Research Article

Rainer Schneckenleitner* and Stefan Takacs

Convergence Theory for IETI-DP Solvers for Discontinuous Galerkin Isogeometric Analysis that is Explicit in h and p

<https://doi.org/10.1515/cmam-2020-0164>

Received October 13, 2020; revised July 5, 2021; accepted July 13, 2021

Abstract: In this paper, we develop a convergence theory for Dual-Primal Isogeometric Tearing and Inter-connecting (IETI-DP) solvers for isogeometric multi-patch discretizations of the Poisson problem, where the patches are coupled using discontinuous Galerkin. The presented theory provides condition number bounds that are explicit in the grid sizes h and in the spline degrees p . We give an analysis that holds for various choices for the primal degrees of freedom: vertex values, edge averages, and a combination of both. If only the vertex values or both vertex values and edge averages are taken as primal degrees of freedom, the condition number bound is the same as for the conforming case. If only the edge averages are taken, both the convergence theory and the experiments show that the condition number of the preconditioned system grows with the ratio of the grid sizes on neighboring patches.

Keywords: Isogeometric Analysis, FETI-DP, Discontinuous Galerkin, p -Analysis

MSC 2010: 65D07, 65F08, 65N22, 65N55

1 Introduction

Isogeometric Analysis (IgA), see [9, 20], is an approach for discretizing partial differential equations (PDEs) that has been developed in order to improve the compatibility between computer aided design (CAD) and simulation in comparison to the standard finite element method (FEM). In CAD systems, the geometry is usually parameterized in terms of geometry functions which are usually spanned by B-splines or non-uniform rational B-splines (NURBS). In IgA, we do not mesh the computational domain, but we use the geometry function to transfer the integrals that occur in the variational formulation of the boundary value problem to the parameter domain, where we set up B-splines or NURBS bases to discretize the problem. Since only simple domains can be represented by just one geometry function (single-patch case), the overall computational domain is usually decomposed into multiple patches, each of which is parameterized by its own geometry function (multi-patch IgA). We focus on non-overlapping patches.

The patches can be coupled either in a conforming way or by means of discontinuous Galerkin methods. For a conforming discretization, both the geometry function and the discretization have to agree on the interfaces between the patches. One promising alternative to overcome these restrictions are discontinuous Galerkin (dG) approaches, cf. [6, 31], where the continuity of the solution at the patch boundaries is only enforced weakly. We consider in this paper in particular the symmetric interior penalty discontinuous Galerkin (SIPG) method, cf. [5]. The idea of applying this technique to couple patches in IgA has been pre-

*Corresponding author: Rainer Schneckenleitner, Institute of Computational Mathematics, Johannes Kepler University Linz, Linz, Austria, e-mail: schneckenleitner@numa.uni-linz.ac.at

Stefan Takacs, Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Linz, Austria, e-mail: stefan.takacs@ricam.oeaw.ac.at

viously discussed in [24, 25, 38]. Such approaches also allow the handling of inexact representations of the geometry that yield gaps between or overlaps of neighboring patches, cf. [19]. For discontinuous Galerkin methods, the discretization error decays in the grid size with the same order as for a conforming discretization, cf. [38]. However, dG methods offer much greater flexibility for the setup of the geometry.

After the discretization of the PDE, we obtain a large-scale linear system, whose condition number grows exponentially in the spline degree p and (like other FEM like methods) like h^{-2} with the grid size h . We are interested in fast iterative solvers for such systems. Since we are in a multi-patch framework, domain decomposition solvers are a canonical choice. One of the most popular domain decomposition solvers for large-scale systems of finite element equations in standard FEM is the Finite Element Tearing and Interconnecting method (FETI), originally proposed in [14]. Since the invention of FETI, various FETI-type methods have been developed, cf. [23, 30, 39]. In [22], it was proposed to use a FETI-type method, namely the Dual-Primal FETI method (FETI-DP), in the context of IgA. This method was called Dual-Primal Isogeometric Tearing and Interconnecting (IETI-DP) method. Later, this approach has been further analyzed, particularly in [18] and more recently in [32]. The original results, cf. [18], are only explicit in the ratio between grid and patch sizes. In [32], the authors of the paper at hand have analyzed the dependence of the condition number of the preconditioned Schur complement system also on the spline degree p . In [33], the authors did a numerical study of 3D problems. The experiments indicate a similar condition number bound as established in [34]. However, the extension of the analysis to 3D problems is known to be a non-trivial task and is ongoing research.

For conforming FEM discretizations, condition number bounds that are explicit in the polynomial degree p have been worked out previously for FETI-DP type, cf. [21, 28], other Schwarz type, cf. [15, 36], and iterative substructuring methods, cf. [2, 29].

The extension of FETI methods or IETI methods to dG discretizations is not straight-forward. We follow the approaches that have been proposed in [12] and adapted for IETI in [16], particularly the idea of using artificial interfaces. For dG discretizations, there are only a few results concerning a p -analysis for domain decomposition approaches, like [3, 35] for two-level Schwarz type approaches or [8, 11], where the estimates for BDDC and FETI-DP type methods for spectral element methods and hp -FEM are given. The publication [4] shows a bound for general non-overlapping Schwarz preconditioners for dG hp -FEM that is in the order of p^2 . A polylogarithmic bound for a BDDC preconditioner for a hybridizable discontinuous Galerkin discretization has been developed in [10].

In this paper, we consider the Poisson problem on planar domains. We consider an IETI-DP solver with a scaled Dirichlet preconditioner. The proof follows the abstract framework from [26] and is a continuation of the paper [32], where we have analyzed IETI-DP methods for conforming discretizations. The presented theory covers three different choices for the primal degrees of freedom: vertex values only (Algorithm A), edge averages only (Algorithm B), and the combination of both (Algorithm C). We prove that the condition of the preconditioned IETI-DP system is bounded by

$$Cp \left(1 + \log p + \max_{k=1,\dots,K} \log \frac{H_k}{h_k} \right)^2$$

for Algorithms A and C and by

$$C\delta p \left(\max_{k=1,\dots,K} \max_{\ell \in \mathcal{N}_\Gamma(k)} \frac{h_k}{h_\ell} \right) \left(1 + \log p + \max_{k=1,\dots,K} \log \frac{H_k}{h_k} \right)^2$$

for Algorithm B, where p is the spline degree, h_k are the grid sizes and H_k are the patch sizes, $\mathcal{N}_\Gamma(k)$ contains the indices of the patches that share an edge with the k -th patch, and $\delta \geq \delta^*$ is a suitably chosen penalty parameter. Both the constant C and the optimal penalty parameter $\delta^* > 0$ are independent of the grid sizes, the patch sizes, the spline degree, and the smoothness of the splines. Hence, the theory covers all discretizations where the smoothness within the patches is between C^0 and C^{p-1} . Up to the knowledge of the authors, so far, there is no condition number estimate that is explicit in the spline degree p . Moreover, we are not aware of an analysis that covers Algorithm B. Such approaches might be of interest if the overall computational domain is decomposed into patches in a way that allows T-junctions or in the case of moving patches like in the case of rotating electrical machines. Note that the constant C depends on the chosen parameterization of the geometry. Numerical experiments indicate that the convergence of the IETI-DP solver only suffers

mildly from distorted geometries, see, e.g., [34]. A theoretical analysis of the dependence on the geometry is not known to the authors.

The structure of this paper is as follows. In Section 2, we introduce our model problem and its discretization using SIPG. Then, in Section 3, we present the IETI-DP solver. Section 4 is devoted to the proof of the condition number bounds. The numerical results are shown in Section 5. In Section 6, we summarize our findings and give some further remarks.

2 The Model Problem and Its Discretization

We consider the discretization of a homogeneous Poisson problem using multi-patch Isogeometric Analysis, where the coupling of the numerical solution between the individual patches is realized using an SIPG approach. In Section 2.1, we introduce the model problem. The representation of the computational domain is discussed in Section 2.2. The function spaces are introduced in Section 2.3; the variational formulation of the discretized problem is then given in Section 2.4.

Since this paper extends the results of the previous paper [32], which covered conforming discretizations, we aim to use the same notation as in the aforementioned paper. To keep the paper self-contained, we briefly reintroduce the notation.

2.1 The Model Problem

We consider an open, bounded, and simply connected domain $\Omega \subset \mathbb{R}^2$ with Lipschitz boundary $\partial\Omega$. We are interested in solving the homogeneous Poisson problem: given $f \in L_2(\Omega)$, find $u \in H_0^1(\Omega)$ such that

$$\int_{\Omega} \nabla u \cdot \nabla v \, dx = \int_{\Omega} f v \, dx \quad \text{for all } v \in H_0^1(\Omega), \quad (2.1)$$

Throughout the paper, we denote by $L_2(\Omega)$, $L_{\infty}(\Omega)$, and $H^s(\Omega)$, $s \in \mathbb{N}$, the usual Lebesgue and Sobolev spaces, respectively. Moreover, $H_0^1(\Omega) \subset H^1(\Omega)$ is the subspace of functions whose trace vanishes on $\partial\Omega$. We use the standard scalar products $(\cdot, \cdot)_{L_2(\Omega)}$ and $(\cdot, \cdot)_{H^1(\Omega)} := (\nabla \cdot, \nabla \cdot)_{L_2(\Omega)}$, norms $\|\cdot\|_{L_2(\Omega)}$, $\|\cdot\|_{L_{\infty}(\Omega)}$, $\|\cdot\|_{H^1(\Omega)}$, and seminorm $|\cdot|_{H^1(\Omega)}$.

