used to incorporate inhomogeneous data, is defined as

$$\begin{aligned} \bar{F}((\bar{\mu}, \bar{\psi})) := & \sum_{\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D} \left[\left(\llbracket \bar{N}_\epsilon(\mathbf{u}_\mathbf{n}^D, \mathbf{r}_\mathbf{n}^D, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon) \nu_\epsilon + \tau_\epsilon \bar{U}_\epsilon(\mathbf{u}_\mathbf{n}^D, \mathbf{r}_\mathbf{n}^D, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon) \rrbracket_\mathbf{n} - \mathbf{f}_\mathbf{n} \right) \cdot \bar{\mu}_\mathbf{n} \right. \\ & \left. + \left(\llbracket \bar{M}_\epsilon(\mathbf{u}_\mathbf{n}^D, \mathbf{r}_\mathbf{n}^D, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon) \nu_\epsilon + \tau_\epsilon \bar{R}_\epsilon(\mathbf{u}_\mathbf{n}^D, \mathbf{r}_\mathbf{n}^D, \mathbf{f}_\epsilon, \mathbf{g}_\epsilon) \rrbracket_\mathbf{n} - \mathbf{g}_\mathbf{n} \right) \cdot \bar{\psi}_\mathbf{n} \right], \end{aligned}$$

where we again set $\mathbf{u}_\mathbf{n}^D = \mathbf{r}_\mathbf{n}^D = 0$ for all $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D$. After having computed the solution $(\bar{\lambda}, \bar{\phi})$ to the condensed HDG formulation (3.17), the HDG solution can be retrieved as follows: The hybrid HDG unknowns are obtained by $\bar{\mathbf{u}}_\mathbf{n} = \bar{\lambda}_\mathbf{n} + \mathbf{u}_\mathbf{n}^D$ and $\bar{\mathbf{r}}_\mathbf{n} = \bar{\phi}_\mathbf{n} + \mathbf{r}_\mathbf{n}^D$ for all $\mathbf{n} \in \mathcal{N}$, and the primal and dual HDG unknowns $\bar{\mathbf{u}}_\epsilon, \bar{\mathbf{r}}_\mathbf{n}, \bar{\mathbf{n}}_\epsilon, \bar{\mathbf{m}}_\epsilon$ can be obtained for all $\epsilon \in \mathcal{E}$ similar to (3.9), but with the local mappings (3.16).

Remark 3.5 (Size of the global system of equations). Importantly, the global system of equations corresponding to (3.17) does not change its size if the polynomial degree p is increased. This is because the hybrid unknowns live in a zero-dimensional domain, and a single parameter can characterize all polynomials in such a domain.

The following lemma also shows the symmetry and positive definiteness of the bilinear form \bar{A} , which proves the well-posedness of the condensed HDG formulation.

Lemma 3.6 (Properties of condensed HDG problem). *The bilinear form \bar{A} can be equivalently rewritten as the following symmetric expression*

$$\begin{aligned} \bar{A}((\bar{\lambda}, \bar{\phi}), (\bar{\mu}, \bar{\psi})) = & \sum_{\epsilon \in \mathcal{E}} \left[(C_\mathbf{n}^{-1} \bar{N}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}), \bar{N}_\epsilon(\bar{\mu}_\mathbf{n}, \bar{\psi}_\mathbf{n}))_\epsilon + (C_\mathbf{m}^{-1} \bar{M}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}), \bar{M}_\epsilon(\bar{\mu}_\mathbf{n}, \bar{\psi}_\mathbf{n}))_\epsilon \right] \\ & + \sum_{\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D} \left[\left[\tau_\epsilon (\bar{U}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}) - \bar{\lambda}_\mathbf{n}) \cdot (\bar{U}_\epsilon(\bar{\mu}_\mathbf{n}, \bar{\psi}_\mathbf{n}) - \bar{\mu}_\mathbf{n}) \right]_\mathbf{n} \right. \\ & \left. + \left[\tau_\epsilon (\bar{R}_\epsilon(\bar{\lambda}_\mathbf{n}, \bar{\phi}_\mathbf{n}) - \bar{\phi}_\mathbf{n}) \cdot (\bar{R}_\epsilon(\bar{\mu}_\mathbf{n}, \bar{\psi}_\mathbf{n}) - \bar{\psi}_\mathbf{n}) \right]_\mathbf{n} \right]. \end{aligned}$$

Therefore, for any $\tau_\epsilon > 0$, the bilinear form \bar{A} is symmetric and positive definite. Consequently, the condensed formulation (3.17) of the HDG method is well-posed.

Proof. The proof of this result uses very similar arguments as the proof of Lemma 3.3, but applied to (3.14) instead of (3.1). Therefore, it is omitted for the sake of brevity. \square

4. ERROR ANALYSIS

In this section, we perform an *a priori* error analysis of the proposed HDG discretization for the Timoshenko beam network problem. The error analysis focuses on the convergence of the HDG discretization with respect to the edge length h_ϵ , for a fixed polynomial degree. While this may initially appear counterintuitive, as the network

is assumed to be given, we implicitly consider a sequence of refined networks obtained by subdividing the edges. Such refinement does not affect the exact solution of the beam network model. The error analysis is inspired by [14], where a projection-based error analysis of the HDG method for an elliptic model problem is performed. The projection is tailored to the HDG structure, allowing for a relatively simple and concise error analysis. In the present setting of Timoshenko beam network problems, the edgewise defined projection is given by the mapping $\Pi: (H^1(\mathfrak{e}))^3 \times (H^1(\mathfrak{e}))^3 \rightarrow (\mathbb{P}_p(\mathfrak{e}))^3 \times (\mathbb{P}_p(\mathfrak{e}))^3$, whose components $\Pi(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}) = (\Pi_1(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}), \Pi_2(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}))$ are for all $\mathfrak{e} \in \mathcal{E}$ defined by the following conditions:

$$(\Pi_1(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}), \bar{\mathbf{v}})_\mathfrak{e} = (\mathbf{u}_\mathfrak{e}, \bar{\mathbf{v}})_\mathfrak{e} \quad \text{for all } \bar{\mathbf{v}} \in \mathbb{P}_{p-1}(\mathfrak{e}), \quad (4.1a)$$

$$(\Pi_2(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}), \bar{\mathbf{v}})_\mathfrak{e} = (\mathbf{n}_\mathfrak{e}, \bar{\mathbf{v}})_\mathfrak{e} \quad \text{for all } \bar{\mathbf{v}} \in \mathbb{P}_{p-1}(\mathfrak{e}), \quad (4.1b)$$

$$[\Pi_2(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e})]_\nu + \tau_\mathfrak{e}[\Pi_1(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e})] = \mathbf{n}_\mathfrak{e}\nu + \tau_\mathfrak{e}\mathbf{u}_\mathfrak{e} \quad \text{for all } \mathbf{n} \sim \mathfrak{e}, \quad (4.1c)$$

where we recall that we write $\mathbf{n} \sim \mathfrak{e}$ if the node \mathbf{n} is an endpoint of the edge \mathfrak{e} . The well-posedness of this projection can be concluded from Theorem 2.1 of [14], where the result was proved in a more general multi-dimensional setting.

To simplify the notation of the following error analysis, we define the errors

$$\begin{aligned} \theta_\mathfrak{e}^u &= \mathbf{u}_\mathfrak{e} - \Pi_1(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}), & \epsilon_\mathfrak{e}^u &= \Pi_1(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}) - \bar{\mathbf{u}}_\mathfrak{e}, & \theta_\mathfrak{e}^u + \epsilon_\mathfrak{e}^u &= \mathbf{u}_\mathfrak{e} - \bar{\mathbf{u}}_\mathfrak{e} \\ \theta_\mathfrak{e}^n &= \mathbf{n}_\mathfrak{e} - \Pi_2(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}), & \epsilon_\mathfrak{e}^n &= \Pi_2(\mathbf{u}_\mathfrak{e}, \mathbf{n}_\mathfrak{e}) - \bar{\mathbf{n}}_\mathfrak{e}, & \theta_\mathfrak{e}^n + \epsilon_\mathfrak{e}^n &= \mathbf{n}_\mathfrak{e} - \bar{\mathbf{n}}_\mathfrak{e}, \end{aligned} \quad (4.2)$$

and similar definitions are used for the variables $\mathbf{r}_\mathfrak{e}$ and $\mathbf{m}_\mathfrak{e}$.

The following lemma states projection error estimates for the operator Π , which are taken from Theorem 2.1 of [14]. We denote the $H^k(\mathfrak{e})$ -seminorm for any $k \in \mathbb{N}_0$ by $|\cdot|_{\mathfrak{e},k}$.

Lemma 4.1 (Projection error estimates). *Given $p_u, p_n \in [0, p]$, there holds for any edge $\mathfrak{e} \in \mathcal{E}$ and all $\mathbf{u}_\mathfrak{e} \in H^{p_u+1}(\mathfrak{e})$, $\mathbf{n}_\mathfrak{e} \in H^{p_n+1}(\mathfrak{e})$ that*

$$\begin{aligned} \|\theta_\mathfrak{e}^u\|_\mathfrak{e} &\lesssim h_\mathfrak{e}^{p_u+1} |\mathbf{u}_\mathfrak{e}|_{\mathfrak{e},p_u+1} + \frac{1}{\tau_\mathfrak{e}} h_\mathfrak{e}^{p_n+1} |\mathbf{n}_\mathfrak{e}|_{\mathfrak{e},p_n+1} \\ \|\theta_\mathfrak{e}^n\|_\mathfrak{e} &\lesssim \tau_\mathfrak{e} h_\mathfrak{e}^{p_u+1} |\mathbf{u}_\mathfrak{e}|_{\mathfrak{e},p_u+1} + h_\mathfrak{e}^{p_n+1} |\mathbf{n}_\mathfrak{e}|_{\mathfrak{e},p_n+1}, \end{aligned}$$

and analogous estimates hold for $\theta_\mathfrak{e}^r$ and $\theta_\mathfrak{e}^m$.

Recall that the primal unknowns are edgewise H^1 -regular and satisfy continuity conditions at the nodes. Therefore, for any edge $\mathfrak{e} \in \mathcal{E}$ and any node $\mathbf{n} \sim \mathfrak{e}$, $\mathbf{u}_\mathfrak{e}(\mathbf{n}) = \mathbf{u}_\mathbf{n}$ and $\mathbf{r}_\mathfrak{e}(\mathbf{n}) = \mathbf{r}_\mathbf{n}$ holds. This motivates omitting the primal unknowns' subscripts by simply writing \mathbf{u} and \mathbf{r} . Similarly, we may omit the subscript of $\mathbf{n}_\mathfrak{e}$ and $\mathbf{m}_\mathfrak{e}$. The following lemma states error equations that will be frequently used in the subsequent error analysis.

Lemma 4.2 (Error equations). *For any edge $\mathfrak{e} \in \mathcal{E}$ and all test functions $\bar{\mathbf{p}}, \bar{\mathbf{q}}, \bar{\mathbf{v}}, \bar{\mathbf{w}} \in (\mathbb{P}_p(\mathfrak{e}))^3$, there hold the identities*

$$(\epsilon_\mathfrak{e}^u, \partial_x \bar{\mathbf{p}})_\mathfrak{e} = (C_\mathbf{n}^{-1}(\mathbf{n} - \bar{\mathbf{n}}_\mathfrak{e}), \bar{\mathbf{p}})_\mathfrak{e} + (\mathbf{i}_\mathfrak{e} \times (\mathbf{r} - \bar{\mathbf{r}}_\mathfrak{e}), \bar{\mathbf{p}})_\mathfrak{e} + \langle \mathbf{u} - \bar{\mathbf{u}}_\mathfrak{n}, \bar{\mathbf{p}} \nu_\mathfrak{e} \rangle_\mathfrak{e}, \quad (4.3a)$$

$$(\epsilon_\mathfrak{e}^r, \partial_x \bar{\mathbf{q}})_\mathfrak{e} = (C_\mathbf{m}^{-1}(\mathbf{m} - \bar{\mathbf{m}}_\mathfrak{e}), \bar{\mathbf{q}})_\mathfrak{e} + \langle \mathbf{r} - \bar{\mathbf{r}}_\mathfrak{n}, \bar{\mathbf{q}} \nu_\mathfrak{e} \rangle_\mathfrak{e}, \quad (4.3b)$$

$$(\partial_x \epsilon_\mathfrak{e}^n, \bar{\mathbf{v}})_\mathfrak{e} = \langle \tau_\mathfrak{e}(\bar{\mathbf{u}}_\mathfrak{e} - \bar{\mathbf{u}}_\mathfrak{n}) - \theta_\mathfrak{e}^n \nu_\mathfrak{e}, \bar{\mathbf{v}} \rangle_\mathfrak{e}, \quad (4.3c)$$

$$(\partial_x \epsilon_\mathfrak{e}^m, \bar{\mathbf{w}})_\mathfrak{e} = \langle \tau_\mathfrak{e}(\bar{\mathbf{r}}_\mathfrak{e} - \bar{\mathbf{r}}_\mathfrak{n}) - \theta_\mathfrak{e}^m \nu_\mathfrak{e}, \bar{\mathbf{w}} \rangle_\mathfrak{e} - (\mathbf{i}_\mathfrak{e} \times (\mathbf{n} - \bar{\mathbf{n}}_\mathfrak{e}), \bar{\mathbf{w}})_\mathfrak{e}. \quad (4.3d)$$

Furthermore, for any nodal functions $\mathbf{v}_\mathbf{n}, \mathbf{w}_\mathbf{n}$ with support in $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D$ we have that

$$0 = \llbracket \epsilon_\mathfrak{e}^n \nu_\mathfrak{e} + \tau_\mathfrak{e} \epsilon_\mathfrak{e}^u - \tau_\mathfrak{e}(\mathbf{u} - \bar{\mathbf{u}}_\mathfrak{n}) \rrbracket_\mathbf{n} \mathbf{v}_\mathbf{n} + \llbracket \epsilon_\mathfrak{e}^m \nu_\mathfrak{e} + \tau_\mathfrak{e} \epsilon_\mathfrak{e}^r - \tau_\mathfrak{e}(\mathbf{r} - \bar{\mathbf{r}}_\mathfrak{n}) \rrbracket_\mathbf{n} \mathbf{w}_\mathbf{n}. \quad (4.3e)$$

Proof. Error equations (4.3a) and (4.3b) can be derived from (3.1) and (3.14), using property (4.1a) of the projection Π . To obtain (4.3c) and (4.3d), an integration by parts is necessary before property (4.1b) of Π can be used. The error equation (4.3e) can be derived using property (4.1c) of Π and the balance conditions (2.2e) and (3.13). \square

The following two lemmas provide preliminary results, combined in Theorem 4.5 to a full error estimate of the proposed HDG discretization. The first lemma provides an estimate for the dual unknowns.

Lemma 4.3 (Estimate for dual unknowns). *It holds that*

$$\sum_{\epsilon \in \mathcal{E}} \|\epsilon_\epsilon^n\|_\epsilon^2 \lesssim \sum_{\epsilon \in \mathcal{E}} [\|\theta_\epsilon^n\|_\epsilon^2 + \|\theta_\epsilon^r\|_\epsilon^2], \quad \sum_{\epsilon \in \mathcal{E}} \|\epsilon_\epsilon^m\|_\epsilon^2 \lesssim \sum_{\epsilon \in \mathcal{E}} [\|\theta_\epsilon^m\|_\epsilon^2 + \|\theta_\epsilon^n\|_\epsilon \|\epsilon_\epsilon^r\|_\epsilon]. \quad (4.4)$$

Proof. We begin this proof with the obvious identity

$$-\langle \epsilon_\epsilon^n \nu_\epsilon + \tau_\epsilon \epsilon_\epsilon^u - \tau_\epsilon (\mathbf{u} - \bar{\mathbf{u}}_n), \mathbf{u} - \bar{\mathbf{u}}_n \rangle_\epsilon = -\langle \epsilon_\epsilon^n \nu, \mathbf{u} - \bar{\mathbf{u}}_n \rangle_\epsilon + \langle \tau_\epsilon (\mathbf{u} - \bar{\mathbf{u}}_n) - \tau_\epsilon \epsilon_\epsilon^u, \mathbf{u} - \bar{\mathbf{u}}_n \rangle_\epsilon. \quad (4.5)$$

To derive an equivalent representation for the left-hand side of (4.5), we test (4.3a) with the test function ϵ_ϵ^n and rearrange the terms which yields that

$$-\langle \mathbf{u} - \bar{\mathbf{u}}_n, \epsilon_\epsilon^n \nu_\epsilon \rangle_\epsilon = (C_n^{-1}(\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \epsilon_\epsilon^n)_\epsilon + (\mathbf{i}_\epsilon \times (\mathbf{r} - \bar{\mathbf{r}}_\epsilon), \epsilon_\epsilon^n)_\epsilon - (\epsilon_\epsilon^u, \partial_x \epsilon_\epsilon^n)_\epsilon. \quad (4.6)$$

The last term on the right-hand side of (4.6) can be further rewritten using the identity

$$(\partial_x \epsilon_\epsilon^n, \epsilon_\epsilon^u)_\epsilon = \langle \tau_\epsilon (\bar{\mathbf{u}}_\epsilon - \bar{\mathbf{u}}_n) - \theta_\epsilon^n \nu_\epsilon, \epsilon_\epsilon^u \rangle_\epsilon = \langle \tau_\epsilon (\mathbf{u} - \bar{\mathbf{u}}_n) - \tau_\epsilon \epsilon_\epsilon^u, \epsilon_\epsilon^u \rangle_\epsilon, \quad (4.7)$$

which can be derived by testing (4.3c) with the test function ϵ_ϵ^u and using property (4.1c) of the projection Π . Inserting (4.6) and (4.7) into (4.5) then yields that

$$\begin{aligned} -\langle \epsilon_\epsilon^n \nu_\epsilon + \tau_\epsilon \epsilon_\epsilon^u - \tau_\epsilon (\mathbf{u} - \bar{\mathbf{u}}_n), \mathbf{u} - \bar{\mathbf{u}}_n \rangle_\epsilon &= (C_n^{-1}(\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \epsilon_\epsilon^n)_\epsilon + (\mathbf{i}_\epsilon \times (\mathbf{r} - \bar{\mathbf{r}}_\epsilon), \epsilon_\epsilon^n)_\epsilon \\ &\quad + \tau_\epsilon \langle \epsilon_\epsilon^u - (\mathbf{u} - \bar{\mathbf{u}}_n), \epsilon_\epsilon^u - (\mathbf{u} - \bar{\mathbf{u}}_n) \rangle_\epsilon, \end{aligned} \quad (4.8)$$

which is the desired equivalent representation.

Summing the latter identity over all edges $\epsilon \in \mathcal{E}$ and using (4.3e) for the nodal functions $\mathbf{v}_n = \mathbf{u} - \bar{\mathbf{u}}_n$, we obtain that

$$0 = \sum_{\epsilon \in \mathcal{E}} [(C_n^{-1}(\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \epsilon_\epsilon^n)_\epsilon + (\mathbf{i}_\epsilon \times (\mathbf{r} - \bar{\mathbf{r}}_\epsilon), \epsilon_\epsilon^n)_\epsilon] + \sum_{n \in \mathcal{N}} \tau_\epsilon \left\| (\epsilon_\epsilon^u - (\mathbf{u} - \bar{\mathbf{u}}_n))^2 \right\|_n, \quad (4.9)$$

where we interpret the square of a vector as a dot product with itself. Algebraic manipulations and the non-negativity of the last term in (4.9) then yield the estimate

$$\sum_{\epsilon \in \mathcal{E}} (C_n^{-1} \epsilon_\epsilon^n, \epsilon_\epsilon^n)_\epsilon \leq \sum_{\epsilon \in \mathcal{E}} [-(C_n^{-1} \theta_\epsilon^n, \epsilon_\epsilon^n)_\epsilon - (\mathbf{i}_\epsilon \times \theta_\epsilon^r, \epsilon_\epsilon^n)_\epsilon].$$

The first inequality in (4.4) can then be concluded from Hölder's and Young's inequalities and the uniform bounds of the coefficients C_n and C_m , cf. (2.1).

Similar considerations can be applied to the rotations and moments, which yields that

$$0 = \sum_{\epsilon \in \mathcal{E}} [(C_m^{-1}(\mathbf{m} - \bar{\mathbf{m}}_\epsilon), \epsilon_\epsilon^m)_\epsilon + (\mathbf{i}_\epsilon \times (\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \epsilon_\epsilon^r)_\epsilon] + \sum_{n \in \mathcal{N}} \tau_\epsilon \left\| (\epsilon_\epsilon^r - (\mathbf{r} - \bar{\mathbf{r}}_n))^2 \right\|_n, \quad (4.10)$$

and the second inequality in (4.4) can again be obtained by using the non-negativity of the last term in (4.10), as well as Hölder's and Young's inequalities and bounds (2.1). \square

The second lemma establishes an estimate for the primal unknowns based on an Aubin–Nitsche-type argument. This requires edgewise elliptic regularity of an auxiliary (dual) problem, whose data and solution we denote by a dagger. The auxiliary problem coincides with (2.2), but with homogeneous concentrated forces,

moments, and Dirichlet data; that is, $\mathbf{f}_n^\dagger = \mathbf{g}_n^\dagger = 0$ for all $\mathbf{n} \in \mathcal{N}$, and $\mathbf{u}_n^\dagger = \mathbf{r}_n^\dagger = 0$ for all $\mathbf{n} \in \mathcal{N}_D$. In the following, we always assume that the dual problem satisfies the elliptic regularity estimate:

$$\sum_{\epsilon \in \mathcal{E}} \left[|\mathbf{u}_\epsilon^\dagger|_{2,\epsilon}^2 + |\mathbf{r}_\epsilon^\dagger|_{2,\epsilon}^2 + |\mathbf{n}_\epsilon^\dagger|_{1,\epsilon}^2 + |\mathbf{m}_\epsilon^\dagger|_{1,\epsilon}^2 \right] \lesssim \sum_{\epsilon \in \mathcal{E}} \left[\|\mathbf{f}_\epsilon^\dagger\|_\epsilon^2 + \|\mathbf{g}_\epsilon^\dagger\|_\epsilon^2 \right]. \quad (4.11)$$

Such an estimate holds if the coefficients C_n and C_m are edgewise Lipschitz continuous.