2.2 Representation of the Computational Domain

The computational domain Ω is given as the composition of K non-overlapping subdomains $\Omega^{(1)}, \dots, \Omega^{(K)}$, i.e., each $\Omega^{(k)}$ is open and bounded with Lipschitz boundary,

$$\overline{\Omega} = \bigcup_{k=1}^K \overline{\Omega^{(k)}} \quad \text{and} \quad \Omega^{(k)} \cap \Omega^{(\ell)} = \emptyset \quad \text{for all } k \neq \ell, \quad (2.2)$$

where $\overline{T}$ denotes the closure of the set T . We refer to the subdomains $\Omega^{(k)}$ as patches. We assume that every patch $\Omega^{(k)}$ is the image of a geometry function

$$G_k: \widehat{\Omega} := (0, 1)^2 \rightarrow \Omega^{(k)} := G_k(\widehat{\Omega}) \subset \mathbb{R}^2,$$

that can be continuously extended to the closure of $\widehat{\Omega}$.

Although in IgA commonly B-spline or NURBS functions are used for the representation of the geometry, this is not required for the proposed method. We only assume that the geometry function is sufficiently smooth such that the following assumption holds.

Assumption 1. There is a constant $C_1 > 0$ such that

$$\|\nabla G_k\|_{L_\infty(T)} \leq C_1 H_k \quad \text{and} \quad \|(\nabla G_k)^{-1}\|_{L_\infty(T)} \leq C_1 \frac{1}{H_k}$$

for $T \in \{\widehat{\Omega}, \partial\widehat{\Omega}\}$ and all $k = 1, \dots, K$, where $H_k > 0$ is the diameter of the patch $\Omega^{(k)}$.

This assumption imposes quality criteria on the representation of the geometry. It might be possible to reduce the constant C_1 by a reparameterization of the geometry, i.e., by finding another function $\widetilde{G}_k$ such that $\widetilde{G}_k(\widehat{\Omega}) = G_k(\widehat{\Omega})$. Moreover, also subdividing a patch into one or more patches might help in reducing the constant C_1 .

Note that equation (2.2) guarantees that every interface between two patches is exactly parameterized by the geometry functions of both involved patches. This, however, does not imply that the parameterizations have to agree. For the setup of the solver, the interfaces between two patches have to consist of whole edges, which means that there are no T-junctions. The next assumption guarantees this.

Assumption 2. For any two patch indices $k \neq \ell$, the intersection $\overline{\Omega^{(k)}} \cap \overline{\Omega^{(\ell)}}$ is either a common edge (with the neighboring vertices), a common vertex, or empty.

This assumption also ensures that the pre-images $\widehat{\Gamma}_D^{(k)} := G_k^{-1}(\partial\Omega \cap \partial\Omega^{(k)})$ of the (Dirichlet) boundary $\Gamma_D = \partial\Omega$ consist of whole edges.

For any two neighboring patches $\Omega^{(k)}$ and $\Omega^{(\ell)}$, the common edge is denoted by $\Gamma^{(k,\ell)} = \Gamma^{(\ell,k)}$, and its pre-images by $\widehat{\Gamma}^{(k,\ell)} := G_k^{-1}(\Gamma^{(k,\ell)})$ and $\widehat{\Gamma}^{(\ell,k)} := G_\ell^{-1}(\Gamma^{(\ell,k)})$. We denote the two end-points of $\Gamma^{(\ell,k)}$ by $\mathbf{x}^{(k,\ell,1)}$ and $\mathbf{x}^{(k,\ell,2)}$. We collect the indices of patches sharing an edge in a set

$$\mathcal{N}_\Gamma(k) := \{\ell \neq k : \Omega^{(k)} \text{ and } \Omega^{(\ell)} \text{ share one edge}\}.$$

The indices of the patches that contain a certain corner are collected in the set

$$\mathcal{P}(\mathbf{x}) := \{k : \mathbf{x} \in \overline{\Omega^{(k)}}\}.$$

Moreover, we require that the number of neighbors of a patch is uniformly bounded.

Assumption 3. There is a constant C_2 such that $|\mathcal{P}(\mathbf{x})| \leq C_2$ for all corners $\mathbf{x}$.

2.3 Local Function Spaces

After the introduction of the computational domain, we establish the isogeometric function spaces. In IgA, those function spaces are either B-spline or NURBS functions. In this paper, we focus on B-splines. Let $p \in \mathbb{N} := \{1, 2, 3, \dots\}$ be a given spline degree. For ease of notation, we assume that it is the same for all patches. For $n \in \mathbb{N}$, a p -open knot vector $\Xi = (\xi_1, \dots, \xi_{n+p+1})$ is the building block for the B-spline basis $(B[p, \Xi, i])_{i=1}^n$ defined via the Cox-de Boor formula, cf. [9, equations (2.1) and (2.2)]. This basis spans the univariate spline space

$$S[p, \Xi] := \text{span}\{B[p, \Xi, 1], \dots, B[p, \Xi, n]\}.$$

Let $\Xi^{(k,1)}$ and $\Xi^{(k,2)}$ be two p -open knot vectors over $(0, 1)$. To get a multivariate spline space $\widehat{V}^{(k)}$ over $\widehat{\Omega}$, we tensorize the two univariate spline spaces. The transformation of $\widehat{V}^{(k)}$ to the physical domain is defined by the pull-back principle. We denote the resulting space by $V^{(k)}$. So we define

$$\widehat{V}^{(k)} := \{v \in S[p, \Xi^{(k,1)}] \otimes S[p, \Xi^{(k,2)}] : v|_{\widehat{\Gamma}_D^{(k)}} = 0\} \quad \text{and} \quad V^{(k)} := \widehat{V}^{(k)} \circ G_k^{-1},$$

where $v|_T$ denotes the restriction of v to T (trace operator). The basis for the space $\widehat{V}^{(k)}$ consists of only those tensor-product basis functions over $\Omega^{(k)}$ that vanish on the pre-image of the Dirichlet boundary $\widehat{\Gamma}_D^{(k)}$. Say, the total number of basis functions of $\widehat{V}^{(k)}$ is $N^{(k)} = N_I^{(k)} + N_\Gamma^{(k)}$. The number $N_I^{(k)}$ denotes the number of basis

function that are supported only in the interior of the patch, whereas $N_\Gamma^{(k)}$ accounts for the number of basis functions that contribute to the boundary of the patch. These definitions give rise to an ordered basis

$$\begin{aligned}\widehat{\Phi}^{(k)} &:= (\widehat{\phi}_i^{(k)})_{i=1}^{N_\Gamma^{(k)}}, \\ \{\widehat{\phi}_i^{(k)}\} &= \{\widehat{\phi} \in H^1(\widehat{\Omega}) : \widehat{\phi}(x, y) = B[p, \Xi^{(k,1)}, j_1](x)B[p, \Xi^{(k,2)}, j_2](y), \\ &\quad \text{where } j_1 \in \{1, \dots, n^{(k,1)}\} \text{ and } j_2 \in \{1, \dots, n^{(k,2)}\} \text{ such that } \widehat{\phi}|_{\Gamma_D^{(k)}} = 0\}, \\ \widehat{\phi}_i^{(k)}|_{\partial\widehat{\Omega}} &= 0 \iff i \in \{1, \dots, N_\Gamma^{(k)}\} \quad \text{and} \quad \widehat{\phi}_i^{(k)}|_{\partial\widehat{\Omega}} \neq 0 \iff i \in N_\Gamma^{(k)} + \{1, \dots, N_\Gamma^{(k)}\}.\end{aligned}\tag{2.3}$$

The pull-back principle gives a basis for $V^{(k)}$,

$$\Phi^{(k)} := (\phi_i^{(k)})_{i=1}^{N_\Gamma^{(k)}} \quad \text{and} \quad \phi_i^{(k)} := \widehat{\phi}_i^{(k)} \circ G_k^{-1}.$$

We assume in the following that the grids on each of the patches are quasi-uniform.

Assumption 4. There are grid sizes $\widehat{h}_k > 0$ for $k = 1, \dots, K$ and a constant $C_3 > 0$ such that

$$C_3 \widehat{h}_k \leq \xi_{i+1}^{(k,\delta)} - \xi_i^{(k,\delta)} \leq \widehat{h}_k$$

holds for all $i = 1, \dots, n^{(k,\delta)} + p$ and all $\delta = 1, 2$ red with $\xi_{i+1}^{(k,\delta)} \neq \xi_i^{(k,\delta)}$.

The corresponding grid size on the physical domain is defined via $h_k := \widehat{h}_k H_k$. For any two patches $\Omega^{(k)}$ and $\Omega^{(\ell)}$ sharing an edge, we define

$$\widehat{h}_{k\ell} := \min\{\widehat{h}_k, \widehat{h}_\ell\} \quad \text{and} \quad h_{k\ell} := \min\{h_k, h_\ell\}.$$

2.4 The Isogeometric SIPG Problem

Since we do not impose any strong coupling of the solution at this point, the global approximation space is the product space of the local spaces $V^{(k)}$, i.e.,

$$V := \prod_{k=1}^K V^{(k)} := V^{(1)} \times \dots \times V^{(K)}.$$

Using these function spaces, the dG discretization of the model problem (2.1) is given by the following: find $u = (u^{(1)}, \dots, u^{(K)}) \in V$ such that

$$a_h(u, v) = \int_{\Omega} f v \, dx \quad \text{for all } v \in V,\tag{2.4}$$

where

$$\begin{aligned}a_h(u, v) &:= \sum_{k=1}^K (a^{(k)}(u, v) + m^{(k)}(u, v) + r^{(k)}(u, v)), \\ a^{(k)}(u, v) &:= \int_{\Omega^{(k)}} \nabla u^{(k)} \cdot \nabla v^{(k)} \, dx, \\ m^{(k)}(u, v) &:= \sum_{\ell \in \mathcal{N}_\Gamma(k)} \int_{\Gamma^{(k,\ell)}} \frac{1}{2} \left(\frac{\partial u^{(k)}}{\partial n_k} (v^{(\ell)} - v^{(k)}) + \frac{\partial v^{(k)}}{\partial n_k} (u^{(\ell)} - u^{(k)}) \right) ds, \\ r^{(k)}(u, v) &:= \sum_{\ell \in \mathcal{N}_\Gamma(k)} \int_{\Gamma^{(k,\ell)}} \frac{\delta p^2}{h_{k\ell}} (u^{(\ell)} - u^{(k)}) (v^{(\ell)} - v^{(k)}) \, ds,\end{aligned}$$

and $\delta > 0$ is a penalty parameter and n_k is the outer normal vector of $\partial\Omega^{(k)}$.