Lemma 4.4 (Estimate for primal unknowns). *It holds that*

$$\sum_{\epsilon \in \mathcal{E}} \left[\|\epsilon_\epsilon^u\|_\epsilon^2 + \|\epsilon_\epsilon^r\|_\epsilon^2 \right] \lesssim \sum_{\epsilon \in \mathcal{E}} (h_\epsilon + \tau_\epsilon h_\epsilon^2)^2 \left[\|\theta_\epsilon^n\|_\epsilon^2 + \|\theta_\epsilon^m\|_\epsilon^2 + \|\theta_\epsilon^r\|_\epsilon^2 \right] + \sum_{\epsilon \in \mathcal{E}} \left(\frac{h_\epsilon}{\tau_\epsilon} \right)^2 \left[\|\theta_\epsilon^n\|_\epsilon^2 + \|\theta_\epsilon^r\|_\epsilon^2 \right].$$

Proof. To simplify notation for the auxiliary problem, we henceforth write \mathbf{u}^\dagger instead of $\mathbf{u}_\epsilon^\dagger$ and \mathbf{u}_n^\dagger , and similarly for \mathbf{r}^\dagger , \mathbf{n}^\dagger , and \mathbf{m}^\dagger . This abuse of notation is justified, as the auxiliary problem satisfies the same continuity properties as the original one, cf. (2.2). We begin the proof by deriving an equivalent expression for $(\epsilon_\epsilon^u, \mathbf{f}_\epsilon^\dagger)_\epsilon$. By the definition of \mathbf{n}^\dagger , cf. (2.2b), integration by parts, and properties (4.1b) and (4.1c), we get that

$$\begin{aligned} (\epsilon_\epsilon^u, \mathbf{f}_\epsilon^\dagger)_\epsilon &= (\epsilon^u, \partial_x \Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger))_\epsilon + \langle \epsilon_\epsilon^u, \tau_\epsilon (\Pi_1(\mathbf{u}^\dagger, \mathbf{n}^\dagger) - \mathbf{u}^\dagger) \rangle_\epsilon \\ &= (C_n^{-1}(\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger))_\epsilon + (\mathbf{i}_\epsilon \times (\mathbf{r} - \bar{\mathbf{r}}_\epsilon), \Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger))_\epsilon \\ &\quad + \langle \mathbf{u} - \bar{\mathbf{u}}_n, \Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger) \nu_\epsilon \rangle_\epsilon + \langle \epsilon_\epsilon^u, \tau_\epsilon (\Pi_1(\mathbf{u}^\dagger, \mathbf{n}^\dagger) - \mathbf{u}^\dagger) \rangle_\epsilon, \end{aligned} \quad (4.12)$$

where we used (4.3a) in the last equality.

In the following, we will sum equation (4.12) over all edges $\epsilon \in \mathcal{E}$. For the last two terms in (4.12), this results in

$$\begin{aligned} &\sum_{\epsilon \in \mathcal{E}} \left[\langle \mathbf{u} - \bar{\mathbf{u}}_n, \Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger) \nu_\epsilon \rangle_\epsilon + \langle \epsilon_\epsilon^u, \tau_\epsilon (\Pi_1(\mathbf{u}^\dagger, \mathbf{n}^\dagger) - \mathbf{u}^\dagger) \rangle_\epsilon \right] \\ &= \sum_{n \in \mathcal{N}} \left[\langle \mathbf{u} - \bar{\mathbf{u}}_n, (\Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger) - \mathbf{n}^\dagger) \nu_\epsilon \rangle_\epsilon \right] + \sum_{n \in \mathcal{N}} \left[\langle \epsilon_\epsilon^u \cdot \tau_\epsilon [\Pi_1(\mathbf{u}^\dagger, \mathbf{n}^\dagger) - \mathbf{u}^\dagger] \rangle_\epsilon \right] \\ &= \sum_{n \in \mathcal{N}} \left[\langle \epsilon_\epsilon^u - (\mathbf{u} - \bar{\mathbf{u}}_n), \tau_\epsilon [\Pi_1(\mathbf{u}^\dagger, \mathbf{n}^\dagger) - \mathbf{u}^\dagger] \rangle_\epsilon \right] = \sum_{\epsilon \in \mathcal{E}} (\epsilon_\epsilon^n, \partial_x \mathbf{u}^\dagger)_\epsilon. \end{aligned} \quad (4.13)$$

Note that the first equality follows after rearranging the summands and inserting the term $\langle (\mathbf{u} - \bar{\mathbf{u}}_n) \cdot \mathbf{n}^\dagger \nu_\epsilon \rangle_n$ which equals zero since $\mathbf{u} - \bar{\mathbf{u}}_n = 0$ for all $\mathbf{n} \in \mathcal{N}_D$ and $\langle \mathbf{n}^\dagger \nu_\epsilon \rangle_n = 0$ for all $\mathbf{n} \in \mathcal{N} \setminus \mathcal{N}_D$. The second equality is a direct consequence of (4.1c). To prove the last equality, we use the following two identities

$$\sum_{n \in \mathcal{N}} \left[\langle \epsilon_\epsilon^u - (\mathbf{u} - \bar{\mathbf{u}}_n), \tau_\epsilon \Pi_1(\mathbf{u}^\dagger, \mathbf{n}^\dagger) \rangle_n \right] = - \sum_{n \in \mathcal{N}} \left[\langle \tau_\epsilon (\bar{\mathbf{u}}_\epsilon - \bar{\mathbf{u}}_n) - \theta_\epsilon^n \nu_\epsilon, \Pi_1(\mathbf{u}^\dagger, \mathbf{n}^\dagger) \rangle_n \right], \quad (4.14)$$

$$\sum_{n \in \mathcal{N}} \left[\langle \epsilon_\epsilon^u - (\mathbf{u} - \bar{\mathbf{u}}_n), \tau_\epsilon \mathbf{u}^\dagger \rangle_n \right] = - \sum_{n \in \mathcal{N}} \left[\langle \epsilon_\epsilon^n \nu_\epsilon, \mathbf{u}^\dagger \rangle_n \right], \quad (4.15)$$

as well as (4.3c) and integrate by parts. Identity (4.14) can be concluded from (4.1c), and (4.15) follows from (4.1c) and the balance condition $\langle \bar{\mathbf{n}}_\epsilon \nu_\epsilon + \tau_\epsilon (\bar{\mathbf{u}}_\epsilon - \bar{\mathbf{u}}_n) \rangle_n = \langle \mathbf{n}_\epsilon \nu_\epsilon \rangle_n = \mathbf{f}_n$.

Denoting by π the edgewise L^2 -orthogonal projection onto polynomials of degree at most $p-1$, we finally obtain for the sum of equation (4.12) over all edges $\epsilon \in \mathcal{E}$ that

$$\sum_{\epsilon \in \mathcal{E}} (\epsilon_\epsilon^u, \mathbf{f}_\epsilon^\dagger)_\epsilon = \sum_{\epsilon \in \mathcal{E}} (C_n^{-1}(\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger) - \mathbf{n}^\dagger)_\epsilon + \sum_{\epsilon \in \mathcal{E}} (\mathbf{i}_\epsilon \times (\mathbf{r} - \bar{\mathbf{r}}_\epsilon), \Pi_2(\mathbf{u}^\dagger, \mathbf{n}^\dagger))_\epsilon$$

$$-\sum_{\epsilon \in \mathcal{E}} (\mathbf{n} - \bar{\mathbf{n}}_\epsilon, \mathbf{i}_\epsilon \times \mathbf{r}^\dagger)_\epsilon + \sum_{\epsilon \in \mathcal{E}} (\Pi_2(\mathbf{u}, \mathbf{n}) - \mathbf{n}, \partial_x \mathbf{u}^\dagger - \pi \partial_x \mathbf{u}^\dagger)_\epsilon,$$

where we used (4.13) as well as the identity

$$(C_{\mathbf{n}}^{-1}(\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \mathbf{n}^\dagger)_\epsilon = -(\mathbf{n} - \bar{\mathbf{n}}_\epsilon, \partial_x \mathbf{u}^\dagger + \mathbf{i}_\epsilon \times \mathbf{r}^\dagger)_\epsilon,$$

which follows by the definition of \mathbf{n}^\dagger , cf. (2.2). Note that the term $\pi \partial_x \mathbf{u}^\dagger$ does not contribute anything since it is a polynomial of degree at most $p-1$.

Similar considerations can be applied to derive an equivalent representation for the sum of $(\epsilon_\epsilon^r, \mathbf{g}_\epsilon^\dagger)$ over all edges $\epsilon \in \mathcal{E}$, which results in

$$\begin{aligned} \sum_{\epsilon \in \mathcal{E}} (\epsilon_\epsilon^r, \mathbf{g}_\epsilon^\dagger)_\epsilon &= \sum_{\epsilon \in \mathcal{E}} (C_{\mathbf{m}}^{-1}(\mathbf{m} - \bar{\mathbf{m}}_\epsilon), \Pi_2(\mathbf{r}^\dagger, \mathbf{m}^\dagger) - \mathbf{m}^\dagger)_\epsilon + \sum_{\epsilon \in \mathcal{E}} (\epsilon_\epsilon^r, \mathbf{i}_\epsilon \times \mathbf{n}^\dagger)_\epsilon \\ &\quad - \sum_{\epsilon \in \mathcal{E}} (\mathbf{i}_\epsilon \times (\mathbf{n} - \bar{\mathbf{n}}_\epsilon), \Pi_1(\mathbf{r}^\dagger, \mathbf{m}^\dagger))_\epsilon + \sum_{\epsilon \in \mathcal{E}} (\Pi_2(\mathbf{r}, \mathbf{m}) - \mathbf{m}, \partial_x \mathbf{r}^\dagger - \pi \partial_x \mathbf{r}^\dagger)_\epsilon. \end{aligned}$$

Combining the above representations, we obtain, after some algebraic manipulation, that

$$\begin{aligned} \sum_{\epsilon \in \mathcal{E}} [(\epsilon_\epsilon^u, \mathbf{f}_\epsilon^\dagger)_\epsilon + (\epsilon_\epsilon^r, \mathbf{g}_\epsilon^\dagger)_\epsilon] &= - \sum_{\epsilon \in \mathcal{E}} [(C_{\mathbf{n}}^{-1}(\epsilon_\epsilon^n + \theta_\epsilon^n), \theta_\epsilon^{n^\dagger})_\epsilon + (C_{\mathbf{m}}^{-1}(\epsilon_\epsilon^m + \theta_\epsilon^m), \theta_\epsilon^{m^\dagger})_\epsilon] \\ &\quad - \sum_{\epsilon \in \mathcal{E}} [(\theta_\epsilon^n, \partial_x \mathbf{u}^\dagger - \pi \partial_x \mathbf{u}^\dagger)_\epsilon + (\theta_\epsilon^m, \partial_x \mathbf{r}^\dagger - \pi \partial_x \mathbf{r}^\dagger)_\epsilon] + \sum_{\epsilon \in \mathcal{E}} (\mathbf{i}_\epsilon \times \theta_\epsilon^r, \mathbf{n}^\dagger - \pi \mathbf{n}^\dagger) \\ &\quad - \sum_{\epsilon \in \mathcal{E}} [(\mathbf{i}_\epsilon \times (\theta_\epsilon^r + \epsilon_\epsilon^r), \theta_\epsilon^{n^\dagger})_\epsilon - (\mathbf{i}_\epsilon \times (\theta_\epsilon^n + \epsilon_\epsilon^n), \theta_\epsilon^{r^\dagger})_\epsilon], \end{aligned} \quad (4.16)$$

where the term $\pi \mathbf{n}^\dagger$ could be inserted without contribution since it is a polynomial of degree at most $p-1$.

Using the identity of norms

$$\left[\sum_{\epsilon \in \mathcal{E}} [\|\epsilon_\epsilon^u\|_\epsilon^2 + \|\epsilon_\epsilon^r\|_\epsilon^2] \right]^{1/2} = \sup_{\sum_{\epsilon \in \mathcal{E}} \|\mathbf{f}_\epsilon^\dagger\|_\epsilon^2 + \|\mathbf{g}_\epsilon^\dagger\|_\epsilon^2 = 1} \sum_{\epsilon \in \mathcal{E}} [(\epsilon_\epsilon^u, \mathbf{f}_\epsilon^\dagger)_\epsilon + (\epsilon_\epsilon^r, \mathbf{g}_\epsilon^\dagger)_\epsilon],$$

which is a consequence of the Riesz representation theorem, as well as (4.16), Lemmas 4.3 and 4.1, classical approximation properties of the L^2 -projection π , and the elliptic regularity result (4.11), we obtain with Young's and Hölder's inequalities that

$$\begin{aligned} \sum_{\epsilon \in \mathcal{E}} [\|\epsilon_\epsilon^u\|_\epsilon^2 + \|\epsilon_\epsilon^r\|_\epsilon^2] &\lesssim \sum_{\epsilon \in \mathcal{E}} (h_\epsilon + \tau_\epsilon h_\epsilon^2)^2 [\|\theta_\epsilon^n\|_\epsilon^2 + \|\theta_\epsilon^m\|_\epsilon^2 + \|\theta_\epsilon^r\|_\epsilon^2] + \sum_{\epsilon \in \mathcal{E}} (h_\epsilon + \tau_\epsilon h_\epsilon^2)^2 \|\epsilon_\epsilon^r\|_\epsilon^2 \\ &\quad + \sum_{\epsilon \in \mathcal{E}} \left(\frac{h_\epsilon}{\tau_\epsilon} \right)^2 [\|\theta_\epsilon^n\|_\epsilon^2 + \|\theta_\epsilon^r\|_\epsilon^2]. \end{aligned}$$

For sufficiently small edge lengths h_ϵ , the assertion can be concluded by absorbing the term involving ϵ_ϵ^r in the left-hand side, where we recall that $\tau_\epsilon h_\epsilon \lesssim 1$. \square

We are now ready to present the desired convergence result for the proposed HDG discretization. It provides estimates for the L^2 -errors of the primal and dual variables in terms of the maximum edge length, denoted by

$$h := \max_{\epsilon \in \mathcal{E}} h_\epsilon.$$

The rates we prove agree with those found in classical HDG theory, cf. [14].

Theorem 4.5 (Convergence of HDG method). *Suppose that for any edge $\mathfrak{e} \in \mathcal{E}$ it holds that $\mathbf{u}_{\mathfrak{e}}, \mathbf{r}_{\mathfrak{e}}, \mathbf{n}_{\mathfrak{e}}, \mathbf{m}_{\mathfrak{e}} \in H^{p+1}(\mathfrak{e})$. Then, given a stabilization parameter which scales like $h_{\mathfrak{e}}^s$ for some $s \in \{-1, 0, 1\}$, the HDG approximation converges to the solution of the Timoshenko beam network model with the error estimates*

$$\left[\sum_{\mathfrak{e} \in \mathcal{E}} [\|\mathbf{u}_{\mathfrak{e}} - \bar{\mathbf{u}}_{\mathfrak{e}}\|_{\mathfrak{e}}^2 + \|\mathbf{r}_{\mathfrak{e}} - \bar{\mathbf{r}}_{\mathfrak{e}}\|_{\mathfrak{e}}^2] \right]^{1/2} \lesssim h^{p+1-s^+}, \quad (4.17)$$

$$\left[\sum_{\mathfrak{e} \in \mathcal{E}} [\|\mathbf{n}_{\mathfrak{e}} - \bar{\mathbf{n}}_{\mathfrak{e}}\|_{\mathfrak{e}}^2 + \|\mathbf{m}_{\mathfrak{e}} - \bar{\mathbf{m}}_{\mathfrak{e}}\|_{\mathfrak{e}}^2] \right]^{1/2} \lesssim h^{p+1-|s|}, \quad (4.18)$$

where we denote $s^+ := \max(s, 0)$.

Proof. To prove the above error estimates, we use that the error can be written as the sum of a projection and a discrete error, cf. (4.2). For estimating the projection error we use Lemma 4.1, while for the discrete error Lemmas 4.4 and 4.3 can be used. For example, for the choice $\tau_{\mathfrak{e}} \sim 1$, this gives a convergence of order $p+1$ for both the projection error and the discrete errors, and thus an overall convergence of order $p+1$, cf. (4.17) and (4.18). \square

Note that using the convergence result for the dual variable, it seems possible to follow classical HDG postprocessing techniques to boost the convergence of the primal variables to order $p+2$.

5. DOMAIN DECOMPOSITION PRECONDITIONER

A practical implementation of the proposed HDG method requires solving the linear system of equations corresponding to (3.17). The linear systems of equations that typically arise when simulating, for example, the structural properties of paper-based materials, cf. Figure 1, can easily become intractable for direct solvers due to their large size. In addition, they are typically poorly conditioned, requiring appropriate preconditioners. This section introduces a two-level overlapping additive Schwarz preconditioner similar to [22], which we will use within a preconditioned conjugate gradient method.

The domain decomposition and coarse space used by the proposed preconditioner are constructed using an artificial (coarse) mesh \mathcal{T}_H of a bounding domain $\Omega \subset \mathbb{R}^3$ of the spatial network. For simplicity, we will assume that Ω is a box equipped with a uniform Cartesian mesh. Note that for thin materials such as cardboard, the corresponding artificial mesh may contain much fewer elements in one spatial dimension than in the others. With respect to the artificial mesh, we then introduce the set of trilinear basis functions $\{\varphi_i\}_{i=1}^m$, where m is the number of nodes of the artificial mesh, and corresponding supports $U_i := \text{supp}(\varphi_i)$. An illustration of an artificial mesh and corresponding basis functions in a two-dimensional setting can be found in Figure 2. The space of continuous piecewise trilinear functions with respect to \mathcal{T}_H , satisfying Dirichlet boundary conditions on boundary segments where the network nodes are fixed, is denoted by V_H . Note that the domain of the functions in V_H is considered to be the nodes of the spatial network.

We employ a preconditioner based on the subspace decomposition

$$V_{\lambda} = V_{\lambda,0} + V_{\lambda,1} + \cdots + V_{\lambda,m}$$

with the coarse space $V_{\lambda,0} := V_H \times V_H \times V_H$ and local subspaces defined for $i = 1, \dots, m$ by $V_{\lambda,i} := \{\mathbf{v} \in V_{\lambda} : \text{supp}(\mathbf{v}) \subset U_i\}$. Given this decomposition, we introduce for any subspace a corresponding subspace projection operator $P_i: V_{\lambda} \times V_{\lambda} \rightarrow V_{\lambda,i} \times V_{\lambda,i}$ such that

$$\bar{A}(P_i(\lambda, \phi), (\mu, \psi)) = \bar{A}((\lambda, \phi), (\mu, \psi))$$

holds for all $(\mu, \psi) \in V_{\lambda,i} \times V_{\lambda,i}$. Note that the existence and uniqueness of such an operator is a direct consequence of \bar{A} being an inner product on $V_{\lambda} \times V_{\lambda}$. A preconditioned version of the operator \bar{A} can then be defined as follows:

$$P := P_0 + P_1 + \cdots + P_m.$$

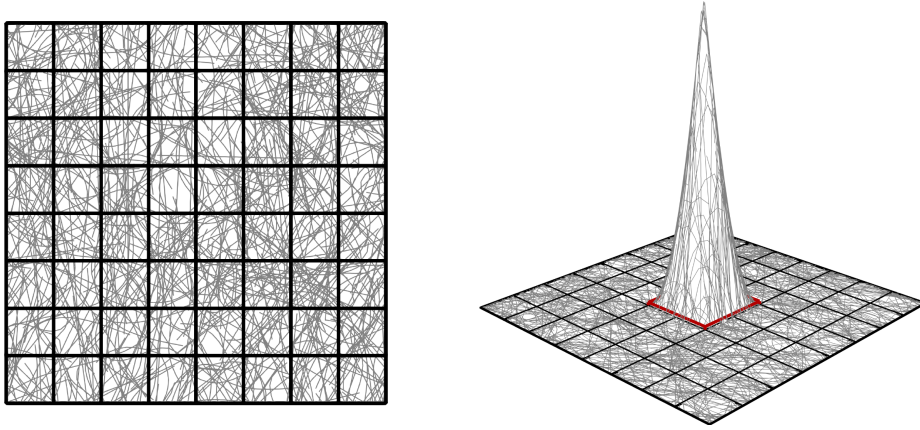


FIGURE 2. An artificial mesh \mathcal{T}_H over a network (*left*) and a basis function φ_i with the boundary of its support marked in red (*right*).