Due to [38, Theorem 8], the parameter δ can always be chosen independently of the spline degree p and mesh sizes h_k such that the bilinear form $a_h(\cdot, \cdot)$ is bounded and coercive in the dG-norm

$$\|v\|_d^2 := d(v, v), \quad \text{where } d(u, v) := \sum_{k=1}^K (a^{(k)}(u, v) + r^{(k)}(u, v)).$$

Note that δ depends on the constant C_1 from Assumption 1. Similar results have been shown previously, but up to the knowledge of the authors, the dependence on p has only been addressed in [38]. Since we have coercivity and boundedness, the Lax–Milgram lemma ensures the existence of a solution of (2.4) and its uniqueness. If the solution u to the continuous problem (2.1) satisfies $u|_{\Omega^{(k)}} \in H^{q+1}(\Omega^{(k)})$ for all $k = 1, \dots, K$ with $1 \leq q \leq p$, then the discretization error satisfies

$$\|u - u_h\|_d^2 \leq Ch^{2q} \sum_{k=1}^K \|u\|_{H^{q+1}(\Omega^{(k)})}^2 \quad (2.5)$$

for some constant $C > 0$ independent of h , see [38, Theorems 11 and 12], where more error estimates, including results on their dependence on the spline degree, are available.

3 The dG IETI-DP Solver

In this section, we propose an IETI-DP solver for (2.4). As for any tearing and interconnecting method, we have to choose local spaces, cf., e.g., [14]. As it has been done previously for IETI methods, the local spaces are chosen on a per-patch basis, cf. [22] and later publications. In the case of dG, the choice is not completely straight-forward. We follow the approach that has been proposed in [12] and that has been used in an IgA context in [16, 17] and others: for each patch, we introduce a local subspace that consists of the degrees of freedom associated to that patch and of the degrees of freedom associated to artificial interfaces. The basis functions on the artificial interface coincide with the traces of basis functions of the neighboring patch, when restricted to the interface. We introduce these local function spaces and the corresponding bilinear form in Section 3.1. In Section 3.2, we restrict the problem to the skeleton. As a next step, we introduce jump matrices that are used for the weak coupling of the degrees of freedom on an interface and on the artificial interface of the neighboring subdomain, see Section 3.3. Following the concept of dual-primal methods, in Section 3.4, we introduce the primal degrees of freedom, where we consider vertex values (Algorithm A), edge averages (Algorithm B), and a combination of both (Algorithm C) as options. Having the required ingredients, we pose the IETI-DP system and the proposed preconditioner in Section 3.5. In Section 3.6, we summarize all steps that are required for a computational realization of the solver.

3.1 Local Function Spaces and Bilinear Forms

The patch-local subspace consists of the local functions on the particular patch and of those functions from the neighboring patches required to realize the bilinear forms $m^{(k)}(\cdot, \cdot)$ and $r^{(k)}(\cdot, \cdot)$. We define the enriched function space

$$V_e^{(k)} := V^{(k)} \times \prod_{\ell \in \mathcal{N}_\Gamma(k)} V^{(k, \ell)},$$

where $V^{(k, \ell)} := \{v^{(\ell)}|_{\Gamma^{(k, \ell)}} : v^{(\ell)} \in V^{(\ell)}\}$ is the trace of $V^{(\ell)}$. We write

$$v_e^{(k)} = (v^{(k)}, (v^{(k, \ell)})_{\ell \in \mathcal{N}_\Gamma(k)}), \quad (3.1)$$

where $v_e^{(k)} \in V_e^{(k)}$, $v^{(k)} \in V^{(k)}$ and $v^{(k, \ell)} \in V^{(k, \ell)}$.

Note that

$$\{\phi_1^{(k, \ell)}, \dots, \phi_{N^{(k, \ell)}}^{(k, \ell)}\} := \{\phi_i^{(\ell)}|_{\Gamma^{(k, \ell)}} : \phi_i^{(\ell)} \in \Phi^{(\ell)}, \phi_i^{(\ell)}|_{\Gamma^{(k, \ell)}} \neq 0\}$$

forms a basis of $V^{(k, \ell)}$, which is uniquely defined up to the ordering of the basis functions. We denote the basis by $\Phi^{(k, \ell)} := (\phi_i^{(k, \ell)})_{i=1}^{N^{(k, \ell)}}$.

A visualization is given in Figure 1, where all basis functions are represented using a symbol located at their maximum. The basis functions for each of the patches are represented via different symbols. The symbols that are located on the edges and corners represent the only functions that are supported on that

Figure 1: Local spaces with artificial interfaces

Figure 2: Basis functions in the interior (yellow), on an interface (red), and on an artificial interface (blue)

edge or corner, respectively. Besides the patches themselves, we have artificial interfaces. While they are co-located with the (standard) interfaces, they are treated as separate entities on which the corresponding trace spaces $V^{(k,\ell)}$ live. The basis functions living on these trace spaces are marked with the same kind of symbol as the basis function from the original space. Examples of basis functions in the interior, on the interfaces, and on the artificial interfaces are visualized in Figure 2.

Following the structure of (3.1), we define a basis

$$\Phi_e^{(k)} := (\phi_{e,i}^{(k)})_{i=1}^{N_e^{(k)}} \quad \text{with } N_e^{(k)} := \dim V_e^{(k)}$$

for the space $V_e^{(k)}$ as follows. We start with the basis functions from $\Phi^{(k)}$,

$$\phi_{e,i}^{(k)} := (\phi_i^{(k)}, (0, \dots, 0)) \quad \text{for } i = 1, \dots, N^{(k)}.$$

Then the basis functions follow from the neighboring patches. Let $\mathcal{N}_\Gamma(k) = \{\ell_1, \ell_2, \dots, \ell_L\}$. We define

$$\begin{aligned} \phi_{e,Y_i^{(k,\ell_1)}}^{(k)} &:= (0, (\phi_i^{(k,\ell_1)}, 0, \dots, 0)) \\ &\quad \text{with } Y_i^{(k,\ell_1)} := N^{(k)} + i \quad \text{for } i = 1, \dots, N^{(k,\ell_1)}, \\ \phi_{e,Y_i^{(k,\ell_2)}}^{(k)} &:= (0, (0, \phi_i^{(k,\ell_2)}, 0, \dots, 0)) \\ &\quad \text{with } Y_i^{(k,\ell_2)} := N^{(k)} + N^{(k,\ell_1)} + i \quad \text{for } i = 1, \dots, N^{(k,\ell_2)}, \\ &\vdots \\ \phi_{e,Y_i^{(k,\ell_L)}}^{(k)} &:= (0, (0, \dots, 0, \phi_i^{(k,\ell_L)})) \\ &\quad \text{with } Y_i^{(k,\ell_L)} := N^{(k)} + N^{(k,\ell_1)} + \dots + N^{(k,\ell_{L-1})} + i \quad \text{for } i = 1, \dots, N^{(k,\ell_L)}. \end{aligned} \quad (3.2)$$

From this construction and from (2.3), we know that the first $N_1^{(k)}$ basis functions of the basis $\Phi_e^{(k)}$ live in the interior of the patch $\Omega^{(k)}$. The following $N_\Gamma^{(k)}$ basis functions live on the interfaces. The remaining basis functions live on the artificial interfaces.

On the spaces $V_e^{(k)}$, we define the bilinear forms $a_e^{(k)}(\cdot, \cdot)$ and $d_e^{(k)}(\cdot, \cdot)$ and the linear functional $\langle f_e^{(k)}, \cdot \rangle$ analogous to the global objects $a_h(\cdot, \cdot)$ and $d(\cdot, \cdot)$ by

$$\begin{aligned} a_e^{(k)}(u_e^{(k)}, v_e^{(k)}) &:= a^{(k)}(u_e^{(k)}, v_e^{(k)}) + m^{(k)}(u_e^{(k)}, v_e^{(k)}) + r^{(k)}(u_e^{(k)}, v_e^{(k)}), \\ d_e^{(k)}(u_e^{(k)}, v_e^{(k)}) &:= a^{(k)}(u_e^{(k)}, v_e^{(k)}) + r^{(k)}(u_e^{(k)}, v_e^{(k)}), \\ \langle f_e^{(k)}, v_e^{(k)} \rangle &:= \int_{\Omega^{(k)}} f v^{(k)} \, dx, \end{aligned}$$

where we write with a slight abuse of notation

$$\begin{aligned} a^{(k)}(u_e^{(k)}, v_e^{(k)}) &:= \int_{\Omega^{(k)}} \nabla u^{(k)} \cdot \nabla v^{(k)} \, dx, \\ m^{(k)}(u_e^{(k)}, v_e^{(k)}) &:= \sum_{\ell \in \mathcal{N}_\Gamma(k)} \int_{\Gamma^{(k,\ell)}} \frac{1}{2} \left(\frac{\partial u^{(k)}}{\partial n_k} (v^{(k,\ell)} - v^{(k)}) + \frac{\partial v^{(k)}}{\partial n_k} (u^{(k,\ell)} - u^{(k)}) \right) ds, \\ r^{(k)}(u_e^{(k)}, v_e^{(k)}) &:= \sum_{\ell \in \mathcal{N}_\Gamma(k)} \int_{\Gamma^{(k,\ell)}} \frac{\delta p^2}{h_{k\ell}} (u^{(k,\ell)} - u^{(k)}) (v^{(k,\ell)} - v^{(k)}) \, ds. \end{aligned}$$

The bilinear form $d_e^{(k)}(\cdot, \cdot)$ induces the local dG-norm $\|\cdot\|_{d_e^{(k)}}^2 := d_e^{(k)}(\cdot, \cdot)$, the bilinear form $a_e^{(k)}(\cdot, \cdot)$ the local energy norm $\|\cdot\|_{a_e^{(k)}}^2 := a_e^{(k)}(\cdot, \cdot)$.