The preconditioner, denoted by B , is then given by the relation $P = BA$. Note that the preconditioner is never explicitly formed. In practice, only the preconditioned operator must be computed, which requires the direct solution of a coarse global problem and m local problems, all of which can be solved independently. Because this approach involves direct solves, it is sometimes called semi-iterative. In the following, we will use this preconditioner for the conjugate gradient method.

The uniform convergence of the resulting preconditioned conjugate gradient method was proved in Theorem 4.3 of [22] under the assumption that the considered linear system of equations is spectrally equivalent to the graph Laplacian, and under certain homogeneity, connectivity, and locality assumptions on the spatial network at coarse scales. The proof is inspired by classical Schwarz theory (see, *e.g.*, [32, 52, 54]) and constructs a quasi-interpolation operator in the spatial network setting, whose approximation and stability properties could be proved using Poincaré's and Friedrichs' inequalities on subgraphs. In practice, this preconditioner has demonstrated its ability to cope with the typically complex geometry of spatial networks and highly varying material properties; see, *e.g.*, [23]. We emphasize that, in contrast, for example standard black-box preconditioners from the class of algebraic multigrid methods, cf. [36, 55], may perform very poorly when applied to spatial network problems. This is because they do not sufficiently consider the problem's geometry.

To formulate the aforementioned spectral equivalence assumption required to prove the convergence of the preconditioned iteration, we introduce two bilinear forms acting on the space V_λ . The first bilinear form is a mass-type operator, while the second is a weighted graph Laplacian operator. They are for any functions $\lambda, \mu \in V_\lambda$ defined by

$$\mathcal{M}(\lambda, \mu) := \sum_{n \in \mathcal{N}} \frac{1}{2} \sum_{e \sim n} \lambda_n \mu_n h_e, \quad \mathcal{L}(\lambda, \mu) := \sum_{n \in \mathcal{N}} \frac{1}{2} \sum_{\substack{e \sim n \\ e = (n, n')}} \frac{(\lambda_n - \lambda_{n'}) \cdot (\mu_n - \mu_{n'})}{h_e}, \quad (5.1)$$

where the weighting with the edge length h_e is chosen to be the same as for the mass and stiffness operators in a one-dimensional finite element implementation.

The following theorem shows that the condensed problem (3.7) is spectrally equivalent to the weighted graph Laplacians defined in (5.1) in each component. This is done by explicitly computing the solutions to the one-dimensional Timoshenko beam equations, where we assume for simplicity that the material coefficients are edgewise constant. Under suitable assumptions on the discretized local solver (3.14), we expect that this result can be transferred to the condensed formulation of the HDG method in (3.17).

Theorem 5.1 (Spectral equivalence to graph Laplacian). *Assume that the maximal edge length is sufficiently small and that the material coefficients $C_{\mathbf{n}}$ and $C_{\mathbf{m}}$ are edgewise constant. Then, there holds for all $(\boldsymbol{\lambda}, \boldsymbol{\phi}) \in V_{\boldsymbol{\lambda}} \times V_{\boldsymbol{\phi}}$ that*

$$\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\lambda}) + \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\phi}) \lesssim A((\boldsymbol{\lambda}, \boldsymbol{\phi}), (\boldsymbol{\lambda}, \boldsymbol{\phi})) \lesssim \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\lambda}) + \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\phi}), \quad (5.2)$$

where the hidden constants depend only on the reciprocal of

$$\lambda_{\min} := \min_{\boldsymbol{\mu} \in V_{\boldsymbol{\lambda}} \setminus \{0\}} \frac{\mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\mu})}{\mathcal{M}(\boldsymbol{\mu}, \boldsymbol{\mu})}, \quad (5.3)$$

which characterizes the smallest eigenvalue of the generalized eigenvalue problem with the bilinear forms \mathcal{L} and \mathcal{M} defined in (5.1) on the left- and right-hand sides, respectively.

Note that λ_{\min} can be uniformly bounded from below using results from classical graph theory; see [11, 12]. The bounds typically involve the constant of a d -dimensional isoperimetric inequality, which is a measure for the connectivity of the underlying graph.

Proof. The proof is done in two steps. First, for each beam, we explicitly solve the corresponding local Timoshenko beam equations, where the Dirichlet boundary data at the beam endpoints is given and the source terms are set to zero. Second, these explicit local Timoshenko solutions are used to derive the desired spectral equivalence result.

Step 1. We consider an arbitrary but fixed edge $\mathfrak{e} \in \mathcal{E}$ and denote its associated local basis by $\{\hat{\mathbf{i}}_{\mathfrak{e}}, \hat{\mathbf{j}}_{\mathfrak{e}}, \hat{\mathbf{k}}_{\mathfrak{e}}\}$. As global basis we use the canonical basis of \mathbb{R}^3 , denoted by $\{\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}\}$. To explicitly solve the local Timoshenko equations corresponding to \mathfrak{e} we write them in local coordinates similar to Remark 2.1. In the following, we denote local variables by hats and write $\mathbf{T}_{\mathfrak{e}}$ for the change of basis matrix between local and global coordinates. Recall that the local coefficient matrices $\hat{C}_{\mathbf{n}} = \mathbf{T}_{\mathfrak{e}}^{\top} C_{\mathbf{n}} \mathbf{T}_{\mathfrak{e}}$ and $\hat{C}_{\mathbf{m}} = \mathbf{T}_{\mathfrak{e}}^{\top} C_{\mathbf{m}} \mathbf{T}_{\mathfrak{e}}$ are diagonal with positive diagonal entries, cf. Remark 2.1. The local Timoshenko equations are posed on the domain $[0, h_{\mathfrak{e}}] \times \{0\}^2$, which we parametrize with the interval $[0, h_{\mathfrak{e}}]$. We omit the subscript \mathfrak{e} to simplify the notation in the following.

Let the following Dirichlet data for the displacement and rotation be given:

$$\hat{\mathbf{u}}(0) = \hat{\boldsymbol{\lambda}}_1, \quad \hat{\mathbf{u}}(h_{\mathfrak{e}}) = \hat{\boldsymbol{\lambda}}_2, \quad \hat{\mathbf{r}}(0) = \hat{\boldsymbol{\phi}}_1, \quad \hat{\mathbf{r}}(h_{\mathfrak{e}}) = \hat{\boldsymbol{\phi}}_2. \quad (5.4)$$

Similar to the definition of the local solver in (3.1), the local source terms are set to zero. To derive the local Timoshenko solutions, we first note that $\partial_{\hat{x}} \hat{\mathbf{n}} = 0$ and hence $\hat{\mathbf{n}} = -\mathbf{c}$ for some $\mathbf{c} \in \mathbb{R}^3$. This implies for the displacement that $\partial_{\hat{x}} \hat{\mathbf{u}} = -\hat{\mathbf{i}} \times \hat{\mathbf{r}} + \hat{C}_{\mathbf{n}}^{-1} \mathbf{c}$. To derive an explicit expression for the rotation $\hat{\mathbf{r}}$ we note that $\partial_{\hat{x}} \hat{\mathbf{m}} = -\hat{\mathbf{i}} \times \hat{\mathbf{n}} = \hat{\mathbf{i}} \times \mathbf{c}$, which implies that $\hat{\mathbf{m}} = (\hat{\mathbf{i}} \times \mathbf{c}) \hat{x} - \mathbf{d}$ for some $\mathbf{d} \in \mathbb{R}^3$. The equation $\partial_{\hat{x}} \hat{\mathbf{r}} = -\hat{C}_{\mathbf{m}}^{-1} \hat{\mathbf{m}}$ can then be used to obtain an explicit expression for the rotation. Combining the above considerations yields the following expressions for the displacement and rotation:

$$\begin{aligned} \hat{\mathbf{u}} &= \frac{1}{6} \left(\hat{\mathbf{i}} \times \hat{C}_{\mathbf{m}}^{-1} (\hat{\mathbf{i}} \times \mathbf{c}) \right) \hat{x}^3 - \frac{1}{2} \left(\hat{\mathbf{i}} \times \hat{C}_{\mathbf{m}}^{-1} \mathbf{d} \right) \hat{x}^2 - \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_1 \right) \hat{x} + \hat{C}_{\mathbf{n}}^{-1} \mathbf{c} \hat{x} + \hat{\boldsymbol{\lambda}}_1, \\ \hat{\mathbf{r}} &= -\frac{1}{2} \hat{C}_{\mathbf{m}}^{-1} (\hat{\mathbf{i}} \times \mathbf{c}) \hat{x}^2 + \hat{C}_{\mathbf{m}}^{-1} \mathbf{d} \hat{x} + \hat{\boldsymbol{\phi}}_1. \end{aligned}$$

To determine the constants \mathbf{c} and \mathbf{d} , we use the second and fourth conditions of (5.4), which we have not yet incorporated in the above formula for $\hat{\mathbf{u}}$ and $\hat{\mathbf{r}}$. Below, we use the abbreviations $\hat{\boldsymbol{\lambda}}_{\Delta} := \hat{\boldsymbol{\lambda}}_2 - \hat{\boldsymbol{\lambda}}_1$, $\hat{\boldsymbol{\phi}}_{\Delta} := \hat{\boldsymbol{\phi}}_2 - \hat{\boldsymbol{\phi}}_1$, and $\hat{\boldsymbol{\phi}}_{\Sigma} := \hat{\boldsymbol{\phi}}_1 + \hat{\boldsymbol{\phi}}_2$ to simplify the notation. For \mathbf{c} we obtain, after some algebraic manipulations, the expression

$$\mathbf{c} = \frac{1}{h_{\mathfrak{e}}} \hat{C}_{\mathbf{n}} \hat{\boldsymbol{\lambda}}_{\Delta} + \frac{1}{2} \hat{C}_{\mathbf{n}} (\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_{\Sigma}) + \frac{h_{\mathfrak{e}}^2}{12} \hat{C}_{\mathbf{n}} \left(\hat{\mathbf{i}} \times \left(\hat{C}_{\mathbf{m}}^{-1} (\hat{\mathbf{i}} \times \mathbf{c}) \right) \right),$$

where \mathbf{c} still appears also on the right-hand side. We explicitly evaluate the cross products to move the term involving \mathbf{c} to the left-hand side. Introducing the matrix

$$D := \mathbf{1}_3 + \frac{h_\epsilon^2}{12} \text{diag}\left(0, \left(\hat{C}_{\mathbf{n}}\right)_{22} \left(\hat{C}_{\mathbf{m}}^{-1}\right)_{33}, \left(\hat{C}_{\mathbf{n}}\right)_{33} \left(\hat{C}_{\mathbf{m}}^{-1}\right)_{22}\right),$$

where $(\cdot)_{ij}$ denote the ij -th entry of a matrix, this results in

$$\mathbf{c} = \frac{1}{h_\epsilon} D^{-1} \hat{C}_{\mathbf{n}} \hat{\boldsymbol{\lambda}}_\Delta + \frac{1}{2} D^{-1} \hat{C}_{\mathbf{n}} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right).$$