By discretizing, we obtain the local linear system

$$A^{(k)} \underline{u}_e^{(k)} = \underline{f}_e^{(k)}, \quad (3.3)$$

where

$$A^{(k)} = [a_e^{(k)}(\phi_{e,j}^{(k)}, \phi_{e,i}^{(k)})]_{i,j=1}^{N_e^{(k)}} \quad \text{and} \quad \underline{f}_e^{(k)} = [f_e^{(k)}(\phi_{e,i}^{(k)})]_{i=1}^{N_e^{(k)}}.$$

The vector $\underline{u}_e^{(k)} = [u_{e,i}^{(k)}]_{i=1}^{N_e^{(k)}}$ is the coefficient vector representing the function $u_e^{(k)} = \sum_{i=1}^{N_e^{(k)}} u_{e,i}^{(k)} \phi_{e,i}^{(k)}$.

We subdivide the stiffness matrix and the load vector into blocks

$$A^{(k)} = \begin{pmatrix} A_{\Pi}^{(k)} & A_{\Gamma}^{(k)} \\ A_{\Gamma}^{(k)} & A_{\Gamma\Gamma}^{(k)} \end{pmatrix} \quad \text{and} \quad \underline{f}_e^{(k)} = \begin{pmatrix} \underline{f}_{\Pi}^{(k)} \\ \underline{f}_{\Gamma}^{(k)} \end{pmatrix},$$

where the first row and the first column correspond to the first $N_I^{(k)}$ basis functions, i.e., to those supported in the interior of the patch $\Omega^{(k)}$. So the remainder accounts for both the standard interfaces and the artificial interfaces.

3.2 The Problem on the Skeleton

In this section, we eliminate the interior degrees of freedom by building a Schur complement system, which is equivalent to (3.3),

$$S^{(k)} \underline{w}_e^{(k)} = \underline{g}_e^{(k)}, \quad (3.4)$$

where

$$S^{(k)} := A_{\Gamma\Gamma}^{(k)} - A_{\Gamma\Gamma}^{(k)} (A_{\Pi}^{(k)})^{-1} A_{\Pi\Gamma}^{(k)} \quad \text{and} \quad \underline{g}_e^{(k)} := \underline{f}_{\Gamma}^{(k)} - A_{\Gamma\Gamma}^{(k)} (A_{\Pi}^{(k)})^{-1} \underline{f}_{\Pi}^{(k)}. \quad (3.5)$$

Once $\underline{w}_e^{(k)}$ has been computed, we get the solution $\underline{u}_e^{(k)}$ of system (3.3) by

$$\underline{u}_e^{(k)} = \begin{pmatrix} \underline{u}_{\Pi}^{(k)} \\ \underline{u}_{\Gamma}^{(k)} \end{pmatrix} = \begin{pmatrix} (A_{\Pi}^{(k)})^{-1} (\underline{f}_{\Pi}^{(k)} - A_{\Pi\Gamma}^{(k)} \underline{w}_e^{(k)}) \\ \underline{w}_e^{(k)} \end{pmatrix}. \quad (3.6)$$

The collection of the systems (3.4) for all patches yields a block diagonal linear system

$$S \underline{w} = \underline{g}. \quad (3.7)$$

Next, we introduce the building blocks in order to rewrite the Schur complement system in a variational setting. First, we define function spaces on the skeleton,

$$W := \prod_{k=1}^K W_e^{(k)}, \quad \text{where } W_e^{(k)} := W^{(k)} \times \prod_{\ell \in \mathcal{N}_\Gamma(k)} W^{(k,\ell)},$$

$W^{(k)} := \{v|_{\partial\Omega^{(k)}} : v \in V^{(k)}\}$ and $W^{(k,\ell)} := V^{(k,\ell)}$. Analogously to (3.1), an extended function $w_e^{(k)} \in W_e^{(k)}$ has the form

$$w_e^{(k)} = (w^{(k)}, (w^{(k,\ell)})_{\ell \in \mathcal{N}_\Gamma(k)}),$$

where $w^{(k)} \in W^{(k)}$ and $w^{(k,\ell)} \in W^{(k,\ell)}$. A basis for $W_e^{(k)}$ is given by

$$\check{\Phi}_e^{(k)} := (\phi_{e,N_I^{(k)}+i}^{(k)})_{i=1}^{\check{N}_e^{(k)}} \quad \text{with } \check{N}_e^{(k)} := N_e^{(k)} - N_I^{(k)}.$$

As above, the coefficient vector $\underline{w}_e^{(k)} = [w_{e,i}^{(k)}]_{i=1}^{\check{N}_e^{(k)}}$ represents the function

$$w_e^{(k)} = \sum_{i=1}^{\check{N}_e^{(k)}} w_{e,i}^{(k)} \phi_{e,N_I^{(k)}+i}^{(k)}. \quad (3.8)$$

The Schur complements realize the discrete harmonic extension $\mathcal{H}_A^{(k)}: W_e^{(k)} \rightarrow V_e^{(k)}$ with respect to the energy norm $\|\cdot\|_{a_e^{(k)}}$, which is defined as follows. For any $w_e^{(k)} = (w^{(k)}, (w^{(k,\ell)})_{\ell \in \mathcal{N}_\Gamma(k)})$, we have

$$\mathcal{H}_A^{(k)} w_e^{(k)} = (u^{(k)}, (w^{(k,\ell)})_{\ell \in \mathcal{N}_\Gamma(k)}),$$

where $u^{(k)}$ is such that

$$\begin{aligned} u^{(k)} &= w^{(k)} \quad \text{on } \partial\Omega^{(k)}, \\ a_e^{(k)}((u^{(k)}, (w^{(k,\ell)})_{\ell \in \mathcal{N}_\Gamma(k)}), (v^{(k)}, 0)) &= 0 \quad \text{for all } v^{(k)} \in V_0^{(k)}, \end{aligned}$$

where $V_0^{(k)} := V^{(k)} \cap H_0^1(\Omega^{(k)})$. Note that the standard discrete harmonic extension $\mathcal{H}_h^{(k)}: W^{(k)} \rightarrow V^{(k)}$ is defined by $\mathcal{H}_h^{(k)} w^{(k)} = u^{(k)}$, where $u^{(k)}$ is such that

$$\begin{aligned} u^{(k)} &= w^{(k)} \quad \text{on } \partial\Omega^{(k)}, \\ a^{(k)}(u^{(k)}, v^{(k)}) &= 0 \quad \text{for all } v^{(k)} \in V_0^{(k)}. \end{aligned}$$

In variational form, problem (3.7) reads as follows: find $w = (w_e^{(1)}, \dots, w_e^{(K)}) \in W$ such that

$$\underbrace{\sum_{k=1}^K a_e^{(k)}(\mathcal{H}_A^{(k)} w_e^{(k)}, \mathcal{H}_A^{(k)} q_e^{(k)})}_{=: s(w, q)} = \underbrace{\sum_{k=1}^K \langle f, \mathcal{H}_A^{(k)} q_e^{(k)} \rangle}_{=: \langle g, q \rangle} \quad \text{for all } q = (q_e^{(1)}, \dots, q_e^{(K)}) \in W.$$

3.3 The Jump Matrices

The next step is the introduction of constraints that yield the continuity of the solution. For any two patches $\Omega^{(k)}$ and $\Omega^{(\ell)}$ sharing an edge $\Gamma^{(k,\ell)}$ and any basis function $\phi_i^{(k)}$ supported on that edge, we introduce a constraint

$$w_{e,i^*}^{(k)} - w_{e,j^*}^{(\ell)} = 0 \quad \text{with } i^* = i - N_I^{(k)} \text{ and } j^* = \Upsilon_i^{(\ell,k)} - N_I^{(\ell)}, \quad (3.9)$$

where $\Upsilon_i^{(\ell,k)}$ is as in (3.2) and $w_{e,i^*}^{(k)}$ and $w_{e,j^*}^{(\ell)}$ are the coefficients of the functions $w_e^{(k)}$ and $w_e^{(\ell)}$. The index shifts by $N_I^{(k)}$ and $N_I^{(\ell)}$ account for the restriction to $W_e^{(k)}$ and $W_e^{(\ell)}$, cf. (3.8). This constraint ensures that the function values of the solution on the artificial interfaces coincide with the function values of the solution on the corresponding standard interfaces.

If the corner values are chosen as primal degrees of freedom (Algorithms A and C), we omit the constraints for the basis functions $\phi_i^{(k)}$ that are supported on a corner of $\Omega^{(k)}$. A visualization of the constraints is given in Figure 3.

If only the edge averages are chosen as primal degrees of freedom (Algorithm B), we realize the constraints at the corners in a fully redundant way. This means that, besides the constraints (3.9), for every basis function $\phi_i^{(k)}$ that is supported on a corner of $\Omega^{(k)}$, we additionally introduce constraints of the form

$$w_{e,j_1^*}^{(\ell_1)} - w_{e,j_2^*}^{(\ell_2)} = 0 \quad \text{with } j_1^* = \Upsilon_i^{(\ell_1,k)} - N_I^{(\ell_1)} \text{ and } j_2^* = \Upsilon_i^{(\ell_2,k)} - N_I^{(\ell_2)}, \quad (3.10)$$

where $\ell_1 < \ell_2$ such that the said corner lies between the edges $\Gamma^{(k,\ell_1)}$ and $\Gamma^{(k,\ell_2)}$. These additional constraints also ensure the continuity between the different artificial interfaces. A visualization of this case is given in Figure 4.

Figure 3: Omitting vertices (Algorithms A and C) **Figure 4:** Fully redundant (Algorithm B)

We collect the constraints of the form (3.9) and (3.10) in a jump matrix B such that the constraints are equivalent to $B\bar{w} = 0$. This is done such that each row of the matrix corresponds to one constraint, i.e., each row has only two non-zero entries, one with value 1 and one with value -1 . The matrix B can also be written as $B = (B^{(1)} \dots B^{(K)})$.

We are now able to write down a saddle point problem in matrix-vector form that is equivalent to problem (2.4): find $(\bar{w}, \bar{\lambda})$ such that

$$\begin{pmatrix} S & B^T \\ B & \end{pmatrix} \begin{pmatrix} \bar{w} \\ \bar{\lambda} \end{pmatrix} = \begin{pmatrix} \bar{g} \\ 0 \end{pmatrix}.$$

Note that the matrices $A^{(k)}$ and $S^{(k)}$ are non-singular only if the corresponding patch contributes to the Dirichlet boundary $\partial\Omega$. Since in general at least one of the matrices $S^{(k)}$ is singular, the matrix S is usually singular. The remedy for this problem are primal degrees of freedom which form a global but small linear system.

3.4 The Primal Degrees of Freedom

As mentioned in the introduction, we consider three options for the choice of primal degrees of freedom: Algorithms A, B, and C. Each of them corresponds to a choice of spaces $\bar{W}$ and $\bar{W}_\Delta$.

Algorithm A (Vertex Values). The space $\bar{W}$ is the subspace of functions where the vertex values agree, i.e.,

$$\bar{W} := \{w \in W : w^{(k)}(\mathbf{x}^{(k,\ell,i)}) = w^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)}) \text{ for all } k, \text{ all } \ell \in \mathcal{N}_\Gamma(k), \text{ and all } i\},$$

where $\mathbf{x}^{(k,\ell,1)}$ and $\mathbf{x}^{(k,\ell,2)}$ are the two end-points of $\Gamma^{(k,\ell)}$. Furthermore, $\bar{W}_\Delta$ satisfies these conditions homogeneously, i.e., $\bar{W}_\Delta := \prod_{k=1}^K \bar{W}_\Delta^{(k)}$ and

$$\bar{W}_\Delta^{(k)} := \{w_e^{(k)} \in W_e^{(k)} : w^{(k)}(\mathbf{x}^{(k,\ell,i)}) = w^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)}) = 0 \text{ for all } \ell \in \mathcal{N}_\Gamma(k) \text{ and all } i\}.$$

Algorithm B (Edge Averages). The space $\bar{W}$ is the subspace of functions where the averages of the function values over the edges agree, i.e.,

$$\bar{W} := \left\{ w \in W : \int_{\Gamma^{(k,\ell)}} w^{(k)} dx = \int_{\Gamma^{(k,\ell)}} w^{(\ell,k)} dx \text{ for all } k \text{ and all } \ell \in \mathcal{N}_\Gamma(k) \right\}.$$

Furthermore, $\bar{W}_\Delta$ satisfies these conditions homogeneously, i.e., $\bar{W}_\Delta := \prod_{k=1}^K \bar{W}_\Delta^{(k)}$ and

$$\bar{W}_\Delta^{(k)} := \left\{ w_e^{(k)} \in W_e^{(k)} : \int_{\Gamma^{(k,\ell)}} w^{(k)} dx = \int_{\Gamma^{(k,\ell)}} w^{(\ell,k)} dx = 0 \text{ for all } \ell \in \mathcal{N}_\Gamma(k) \right\}.$$

Algorithm C (Vertex Values and Edge Averages). The spaces $\bar{W}$ and $\bar{W}_\Delta^{(k)}$ and $\bar{W}_\Delta$ are the intersections of the corresponding spaces obtained by Algorithms A and B.

Figure 5: Constraints for Algorithm A

Figure 6: Constraints for Algorithm B

Figures 5 and 6 visualize the constraints for Algorithms A and B, respectively. The arrows in Figure 5 indicate which corner degrees of freedom are coupled in Algorithm A. The different lines in Figure 6 indicate the coupling of the edge averages. Interfaces drawn in the same style have to have the same average.

The primal space $\widetilde{W}_\Pi$ is the subspace of energy minimizing functions, cf. [30]. This means that $\widetilde{W}_\Pi$ is S -orthogonal to $\widetilde{W}_\Delta$, i.e.,

$$\widetilde{W}_\Pi := \{w \in \widetilde{W} : s(w, q) = 0 \text{ for all } q \in \widetilde{W}_\Delta\}. \quad (3.11)$$

Let $\psi^{(1)}, \dots, \psi^{(N_\Pi)}$ be a basis of $\widetilde{W}_\Pi$. In practice, one usually chooses a nodal basis, where the vertex values and/or the edge averages form the nodal values. The matrix Ψ represents the basis in terms of the basis for the space W , i.e.,

$$\Psi^\top = ((\psi^{(1)})^\top \dots (\psi^{(N_\Pi)})^\top)^\top. \quad (3.12)$$

A representation for the spaces $\widetilde{W}_\Delta^{(k)}$ can be obtained by full-rank matrices $C^{(k)}$, $C^{(k)} \underline{w}_e^{(k)} = 0 \Leftrightarrow w_e^{(k)} \in \widetilde{W}_\Delta^{(k)}$. The block-diagonal collection of the matrices $C^{(k)}$ forms a matrix C .

3.5 The IETI-DP System and the Scaled Dirichlet Preconditioner

As in [26], the following problem is an equivalent reformulation of (2.4): find $(\underline{w}_\Delta, \underline{\mu}, \underline{w}_\Pi, \underline{\lambda})$ such that

$$\begin{pmatrix} S & C^\top & 0 & B^\top \\ C & 0 & 0 & 0 \\ 0 & 0 & \Psi^\top S \Psi & \Psi^\top B^\top \\ B & 0 & B \Psi & 0 \end{pmatrix} \begin{pmatrix} \underline{w}_\Delta \\ \underline{\mu} \\ \underline{w}_\Pi \\ \underline{\lambda} \end{pmatrix} = \begin{pmatrix} \underline{g} \\ 0 \\ \Psi^\top \underline{g} \\ 0 \end{pmatrix}.$$

The solution for the original problem is obtained by $\underline{w} = \underline{w}_\Delta + \Psi \underline{w}_\Pi$. Following the DP approach, this system can be reduced to

$$F \underline{\lambda} = \underline{d}, \quad (3.13)$$

where

$$F := \underbrace{(B \ 0 \ B \Psi) \begin{pmatrix} S & C^\top & 0 \\ C & 0 & 0 \\ 0 & 0 & \Psi^\top S \Psi \end{pmatrix}^{-1} \begin{pmatrix} I \\ 0 \\ \Psi^\top \end{pmatrix} B^\top}_{=: F_0} \quad \text{and} \quad \underline{d} := F_0 \underline{g}.$$

System (3.13) is solved with a preconditioned conjugate gradient (PCG) solver. In order to obtain an effective solver, we need a proper preconditioner. We use the scaled Dirichlet preconditioner M_{SD} , defined by

$$M_{\text{SD}} := B D^{-1} S D^{-1} B^\top,$$

where $D \in \mathbb{R}^{N_\Gamma \times N_\Gamma}$ is a diagonal matrix defined based on the principle of multiplicity scaling: each coefficient $d_{i,i}$ of D is assigned the number of constraints for the corresponding degree of freedom plus one.

3.6 The IETI-DP Algorithm

The dG IETI-DP method requires the execution of the following steps.

- Compute the right-hand sides $\underline{g}_e^{(k)}$ according to (3.5).
- Compute a basis Ψ for the primal space in accordance with (3.11) and (3.12).
- Compute the primal Schur complement $S_\Pi := \Psi^\top S \Psi$.
- Solve (3.13) for $\underline{\lambda}$ using a PCG solver. This requires the computation of the residual and the application of the preconditioner.

The computation of the residual $\widehat{w} := F\underline{\lambda} - \underline{d}$ requires the following steps:

- (1) Compute $\widehat{q} = ((\widehat{q}_e^{(1)})^\top \dots (\widehat{q}_e^{(K)})^\top)^\top := B^\top \underline{\lambda} - \underline{g}$.
- (2) Solve the linear system

$$\begin{pmatrix} S^{(k)} & (C^{(k)})^\top \\ C^{(k)} & \end{pmatrix} \begin{pmatrix} \widehat{w}_\Delta^{(k)} \\ \widehat{p}^{(k)} \end{pmatrix} = \begin{pmatrix} \widehat{q}_e^{(k)} \\ 0 \end{pmatrix}$$

for all $k = 1, \dots, K$. For Algorithms A and C, one usually reduces the local systems by eliminating the degrees of freedom corresponding to the vertex values and the corresponding Lagrange multipliers.

- (3) Solve the (usually small) global linear system $S_\Pi \widehat{w}_\Pi = \Psi^\top \widehat{q}$.
- (4) Compute the residual $\widehat{w} := B((\widehat{w}_\Delta^{(1)})^\top \dots (\widehat{w}_\Delta^{(K)})^\top)^\top + B\Psi \widehat{w}_\Pi$.

Then apply the preconditioner, i.e., compute $M_{\text{SD}} \widehat{w}$.

- The vector $\underline{w}$, representing the solution on the skeleton, is computed analogously to steps (2), (3), and (4) based on $\underline{q} = ((q^{(1)})^\top \dots (q^{(K)})^\top)^\top := B^\top \underline{\lambda}$. Finally, the solution vector $\underline{u}$ is derived by (3.6).

4 The Condition Number Estimate

In this section, we prove the following condition number estimates.