This equation can then be used to derive the following explicit expression for \mathbf{d} :

$$\mathbf{d} = \frac{1}{h_\epsilon} \hat{C}_{\mathbf{m}} \hat{\boldsymbol{\phi}}_\Delta + \frac{1}{2} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \hat{\boldsymbol{\lambda}}_\Delta \right) \right) + \frac{h_\epsilon}{4} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right) \right).$$

Inserting the above expressions for the constants \mathbf{c} and \mathbf{d} in the equations $\hat{\mathbf{n}} = -\mathbf{c}$ and $\hat{\mathbf{m}} = (\hat{\mathbf{i}} \times \mathbf{c})\hat{x} - \mathbf{d}$, we obtain for the forces and moments that

$$\begin{aligned} \hat{\mathbf{n}} &= -\frac{1}{h_\epsilon} D^{-1} \hat{C}_{\mathbf{n}} \hat{\boldsymbol{\lambda}}_\Delta - \frac{1}{2} D^{-1} \hat{C}_{\mathbf{n}} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right), \\ \hat{\mathbf{m}} &= \frac{1}{2h_\epsilon} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \hat{\boldsymbol{\lambda}}_\Delta \right) \right) (2\hat{x} - h_\epsilon) + \frac{1}{4} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right) \right) (2\hat{x} - h_\epsilon) - \frac{1}{h_\epsilon} \hat{C}_{\mathbf{m}} \hat{\boldsymbol{\phi}}_\Delta. \end{aligned}$$

To establish an expression for the bilinear form A in terms of the Dirichlet data at the nodes, we will use its equivalent representation given in Lemma 3.3. This representation includes integrals involving the forces and moments, which can be computed as follows:

$$\begin{aligned} \int_0^{h_\epsilon} \hat{C}_{\mathbf{n}}^{-1} \hat{\mathbf{n}} \cdot \hat{\mathbf{n}} \, d\hat{x} &= \frac{1}{h_\epsilon} \left| D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \hat{\boldsymbol{\lambda}}_\Delta \right|^2 + D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \hat{\boldsymbol{\lambda}}_\Delta \cdot D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \\ &\quad + \frac{h_\epsilon}{4} \left| D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right|^2, \\ \int_0^{h_\epsilon} \hat{C}_{\mathbf{m}}^{-1} \hat{\mathbf{m}} \cdot \hat{\mathbf{m}} \, d\hat{x} &= \frac{1}{h_\epsilon} \left| \hat{C}_{\mathbf{m}}^{1/2} \hat{\boldsymbol{\phi}}_\Delta \right|^2 + \frac{h_\epsilon}{12} \left| \hat{C}_{\mathbf{m}}^{-1/2} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \hat{\boldsymbol{\lambda}}_\Delta \right) \right) \right|^2 \\ &\quad + \frac{h_\epsilon}{48} \left| \hat{C}_{\mathbf{m}}^{-1/2} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right) \right) \right|^2 \\ &\quad + \frac{h_\epsilon^2}{12} \left(\hat{C}_{\mathbf{m}}^{-1/2} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \hat{\boldsymbol{\lambda}}_\Delta \right) \right) \right) \cdot \left(\hat{C}_{\mathbf{m}}^{-1/2} \left(\hat{\mathbf{i}} \times \left(D^{-1} \hat{C}_{\mathbf{n}} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right) \right) \right). \end{aligned}$$

To prove the lower bound in (5.2), we sum up the previously computed integrals and bound the result from below using the weighted Young inequality. This yields for any $0 < \epsilon < 1$, which we will specify later, that

$$\begin{aligned} \int_0^{h_\epsilon} \hat{C}_{\mathbf{n}}^{-1} \hat{\mathbf{n}} \cdot \hat{\mathbf{n}} \, d\hat{x} + \int_0^{h_\epsilon} \hat{C}_{\mathbf{m}}^{-1} \hat{\mathbf{m}} \cdot \hat{\mathbf{m}} \, d\hat{x} &\geq \frac{1}{h_\epsilon} \left| D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \hat{\boldsymbol{\lambda}}_\Delta \right|^2 + \frac{1}{h_\epsilon} \left| \hat{C}_{\mathbf{m}}^{1/2} \hat{\boldsymbol{\phi}}_\Delta \right|^2 + \frac{h_\epsilon}{4} \left| D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right|^2 \\ &\quad + \left(D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \hat{\boldsymbol{\lambda}}_\Delta \right) \cdot \left(D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right) \\ &\geq \frac{1-\epsilon}{h_\epsilon} \left| D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \hat{\boldsymbol{\lambda}}_\Delta \right|^2 + \frac{1}{h_\epsilon} \left| \hat{C}_{\mathbf{m}}^{1/2} \hat{\boldsymbol{\phi}}_\Delta \right|^2 - \frac{h_\epsilon(1-\epsilon)}{4\epsilon} \left| D^{-1} \hat{C}_{\mathbf{n}}^{1/2} \left(\hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right) \right|^2. \end{aligned}$$

Using the uniform bounds on the coefficients (2.1) and the assumption that the edge length h_ϵ is sufficiently small, the latter estimate can be simplified to

$$\int_0^{h_\epsilon} \hat{C}_{\mathbf{n}}^{-1} \hat{\mathbf{n}} \cdot \hat{\mathbf{n}} \, d\hat{x} + \int_0^{h_\epsilon} \hat{C}_{\mathbf{m}}^{-1} \hat{\mathbf{m}} \cdot \hat{\mathbf{m}} \, d\hat{x} \gtrsim \frac{1-\epsilon}{h_\epsilon} \left| \hat{\boldsymbol{\lambda}}_\Delta \right|^2 + \frac{1}{h_\epsilon} \left| \hat{\boldsymbol{\phi}}_\Delta \right|^2 - C \frac{h_\epsilon(1-\epsilon)}{4\epsilon} \left| \hat{\mathbf{i}} \times \hat{\boldsymbol{\phi}}_\Sigma \right|^2, \quad (5.5)$$

where $C > 0$ only depends on the bounds of the coefficients.

Step 2. Estimate (5.5) was formulated in local coordinates, but it also holds in global coordinates, *i.e.*, without hats. This is a direct consequence of the orthogonality of the change-of-basis matrix \mathbf{T}_ϵ . Summing up the global estimates for all edges then yields that

$$\begin{aligned} \sum_{\epsilon \in \mathcal{E}} \int_{\epsilon} C_{\mathbf{n}}^{-1} \mathbf{n}_\epsilon \cdot \mathbf{n}_\epsilon \, d\sigma + \sum_{\epsilon \in \mathcal{E}} \int_{\epsilon} C_{\mathbf{m}}^{-1} \mathbf{m}_\epsilon \cdot \mathbf{m}_\epsilon \, d\sigma \\ \gtrsim \sum_{\substack{\epsilon \in \mathcal{E} \\ \epsilon = (\mathbf{n}, \mathbf{n}')}} \left(\frac{1-\epsilon}{h_\epsilon} |\boldsymbol{\lambda}_{\mathbf{n}} - \boldsymbol{\lambda}_{\mathbf{n}'}|^2 + \frac{1}{h_\epsilon} |\boldsymbol{\phi}_{\mathbf{n}} - \boldsymbol{\phi}_{\mathbf{n}'}|^2 - C \frac{h_\epsilon(1-\epsilon)}{4\epsilon} |\mathbf{i}_\epsilon \times (\boldsymbol{\phi}_{\mathbf{n}} + \boldsymbol{\phi}_{\mathbf{n}'})|^2 \right). \end{aligned} \quad (5.6)$$

Rearranging the above sum into a sum over all nodes and another sum over all edges adjacent to the nodes, one observes that the first and second terms in (5.6) can be written in terms of the bilinear form \mathcal{L} of the graph Laplacian defined in (5.1). Similarly, the last term in (5.6) can be estimated using the mass-type operator \mathcal{M} . This and using the definition of λ_{\min} in (5.3) yields that

$$\begin{aligned} (5.6) &\geq (1 - \epsilon) \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\lambda}) + \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\phi}) - C \frac{1-\epsilon}{\epsilon} \mathcal{M}(\boldsymbol{\phi}, \boldsymbol{\phi}) \\ &\geq (1 - \epsilon) \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\lambda}) + (1 - C \frac{1-\epsilon}{\epsilon} \lambda_{\min}^{-1}) \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\phi}). \end{aligned}$$

An investigation shows that for any $\epsilon \in (\frac{C}{\lambda_{\min} + C}, 1)$, the terms in the above estimate are positive, which proves the lower bound in (5.2). The upper bound can be proved using similar arguments, and the proof is omitted for brevity. This concludes the proof. \square

6. NUMERICAL EXPERIMENTS

In this section, we present numerical experiments that support the theoretical predictions of this paper. The numerical experiments have been implemented in the HyperHDG software package [48], which is described in detail in [49]. There, numerical experiments demonstrating the convergence of the HDG method for diffusion-type problems on hypergraphs were presented. The code to reproduce the numerical experiments of this paper is available at <https://github.com/HyperHDG/>, and the data describing the fiber network model of paper used in the second numerical experiment can be found at [26].