Theorem 1. *Provided that the IETI-DP solver is set up as outlined in the previous sections, the condition number of the preconditioned system satisfies*

$$\kappa(M_{\text{SD}} F) \leq Cp \left(1 + \log p + \max_{k=1, \dots, K} \log \frac{H_k}{h_k} \right)^2$$

for Algorithms A and C and

$$\kappa(M_{\text{SD}} F) \leq C\delta p \left(\max_{k=1, \dots, K} \max_{\ell \in \mathcal{N}_\Gamma(k)} \frac{h_k}{h_\ell} \right) \left(1 + \log p + \max_{k=1, \dots, K} \log \frac{H_k}{h_k} \right)^2$$

for Algorithm B, where in both cases the constant C only depends on the constants from Assumptions 1, 3, and 4.

If some Lagrange multipliers are redundant to each other or to primal constraints, the matrices M_{SD} and F are singular. Standard iteration schemes live in the corresponding factor space. Here and in what follows, the condition number is estimated for the restriction to the factor space, cf. [26, Remark 23].

In Section 4.1, we introduce the notation and show some results concerning the well-posedness of the local systems. Then, in Section 4.2, we estimate the norm of $B_D^\top B \underline{u}$, where $B_D := BD^{-1}$. We give inequalities that allow to estimate terms living on the artificial interface in Section 4.3. Finally, in Section 4.4, we prove Theorem 1.

4.1 Preliminaries

First, we introduce some required notation. We write $a \lesssim b$ if there is a constant $c > 0$ that only depends on the constants from Assumptions 1, 3, and 4 such that $a \leq cb$. Moreover, we write $a \approx b$ if $a \lesssim b \lesssim a$.

The space $H^{1/2}$ is the standard Sobolev–Slobodeckij space with a norm defined as follows. For $T \subset \mathbb{R}^2$ open, bounded, and Lipschitz and $\Gamma \subseteq \partial T$, we define the seminorm $|\cdot|_{H^{1/2}(\Gamma)}$ by

$$|u|_{H^{1/2}(\Gamma)}^2 := \int_{\Gamma} \int_{\Gamma} \frac{|u(x) - u(y)|^2}{|x - y|^2} dx dy$$

and the corresponding norm by $\|\cdot\|_{H^{1/2}(\Gamma)}^2 := \|\cdot\|_{L_2(\Gamma)}^2 + |\cdot|_{H^{1/2}(\Gamma)}^2$. As in [32], we define the seminorm

$$|v|_{L_{\infty}^0(T)} := \inf_{c \in \mathbb{R}} \|v - c\|_{L_{\infty}(T)}.$$

Before we start with the condition number estimate, we show that it is sufficient to get the necessary estimates on the parameter domain.

Lemma 1. *For all patches $k = 1, \dots, K$, the equivalences*

- $|u|_{H^s(\Omega^{(k)})} \approx (H_k)^{1-s} |u \circ G_k|_{H^s(\widehat{\Omega})}$ for all $u \in H^s(\Omega^{(k)})$ and $s \in \{0, 1\}$, and
- $|w|_{H^s(\partial\Omega^{(k)})} \approx (H_k)^{1/2-s} |w \circ G_k|_{H^s(\partial\widehat{\Omega})}$ for all $w \in H^s(\partial\Omega^{(k)})$ and $s \in \{0, 1/2, 1\}$

hold, where we use $H^0(\cdot) := L_2(\cdot)$. The equivalences for the boundary do also hold if $\partial\Omega^{(k)}$ and $\partial\widehat{\Omega}$ and are replaced by $\Gamma^{(k,\ell)}$ and $\widehat{\Gamma}^{(k,\ell)}$, respectively.

Proof. The results for $s \in \{0, 1\}$ are obtained by Assumption 1, the chain rule and the substitution rule for differentiation, see e.g., [7, Lemma 3.5]. The results for $s = 1/2$ follow by Hilbert space interpolation theory, cf. [1, Theorem 7.23, Theorem 7.31]. $\square$

The next lemma states that the bilinear forms $a_e^{(k)}(\cdot, \cdot)$ and $d_e^{(k)}(\cdot, \cdot)$ are patchwise equivalent.

Lemma 2. *For every patch $\Omega^{(k)}$, the relation $a_e^{(k)}(u_e^{(k)}, u_e^{(k)}) \lesssim d_e^{(k)}(u_e^{(k)}, u_e^{(k)}) \lesssim a_e^{(k)}(u_e^{(k)}, u_e^{(k)})$ holds for all $u_e^{(k)} \in V_e^{(k)}$.*

Proof. Let $u_e^{(k)} = (u^{(k)}, (u^{(k,\ell)})_{\ell \in \mathcal{N}_{\Gamma}(k)})$ and $v_e^{(k)} = (v^{(k)}, (v^{(k,\ell)})_{\ell \in \mathcal{N}_{\Gamma}(k)}) \in V_e^{(k)}$. By Lemma 1, the fact that n_k has length 1, the discrete trace inequality [13, Lemma 4.3], Assumption 4, and again Lemma 1, we obtain

$$\begin{aligned} \|\nabla u^{(k)} \cdot n_k\|_{L_2(\Gamma^{(k,\ell)})}^2 &\lesssim \|\nabla u^{(k)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \|n_k\|_{L_{\infty}(\Gamma^{(k,\ell)})}^2 \lesssim \|\nabla(u^{(k)} \circ G_k)\|_{L_2(\widehat{\Gamma}^{(k,\ell)})}^2 \\ &\lesssim \frac{p^2}{h_{ke}} |u^{(k)} \circ G_k|_{H^1(\widehat{\Omega})}^2 \lesssim \frac{p^2}{h_{ke}} |u^{(k)}|_{H^1(\Omega^{(k)})}^2. \end{aligned} \quad (4.1)$$

Using the Cauchy–Schwarz inequality, estimate (4.1), and $\mathcal{N}_{\Gamma}(k) \leq 1$, we get

$$|m^{(k)}(u_e^{(k)}, v_e^{(k)})| \leq 2 \frac{c}{\delta^{1/2}} \|u_e^{(k)}\|_{d_e^{(k)}} \|v_e^{(k)}\|_{d_e^{(k)}} \quad (4.2)$$

for some constant $c > 0$, cf. [38] for details. With the choice $\delta \geq 16c^2$, we obtain

$$a_e^{(k)}(u_e^{(k)}, u_e^{(k)}) = \|u_e^{(k)}\|_{d_e^{(k)}}^2 + m^{(k)}(u_e^{(k)}, u_e^{(k)}) \geq \|u_e^{(k)}\|_{d_e^{(k)}}^2 - |m^{(k)}(u_e^{(k)}, u_e^{(k)})| \geq \frac{1}{2} \|u_e^{(k)}\|_{d_e^{(k)}}^2,$$

i.e., coercivity. Since we obtain boundedness using the Cauchy–Schwarz inequality and (4.2), this finishes the proof. $\square$

4.2 Estimates for the Action of $B_D^{\top} B$

Analogous to [32, Lemma 4.16], the following lemma estimates the effect of $B_D^{\top} B$.

Lemma 3. *Let*

$$u = (u_e^{(1)}, \dots, u_e^{(K)}) = ((u^{(1)}, (u^{(1,\ell)})_{\ell \in \mathcal{N}_{\Gamma}(1)}), \dots) \in \widetilde{W}$$

with coefficient vector $\underline{u}$, and let

$$w = (w_e^{(1)}, \dots, w_e^{(K)}) = ((w^{(1)}, (w^{(1,\ell)})_{\ell \in \mathcal{N}_{\Gamma}(1)}), \dots) \in \widetilde{W}$$

with coefficient vector $\underline{w}$ be such that $\underline{w} = B_D^T B \underline{u}$. Then we have, for each patch $\Omega^{(k)}$ and each edge $\Gamma^{(k,\ell)}$ connecting the vertices $\mathbf{x}^{(k,\ell,1)}$ and $\mathbf{x}^{(k,\ell,2)}$,

$$\begin{aligned}\|w^{(k)} - w^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 &\lesssim \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \|u^{(\ell)} - u^{(\ell,k)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \sum_{i=1}^2 \left(\frac{h_k}{p} \Delta^{(k,\ell,i)} + \frac{h_\ell}{p} \Delta^{(\ell,k,i)} \right), \\ |w^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 &\lesssim |u^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |u^{(\ell,k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + \sum_{i=1}^2 \Delta^{(k,\ell,i)}, \\ |w^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2 &\lesssim |u^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2 + |u^{(\ell,k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2 + \sum_{i=1}^2 \Delta^{(k,\ell,i)},\end{aligned}$$

where

$$\Delta^{(k,\ell,i)} := \begin{cases} 0 & \text{for Algorithms A and C,} \\ \sum_{j \in \mathcal{P}(\mathbf{x}^{(k,\ell,i)}) \cap \mathcal{N}_\Gamma(k)} |u^{(k)}(\mathbf{x}^{(k,\ell,i)}) - u^{(j,k)}(\mathbf{x}^{(k,\ell,i)})|^2 & \text{for Algorithm B.} \end{cases} \quad (4.3)$$

Proof. We start by recalling how the scaling matrix D looks like. Remember D is a diagonal matrix. We denote the coefficients by $d_{i,i}$ for $i = 1, \dots, N_\Gamma$. The entries $d_{i,i}$ are defined to be one plus the number of Lagrange multipliers of the corresponding degree of freedom i . Note that we have $d_{i,i} = 2$ unless the corresponding degree of freedom is a corner degree of freedom. In that case, $d_{i,i}$ takes a value, which is bounded from below by 1 and bounded from above due to Assumption 3. Thus, $d_{i,i} \approx 1$ in any case.

We will start with Algorithms A and C: simple calculations reveal for Algorithms A and C that

$$\begin{aligned}w^{(k)}|_{\Gamma^{(k,\ell)}} &= \frac{1}{2} \left(u^{(k)}|_{\Gamma^{(k,\ell)}} - u^{(\ell,k)} - \sum_{i=1}^2 \theta^{(k,\ell,i)} (u^{(k)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)})) \right), \\ w^{(k,\ell)} &= \frac{1}{2} \left(u^{(k,\ell)} - u^{(\ell)}|_{\Gamma^{(k,\ell)}} - \sum_{i=1}^2 \theta^{(\ell,k,i)} (u^{(k,\ell)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell)}(\mathbf{x}^{(k,\ell,i)})) \right),\end{aligned}$$

where $\theta^{(k,\ell,i)}$ is the basis function in $\Phi^{(k)}$ such that $\theta^{(k,\ell,i)}(\mathbf{x}^{(k,\ell,i)}) = 1$, and $\theta^{(\ell,k,i)}$ is defined analogously. Since u satisfies the primal constraints, we have $u^{(k)}(\mathbf{x}^{(k,\ell,i)}) = u^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)})$ and $u^{(k,\ell)}(\mathbf{x}^{(k,\ell,i)}) = u^{(\ell)}(\mathbf{x}^{(k,\ell,i)})$ and thus

$$w^{(k)}|_{\Gamma^{(k,\ell)}} = \frac{1}{2} (u^{(k)}|_{\Gamma^{(k,\ell)}} - u^{(\ell,k)}), \quad w^{(k,\ell)} = \frac{1}{2} (u^{(k,\ell)} - u^{(\ell)}|_{\Gamma^{(k,\ell)}}).$$

Therefore, we obtain, by using the triangle inequality,

$$\|w^{(k)} - w^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \lesssim \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \|u^{(\ell)} - u^{(\ell,k)}\|_{L_2(\Gamma^{(k,\ell)})}^2$$

for the L_2 -norm. For the $H^{1/2}$ -seminorm, we get using the triangle inequality that

$$|w^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 \lesssim |u^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |u^{(\ell,k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2.$$

Analogously, we obtain, using the triangle inequality,

$$|w^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2 \lesssim |u^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2 + |u^{(\ell,k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2,$$

which finishes the proof for Algorithms A and C.

For Algorithm B, we have that $w^{(k)}$ on $\Gamma^{(k,\ell)}$ can be expanded as

$$\begin{aligned}w^{(k)}|_{\Gamma^{(k,\ell)}} &= \frac{1}{2} \left(u^{(k)}|_{\Gamma^{(k,\ell)}} - u^{(\ell,k)} - \sum_{i=1}^2 \theta^{(k,\ell,i)} (u^{(k)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)})) \right) \\ &\quad + \sum_{i=1}^2 \frac{1}{d_{n_i, n_i}} \theta^{(k,\ell,i)} \sum_{j \in \mathcal{P}(\mathbf{x}^{(k,\ell,i)}) \cap \mathcal{N}_\Gamma(k)} (u^{(k)}(\mathbf{x}^{(k,\ell,i)}) - u^{(j,k)}(\mathbf{x}^{(k,\ell,i)})), \\ w^{(k,\ell)} &= \frac{1}{2} \left(u^{(k,\ell)} - u^{(\ell)}|_{\Gamma^{(k,\ell)}} - \sum_{i=1}^2 \theta^{(\ell,k,i)} (u^{(k,\ell)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell)}(\mathbf{x}^{(k,\ell,i)})) \right) \\ &\quad + \sum_{i=1}^2 \frac{1}{d_{m_i, m_i}} \theta^{(\ell,k,i)} \sum_{j \in (\mathcal{P}(\mathbf{x}^{(k,\ell,i)}) \cap \mathcal{N}_\Gamma(\ell)) \cup \{\ell\}} (u^{(k,\ell)}(\mathbf{x}^{(k,\ell,i)}) - u^{(j,\ell)}(\mathbf{x}^{(k,\ell,i)})),\end{aligned}$$

where d_{n_i, n_i} and d_{m_i, m_i} denote the entries of the matrix D for the corresponding dofs and we use $u^{(\ell, \ell)} := u^{(\ell)}$. Note that $\theta^{(k, \ell, i)}$ behaves like $\max\{0, 1 - |x - \mathbf{x}^{(k, \ell, i)}|/h^{(k)}\}^p$. Hence,

$$\|\theta^{(k, \ell, i)}\|_{L_2(\Gamma^{(k, \ell)})}^2 \approx \frac{h_k}{p} \quad \text{and} \quad |\theta^{(k, \ell, i)}|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 \approx 1.$$

Certainly, we also have $\|\theta^{(k, \ell, i)}\|_{L_\infty(\Gamma^{(k, \ell)})}^2 = 1$. An application of the triangle inequality for the corner expressions yields the stated result for Algorithm B. $\square$

4.3 Estimates for Contributions from Artificial Interfaces

The following lemma allows to estimate the $H^{1/2}$ -seminorm on the artificial edges.

Lemma 4. *For any two patches $\Omega^{(k)}$ and $\Omega^{(\ell)}$ that share an edge $\Gamma^{(k, \ell)}$,*

$$|u^{(k, \ell)}|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 \leq |u^{(k)}|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 + \frac{p^2}{h_{k\ell}} \|u^{(k, \ell)} - u^{(k)}\|_{L_2(\Gamma^{(k, \ell)})}^2$$

holds for all $u_e^{(k)} = (u^{(k)}, (u^{(k, \ell)})_{\ell \in \mathcal{N}_\Gamma(k)}) \in V_e^{(k)}$.

Proof. Let $c \in \mathbb{R}$ be arbitrary but fixed. Using the triangle inequality, we have

$$\mathcal{A} := \|u^{(k, \ell)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 \leq \|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 + \|u^{(k)} - u^{(k, \ell)}\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2.$$

Using [1, Theorem 5.2, equation (3)] and the triangle inequality, we obtain further

$$\begin{aligned} \mathcal{A} &\leq \|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 + \|u^{(k)} - u^{(k, \ell)}\|_{L_2(\Gamma^{(k, \ell)})} \|u^{(k)} - c - (u^{(k, \ell)} - c)\|_{H^1(\Gamma^{(k, \ell)})} \\ &\leq \|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 + \|u^{(k)} - u^{(k, \ell)}\|_{L_2(\Gamma^{(k, \ell)})} (\|u^{(k)} - c\|_{H^1(\Gamma^{(k, \ell)})} + \|u^{(k, \ell)} - c\|_{H^1(\Gamma^{(k, \ell)})}). \end{aligned}$$

Using the equivalence of the norms on the parameter domain and the physical domain, cf. Lemma 1, and an inverse inequality, cf. [32, Lemma 4.3], and using $h_{k\ell} = \min\{h_k, h_\ell\}$, we obtain further

$$\mathcal{A} \leq \|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 + \frac{Cp}{h_{k\ell}^{1/2}} \|u^{(k)} - u^{(k, \ell)}\|_{L_2(\Gamma^{(k, \ell)})} (\|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})} + \|u^{(k, \ell)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}),$$

where $C > 0$ is an appropriately chosen constant (that only depends on the constant from Assumption 1).

Using $a(b + c) \leq a^2 + (b + c)^2/4 \leq a^2 + b^2/2 + c^2/2$, we have

$$\mathcal{A} \leq \frac{3}{2} \|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 + \frac{C^2 p^2}{h_{k\ell}} \|u^{(k)} - u^{(k, \ell)}\|_{L_2(\Gamma^{(k, \ell)})}^2 + \frac{1}{2} \mathcal{A}.$$

By subtracting $\frac{1}{2}\mathcal{A}$, we immediately obtain

$$\mathcal{A} \leq \|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 + \frac{p^2}{h_{k\ell}} \|u^{(k)} - u^{(k, \ell)}\|_{L_2(\Gamma^{(k, \ell)})}^2$$

for all $u_e^{(k)} \in V_e^{(k)}$. Since $|u^{(k, \ell)}|_{H^{1/2}(\Gamma^{(k, \ell)})}^2 \leq \mathcal{A}$ and since the bound for $\mathcal{A}$ holds for all $c \in \mathbb{R}$, the Poincaré inequality yields the desired result. $\square$

The next step is to show that a similar estimate holds for the seminorm $|\cdot|_{L_\infty^0(T)}$. Before we can prove that result, we need the following auxiliary result.

Lemma 5. $\|u\|_{L_\infty^0(0,1)}^2 \leq \sqrt{2} \|u\|_{L_2(0,1)} \|u\|_{H^1(0,1)}$ holds for all $u \in H^1(0, 1)$.

Proof. Since u is continuous, u takes its maximum for some $z \in [0, 1]$. Then, by the fundamental theorem of differential and integral calculus, we can write $|u(z)|^2 = |u(t)|^2 - \int_z^t u(s)u'(s) ds$. The desired result is obtained by integrating over the unit interval and by utilizing the Cauchy–Schwarz inequality. $\square$

Lemma 6. For any two patches $\Omega^{(k)}$ and $\Omega^{(\ell)}$ that share an edge $\Gamma^{(k,\ell)}$,

$$|u^{(k,\ell)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2 \leq |u^{(k)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2 + |u^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + \frac{p^2}{h_{k\ell}} \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2$$

holds for all $u_e^{(k)} = (u^{(k)}, (u^{(k,\ell)})_{\ell \in \mathcal{N}_{\Gamma}(k)}) \in V_e^{(k)}$.