Optimal order convergence of HDG method

The first numerical example considers a toy problem to study the convergence properties of the proposed HDG method for Timoshenko beam networks. Specifically, we consider a network representing a two-dimensional unit cross embedded in the three-dimensional Euclidean space, *i.e.*,

$$([-1, 1] \times \{0\} \cup \{0\} \times [-1, 1]) \times \{0\}.$$

Before mesh refinement, the network consists of four edges and five vertices. Dirichlet boundary conditions are imposed at the four nodes located at the tips of the cross. The bounding domain is chosen to be $\Omega = [-1, 1]^2 \times \{0\}$. To construct suitable data and corresponding solutions, we use the method of manufactured solutions. More precisely, we consider homogeneous material coefficients, *i.e.*, $C_{\mathbf{n}} = C_{\mathbf{m}} = \mathbb{1}$, and choose the force terms such that the problem admits the following displacements and rotations as its solutions:

$$\mathbf{u}(x, y, z) = \begin{pmatrix} 0 \\ \cos(\pi y) \\ \cos(\pi x) \end{pmatrix}, \quad \mathbf{r}(x, y, z) = \begin{pmatrix} 0 \\ \sin(\pi x) \\ \sin(\pi y) \end{pmatrix}.$$

Figure 3 shows convergence plots for the L^2 -error of the primal HDG variables, which we will subsequently abbreviate by $\text{err}_{L^2}(\mathbf{u}, \mathbf{r})$, cf. (4.17). In the top right and bottom left convergence plots, one observes optimal

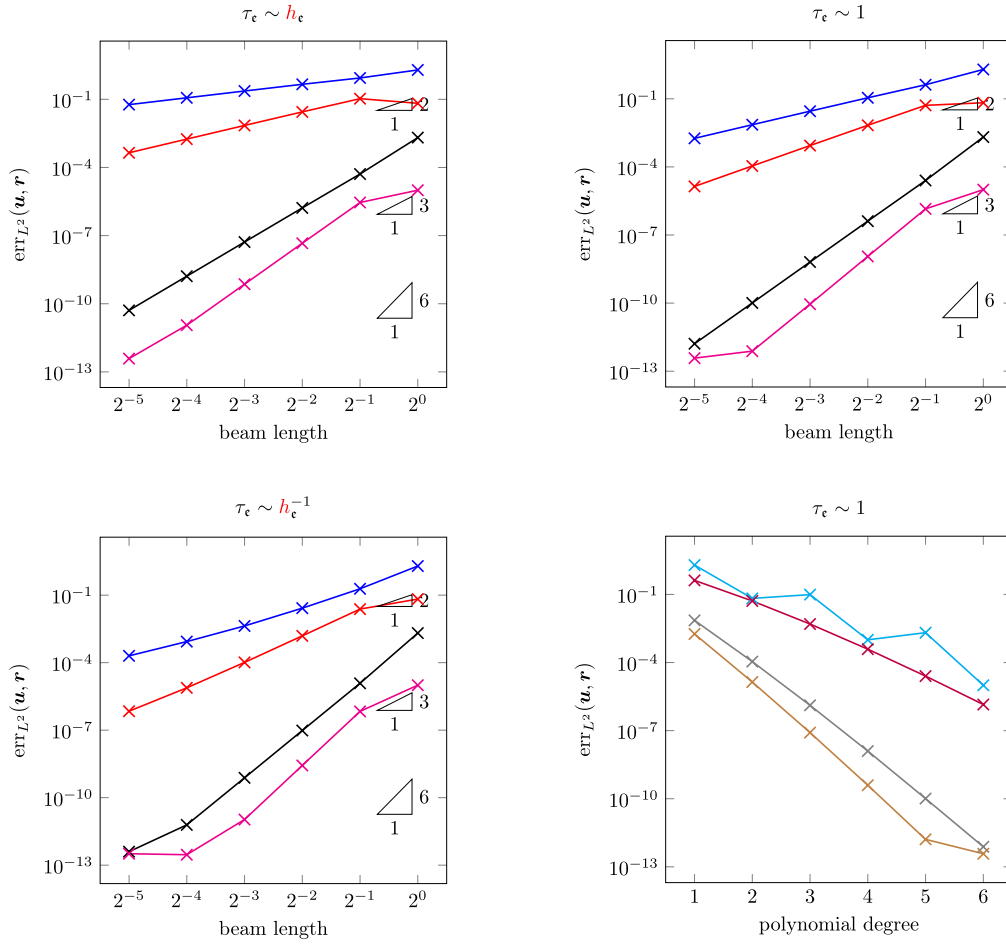


FIGURE 3. All four plots show the L^2 -errors of the primal variables. The top left, top right, and bottom left plots show the errors for the polynomial degrees 1 (blue), 2 (red), 5 (black), and 6 (magenta) as a function of the mesh size, for different choices of the stabilization parameter τ . The bottom right plot shows the errors for the beam lengths 1 (cyan), 2^{-1} (purple), 2^{-4} (gray), and 2^{-5} (brown) as a function of the polynomial degree.

$(p + 1)$ -th order convergence for the classical choices of stabilization parameters $\tau_\epsilon \sim 1$ and $\tau_\epsilon \sim h_\epsilon^{-1}$. Note that for $\tau_\epsilon \sim h_\epsilon^{-1}$ one encounters a pre-asymptotic regime with even faster convergence. For the stabilization parameter $\tau_\epsilon \sim h_\epsilon$ one observes a suboptimal p -th order convergence, cf. Figure 3 (top left). These convergence results are consistent with the theoretical predictions of Theorem 4.5. Note that the convergence rates we obtain are also consistent with those found in the literature. Convergence tests for the dual variables also confirm the rates predicted by Theorem 4.5. For the sake of brevity, however, the corresponding plots are not shown here. In addition to the convergence under mesh refinement, we also want to numerically investigate the convergence as the polynomial degree is increased. The corresponding plot can be found at the bottom right in Figure 3. One observes an exponential convergence as the polynomial degree is increased, with the rate of decay depending on the fixed mesh size. A similar exponential convergence behavior can also be observed for the dual variables.

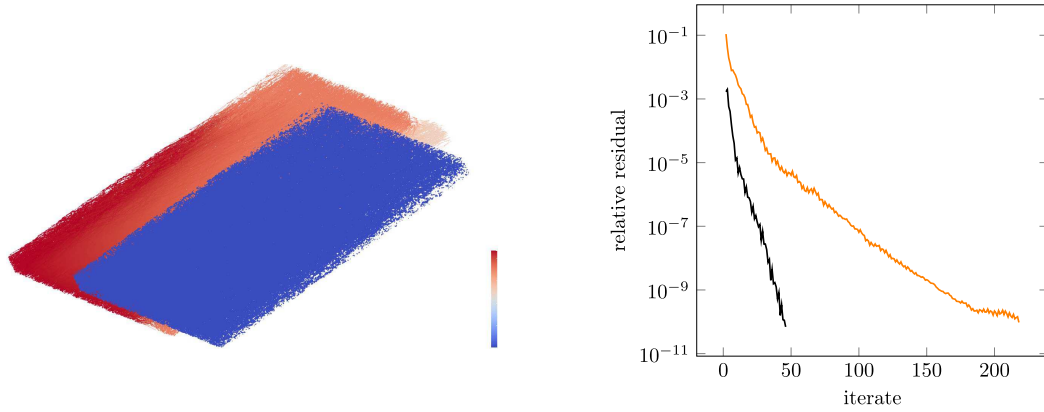


FIGURE 4. *Left*: an illustration of the network before (blue) and after (red) deformation is shown. *Right*: a convergence plot of the preconditioned conjugate gradient method. There the relative residual is plotted as a function of the iteration number for constant material parameters (black) and realistic material parameters (orange).

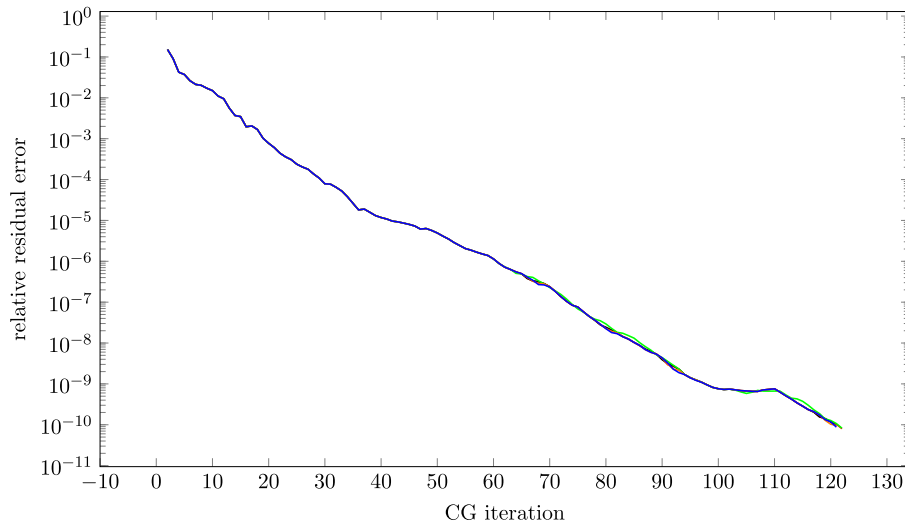


FIGURE 5. Relative residual of the preconditioned conjugate gradient method as a function of the iteration number for different polynomial degrees p and stabilization parameters τ . The black, red, and green curves correspond to $p = 1, 5, 10$ for $\tau = 1$, while the orange and blue curves represent $\tau = 1000, 0.001$ for $p = 5$.

Elastic deformation of paper

The purpose of the second numerical experiment is to demonstrate the applicability of the proposed HDG method to a realistic example. We consider the elastic deformation of about $2\text{ mm} \times 4\text{ mm}$ piece of paper, where our collaborators at the Fraunhofer-Chalmers Centre (FCC) provided the corresponding spatial network and material parameters. The spatial network consists of about 615K edges and 424K nodes. We consider the stretching of the paper caused by inhomogeneous Dirichlet boundary conditions at nodes at the lateral boundary. We use a HDG discretization with polynomial degree 5 for all edges. The resulting linear system of equations

is then solved by the preconditioned conjugate gradient method with the domain decomposition preconditioner as described in Section 5. For the preconditioner, we use an artificial coarse mesh consisting of 64 elements, where only one element is used in the z -direction. The local subproblems arising within the preconditioner are solved with a conjugate gradient method without preconditioning using a termination criterion of a relative residual of 10^{-3} . This is practically feasible since we use a mild termination criterion. The computations were performed in parallel on a cluster with 64 cores. For the case of constant material parameters, our solver needed 46 iterations and about 9 h to solve the problem up to a relative residual of 10^{-10} . For the realistic parameters, it took 218 iterations and about 33 h to reach the same accuracy. The resulting deformed piece of paper and the convergence plot of the preconditioned conjugate gradient method are shown in Figure 4.

Influence of polynomial degree and stabilization parameter

We next examine the influence of the polynomial degree p and the stabilization parameter τ on the convergence behavior of the preconditioned conjugate gradient method. To this end, we consider a smaller network consisting of 14K edges and 10K nodes, using realistic material parameters similar to the previous numerical experiment. As shown in Figure 5, varying p and τ have almost no impact on the convergence. It is worth noting that the iteration counts are comparable to those observed in the previous experiment. However, due to the significantly smaller network size, the computational cost per iteration is considerably smaller.

ACKNOWLEDGMENTS

We would like to thank Morgan Görtz (Fraunhofer-Chalmers Center) for many fruitful discussions on the application of fiber-based materials, for providing some of the figures, and for generating the networks used in the second and third numerical experiments.

FUNDING

M. Hauck has been supported by the Knut and Alice Wallenberg foundation postdoctoral program in mathematics for researchers from outside Sweden, grant number KAW 2022.0260 and A. Målvist by the Swedish research council, grant number 2023-03258.VR. A. Rupp has been supported by the Research Council of Finland's decision numbers 350101, 354489, 359633, 358944 and Business Finland's project number 539/31/2023.

DATA AVAILABILITY STATEMENT

The research data associated with this article are included in the article.

REFERENCES

- [1] R. Altmann, P. Henning and D. Peterseim, Numerical homogenization beyond scale separation. *Acta Numer.* **30** (2021) 1–86.
- [2] I. Babuska and R. Lipton, Optimal local approximation spaces for generalized finite element methods with application to multiscale problems. *Multiscale Model. Simul.* **9** (2011) 373–406.
- [3] O.A. Bauchau and J.I. Craig, Structural Analysis. Springer Netherlands (2009).
- [4] D. Boffi, F. Brezzi and M. Fortin, Mixed Finite Element Methods and Applications. Springer Berlin Heidelberg (2013).
- [5] E. Carrera, G. Giunta and M. Petrolo, Beam Structures: Classical and Advanced Theories. Wiley (2011).
- [6] F. Celiker and B. Cockburn, Element-by-element post-processing of discontinuous Galerkin methods for Timoshenko beams. *J. Sci. Comput.* **27** (2006) 177–187.
- [7] F. Celiker, B. Cockburn and H.K. Stolarski, Locking-free optimal discontinuous Galerkin methods for Timoshenko beams. *SIAM J. Numer. Anal.* **44** (2006) 2297–2325.
- [8] F. Celiker, B. Cockburn and K. Shi, Hybridizable discontinuous Galerkin methods for Timoshenko beams. *J. Sci. Comput.* **44** (2010) 1–37.
- [9] F. Celiker, B. Cockburn and K. Shi, A projection-based error analysis of HDG methods for Timoshenko beams. *Math. Comput.* **81** (2012) 131–151.
- [10] J. Chu, B. Engquist, M. Prodanović and R. Tsai, A multiscale method coupling network and continuum models in porous media I: steady-state single phase flow. *Multiscale Model. Simul.* **10** (2012) 515–549.

- [11] F.R.K. Chung, Laplacians and the cheeger inequality for directed graphs. *Ann. Comb.* **9** (2005) 1–19.
- [12] F.R.K. Chung and S.-T. Yau, Eigenvalues of graphs and Sobolev inequalities. *Comb. Probab. Comput.* **4** (1995) 11–25.
- [13] B. Cockburn, J. Gopalakrishnan and R. Lazarov, Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. *SIAM J. Numer. Anal.* **47** (2009) 1319–1365.
- [14] B. Cockburn, J. Gopalakrishnan and F.-J. Sayas, A projection-based error analysis of HDG methods. *Math. Comput.* **79** (2010) 1351–1367.
- [15] B. Cockburn, O. Dubois, J. Gopalakrishnan and S. Tan, Multigrid for an HDG method. *IMA J. Numer. Anal.* **34** (2013) 1386–1425.
- [16] R. Davis, R.D. Henshell and G.B. Warburton, A Timoshenko beam element. *J. Sound Vib.* **22** (1972) 475–487.
- [17] F. Edelvik, M. Görtz, F. Hellman, G. Kettil and A. Målqvist, Numerical homogenization of spatial network models. *Comput. Methods Appl. Mech. Eng.* **418** (2024) 116593.
- [18] M.S. Fabien, M.G. Knepley, R.T. Mills and B.M. Rivière, Manycore parallel computing for a hybridizable discontinuous Galerkin nested multigrid method. *SIAM J. Sci. Comput.* **41** (2019) C73–C96.
- [19] P. Freese, M. Hauck, T. Keil and D. Peterseim, A super-localized generalized finite element method. *Numer. Math.* **156** (2024) 205–235.
- [20] M. Fritz, T. Köppl, J. T. Oden, A. Wagner and B. Wohlmuth, A 1D–0D–3D coupled model for simulating blood flow and transport processes in breast tissue. *Int. J. Numer. Method Biomed. Eng.* **38** (2022) e3612.
- [21] M. Görtz, G. Kettil, A. Målqvist, M. Fredlund, K. Wester and F. Edelvik, Network model for predicting structural properties of paper. *Nord. Pulp Pap. Res. J.* **37** (2022) 712–724.
- [22] M. Görtz, F. Hellman and A. Målqvist, Iterative solution of spatial network models by subspace decomposition. *Math. Comput.* **93** (2024) 233–258.
- [23] M. Görtz, G. Kettil, A. Målqvist, M. Fredlund and F. Edelvik, Iterative method for large-scale Timoshenko beam models assessed on commercial-grade paperboard. *Comput. Mech.* **75** (2025) 1–13.
- [24] M. Hauck and A. Målqvist, Super-localization of spatial network models. *Numer. Math.* **156** (2024) 901–926.
- [25] M. Hauck and D. Peterseim, Super-localization of elliptic multiscale problems. *Math. Comput.* **92** (2023) 981–1003.
- [26] M. Hauck and A. Rupp, Fiber network models of paper. *Zenodo* (2024). <https://doi.org/10.5281/zenodo.12751486>.
- [27] M. Hauck, R. Maier and A. Målqvist, An algebraic multiscale method for spatial network models. Preprint [arXiv:2312.09752](https://arxiv.org/abs/2312.09752) (2023).
- [28] P. Henning and D. Peterseim, Oversampling for the multiscale finite element method. *Multiscale Model. Simul.* **11** (2013) 1149–1175.
- [29] G. Jelenić and E. Papa, Exact solution of 3d Timoshenko beam problem using linked interpolation of arbitrary order. *Arch. Appl. Mech.* **81** (2009) 171–183.
- [30] K.K. Kapur, Vibrations of a Timoshenko beam, using finite-element approach. *J. Acoust. Soc. Am.* **40** (1966) 1058–1063.
- [31] G. Kettil, A. Målqvist, A. Mark, M. Fredlund, K. Wester and F. Edelvik, Numerical upscaling of discrete network models. *BIT* **60** (2020) 67–92.
- [32] R. Kornhuber and H. Yserentant, Numerical homogenization of elliptic multiscale problems by subspace decomposition. *Multiscale Model. Simul.* **14** (2016) 1017–1036.
- [33] T. Kufner, G. Leugering, J. Semmler, M. Sting and C. Strohmeier, Simulation and structural optimization of 3d Timoshenko beam networks based on fully analytic network solutions. *ESAIM Math. Model. Numer. Anal. (M2AN)* **52** (2018) 2409–2431.
- [34] J.E. Lagnese, G. Leugering and E.J.P.G. Schmidt, Modeling, Analysis and Control of Dynamic Elastic Multi-Link Structures. Birkhäuser Boston (1994).
- [35] A.W. Lees and D.L. Thomas, Unified Timoshenko beam finite element. *J. Sound Vib.* **80** (1982) 355–366.
- [36] O.E. Livne and A. Brandt, Lean algebraic multigrid (LAMG): fast graph Laplacian linear solver. *SIAM J. Sci. Comput.* **34** (2012) B499–B522.
- [37] P. Lu, A. Rupp and G. Kanschat, Homogeneous multigrid for HDG. *IMA J. Numer. Anal.* **42** (2021) 3135–3153.
- [38] P. Lu, A. Rupp and G. Kanschat, Analysis of injection operators in geometric multigrid solvers for HDG methods. *SIAM J. Numer. Anal.* **60** (2022) 2293–2317.
- [39] P. Lu, A. Rupp and G. Kanschat, Two-level Schwarz methods for hybridizable discontinuous Galerkin methods. *J. Sci. Comput.* **95** (2023) 9.

- [40] C. Ma, R. Scheichl and T. Dodwell, Novel design and analysis of generalized finite element methods based on locally optimal spectral approximations. *SIAM J. Numer. Anal.* **60** (2022) 244–273.
- [41] A. Målqvist and D. Peterseim, Localization of elliptic multiscale problems. *Math. Comput.* **83** (2014) 2583–2603.
- [42] A. Målqvist and D. Peterseim, Numerical Homogenization by Localized Orthogonal Decomposition. SIAM, Philadelphia, PA (2020).
- [43] S. Mukherjee and G. Prathap, Analysis of shear locking in Timoshenko beam elements using the function space approach. *Commun. Numer. Methods Eng.* **17** (2001) 385–393.
- [44] H. Owhadi, Multigrid with rough coefficients and multiresolution operator decomposition from hierarchical information games. *SIAM Rev.* **59** (2017) 99–149.
- [45] H. Owhadi and C. Scovel, Operator-Adapted Wavelets, Fast Solvers, and Numerical Homogenization. Vol. 35 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press (2019).
- [46] G. Prathap and G.R. Bhashyam, Reduced integration and the shear-flexible beam element. *Int. J. Num. Meth. Eng.* **18** (1982) 195–210.
- [47] J.N. Reddy, On locking-free shear deformable beam finite elements. *Comput. Methods Appl. Mech. Eng.* **149** (1997) 113–132.
- [48] A. Rupp and G. Kanschat, HyperHDG: Hybrid discontinuous Galerkin methods for PDEs on hypergraphs (2021). Published online. <https://github.com/HyperHDG>.
- [49] A. Rupp, M. Gahn and G. Kanschat, Partial differential equations on hypergraphs and networks of surfaces: derivation and hybrid discretizations. *ESAIM Math. Model. Numer. Anal. (M2AN)* **56** (2022) 505–528.
- [50] J. Thomas and B.A.H. Abbas, Finite element model for dynamic analysis of Timoshenko beam. *J. Sound Vib.* **41** (1975) 291–299.
- [51] S.P. Timoshenko, On the correction for shear of the differential equation for transverse vibrations of prismatic bars. *London Edinburgh Philos. Mag. & J. Sci.* **41** (1921) 744–746.
- [52] A. Toselli and O. Widlund, Domain Decomposition Methods—Algorithms and Theory. Vol. 34 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin (2005).
- [53] T. Wildey, S. Muralikrishnan and T. Bui-Thanh, Unified geometric multigrid algorithm for hybridized high-order finite element methods. *SIAM J. Sci. Comput.* **41** (2019) S172–S195.
- [54] J. Xu, Iterative methods by space decomposition and subspace correction. *SIAM Rev.* **34** (1992) 581–613.
- [55] J. Xu and L. Zikatanov, Algebraic multigrid methods. *Acta Numer.* **26** (2017) 591–721.
- [56] T. Yokoyama, A reduced integration Timoshenko beam element. *J. Sound Vib.* **169** (1994) 411–418.



Please help to maintain this journal in open access!

This journal is currently published in open access under the Subscribe to Open model (S2O). We are thankful to our subscribers and supporters for making it possible to publish this journal in open access in the current year, free of charge for authors and readers.

Check with your library that it subscribes to the journal, or consider making a personal donation to the S2O programme by contacting subscribers@edpsciences.org.

More information, including a list of supporters and financial transparency reports, is available at <https://edpsciences.org/en/subscribe-to-open-s2o>.