Proof. Using the triangle inequality, we obtain

$$|u^{(k,\ell)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2 \leq |u^{(k)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2 + |u^{(k)} - u^{(k,\ell)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2.$$

As a next step, we apply Lemma 5 to the difference $|u^{(k)} - u^{(k,\ell)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2$. Since the L_{∞} -norm does not change if we are on the physical or the parameter domain, we can apply Lemma 5 and subsequently utilize the equivalence of the norms on the parameter and the physical domain, cf. Lemma 1, to get

$$\mathcal{A} := |u^{(k)} - u^{(k,\ell)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2 \leq \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})} \|u^{(k)} - u^{(k,\ell)}\|_{H^1(\Gamma^{(k,\ell)})}.$$

The triangle inequality allows to estimate

$$\mathcal{A} \leq \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})} (\|u^{(k)} - c\|_{H^1(\Gamma^{(k,\ell)})} + \|u^{(k,\ell)} - c\|_{H^1(\Gamma^{(k,\ell)})})$$

for all $c \in \mathbb{R}$. The equivalence of the norms on the physical domain and the parameter domain, cf. Lemma 1, an inverse estimate, cf. [32, Lemma 4.3], and $a(b+c) \leq a^2 + b^2 + c^2$ give

$$\begin{aligned} \mathcal{A} &\leq \frac{p}{h_{k\ell}^{1/2}} \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})} (\|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k,\ell)})} + \|u^{(k,\ell)} - c\|_{H^{1/2}(\Gamma^{(k,\ell)})}) \\ &\leq \frac{p^2}{h_{k\ell}} \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \|u^{(k)} - c\|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + \|u^{(k,\ell)} - c\|_{H^{1/2}(\Gamma^{(k,\ell)})}^2. \end{aligned}$$

Since this holds for all $c \in \mathbb{R}$, the Poincaré inequality gives further

$$\mathcal{A} \leq \frac{p^2}{h_{k\ell}} \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + |u^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |u^{(k,\ell)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2.$$

Lemma 4 yields the final result. $\square$

Lemma 4.17 in [32] states that the term $\Delta^{(k,\ell,i)}$ can be estimated by expressions which only involve patches sharing an edge. Now we prove a variant of [32, Lemma 4.18] that fits our needs.

Lemma 7. For any two patches $\Omega^{(k)}$ and $\Omega^{(\ell)}$, sharing an edge $\Gamma^{(k,\ell)}$ which connects the vertices $\mathbf{x}^{(k,\ell,1)}$ and $\mathbf{x}^{(k,\ell,2)}$ and $\int_{\Gamma^{(k,\ell)}} u^{(k)}(s) - u^{(\ell,k)}(s) ds = 0$ and $i = 1, 2$, we have

$$\Delta^{(k,\ell,i)} \leq \Lambda (|\mathcal{H}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 + |\mathcal{H}_h^{(\ell)} u^{(\ell)}|_{H^1(\Omega^{(\ell)})}^2) + \frac{p^2}{h_{k\ell}} \|u^{(\ell)} - u^{(\ell,k)}\|_{L_2(\Gamma^{(k,\ell)})}^2$$

for all $u = (u_e^{(1)}, \dots, u_e^{(K)}) \in \widetilde{W}$, where $\Lambda = 1 + \log p + \max_{j=1,\dots,K} \log \frac{H_j}{h_j}$ and $\Delta^{(k,\ell,i)}$ is as in (4.3).

Proof. Using the triangle inequality, we have

$$\Delta^{(k,\ell,i)} \leq |u^{(k)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell)}(\mathbf{x}^{(k,\ell,i)})|^2 + |u^{(\ell)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)})|^2. \quad (4.4)$$

Let

$$\rho := \frac{1}{|\Gamma^{(k,\ell)}|} \left(\int_{\Gamma^{(k,\ell)}} u^{(k)} - u^{(\ell)} ds \right) = \frac{1}{|\Gamma^{(k,\ell)}|} \left(\int_{\Gamma^{(k,\ell)}} u^{(\ell,k)} - u^{(\ell)} ds \right),$$

and observe that the Cauchy–Schwarz inequality and Assumption 1 yield

$$\rho^2 \leq \frac{1}{|\Gamma^{(k,\ell)}|} \|u^{(\ell,k)} - u^{(\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \leq \frac{1}{h_{k\ell}} \|u^{(\ell,k)} - u^{(\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2. \quad (4.5)$$

The triangle inequality, and then (4.5) and [32, Lemma 4.18] yield

$$\begin{aligned} |u^{(k)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell)}(\mathbf{x}^{(k,\ell,i)})|^2 &\leq |(u^{(k)} - \rho)(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell)}(\mathbf{x}^{(k,\ell,i)})|^2 + |\rho|^2 \\ &\leq \Lambda(|\mathcal{I}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 + |\mathcal{I}_h^{(\ell)} u^{(\ell)}|_{H^1(\Omega^{(\ell)})}^2) + \frac{p^2}{h_{k\ell}} \|u^{(\ell,k)} - u^{(\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2. \end{aligned} \quad (4.6)$$

Now, we estimate $|u^{(\ell)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)})|^2$. Thus, we set $\hat{u}^{(\ell)} = u^{(\ell)} \circ G_\ell$ and $\hat{u}^{(\ell,k)} = u^{(\ell,k)} \circ G_k$. With unitary transformations, i.e., rotations and reflections, we can transform the patches such that the pre-image of $\Gamma^{(k,\ell)}$ is $(0, 1) \times \{0\}$ and that the pre-image of $\mathbf{x}^{(k,\ell,i)}$ is 0. Let $\tilde{u}^{(\ell)} := \hat{u}^{(\ell)}(\cdot, 0)$ and $\tilde{u}^{(\ell,k)} := \hat{u}^{(\ell,k)}(\cdot, 0)$. Let η be the largest value such that $\tilde{u}^{(\ell)}$ and $\tilde{u}^{(\ell,k)}$ are polynomial on $(0, \eta)$. Using the quasi-uniformity assumption, cf. Assumption 4, we obtain $\eta \approx \hat{h}_{k\ell}$. Arguments that are analogous to those used in the proof of Lemma 5, together with the inverse inequality [37, Theorem 4.76, equation (4.6.5)] yield

$$\begin{aligned} |\hat{u}^{(\ell)}(0) - \hat{u}^{(\ell,k)}(0)|^2 &\leq \frac{1}{\eta} \|\tilde{u}^{(\ell)} - \tilde{u}^{(\ell,k)}\|_{L_2(0,\eta)}^2 + 2\|\tilde{u}^{(\ell)} - \tilde{u}^{(\ell,k)}\|_{L_2(0,\eta)} \|\tilde{u}^{(\ell)} - \tilde{u}^{(\ell,k)}\|_{H^1(0,\eta)} \\ &\leq \frac{1}{\eta} \|\tilde{u}^{(\ell)} - \tilde{u}^{(\ell,k)}\|_{L_2(0,\eta)}^2 + \frac{p^2}{\eta} \|\tilde{u}^{(\ell)} - \tilde{u}^{(\ell,k)}\|_{L_2(0,\eta)}^2. \end{aligned}$$

Since the L_2 -norm on $(0, 1)$ is larger than the L_2 -norm on $(0, \eta)$ and $\eta \approx \hat{h}_{k\ell}$,

$$|\hat{u}^{(\ell)}(0) - \hat{u}^{(\ell,k)}(0)|^2 \leq \frac{p^2}{\hat{h}_{k\ell}} \|\tilde{u}^{(\ell)} - \tilde{u}^{(\ell,k)}\|_{L_2(0,1)}^2 = \frac{p^2}{\hat{h}_{k\ell}} \|\hat{u}^{(\ell)} - \hat{u}^{(\ell,k)}\|_{L_2(\hat{\Gamma}^{(k,\ell)})}^2.$$

An application of Lemma 1 finally yields

$$|u^{(\ell)}(\mathbf{x}^{(k,\ell,i)}) - u^{(\ell,k)}(\mathbf{x}^{(k,\ell,i)})|^2 \leq \frac{p^2}{h_{k\ell}} \|u^{(\ell)} - u^{(\ell,k)}\|_{L_2(\Gamma^{(k,\ell)})}^2. \quad (4.7)$$

The combination of (4.4), (4.6), and (4.7) finishes the proof. $\square$

4.4 Proof of the Main Theorem

Before we give a proof of the main theorem, we estimate the sum of $w_e^{(k)}$ -seminorms over all patches.

Lemma 8. *Let u and w be as in Lemma 3. Then we have*

$$\begin{aligned} \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} (|w^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |w^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2) \\ \leq \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \left(|u^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |u^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2 + \frac{p^2}{h_{k\ell}} \|u^{(k,\ell)} - u^{(k)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \right) + \Lambda \sum_{k=1}^K |\mathcal{I}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2, \end{aligned}$$

where $\Lambda := 1 + \log p + \max_{k=1,\dots,K} \log \frac{H_k}{h_k}$.

Proof. Lemma 3 yields

$$\begin{aligned} \mathcal{A} &:= \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} (|w^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |w^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2) \\ &\leq \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} (|u^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |u^{(\ell,k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |u^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2 + |u^{(\ell,k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2) + \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \sum_{i=1}^2 \Delta^{(k,\ell,i)}. \end{aligned}$$

Lemmas 4 and 6 allow to estimate the first sum. This finishes the proof for Algorithms A and C since the second sum vanishes in this case. For Algorithm B, the second sum is estimated using Lemma 7 and $|\mathcal{N}_\Gamma(k)| \leq 4$. $\square$

Lemma 9. *Let u and w be as in Lemma 3. Then we have*

$$\sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{\delta p^2}{h_{k\ell}} \|w^{(k)} - w^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \leq \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \left(\frac{\delta p^2}{h_{k\ell}} \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \Lambda |\mathcal{I}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 \right),$$

where

$$\Sigma := \begin{cases} 1 & \text{for Algorithms A and C,} \\ \delta p \max_{k=1,\dots,K} \max_{\ell \in \mathcal{N}_\Gamma(k)} \frac{h_k}{h_\ell} & \text{for Algorithm B.} \end{cases} \quad (4.8)$$

Proof. Lemma 3 and the observation that $\ell \in \mathcal{N}_\Gamma(k) \Leftrightarrow k \in \mathcal{N}_\Gamma(\ell)$ yield

$$\mathcal{A} := \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{\delta p^2}{h_{k\ell}} \|w^{(k)} - w^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \lesssim \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{\delta p^2}{h_{k\ell}} \left(\|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \sum_{i=1}^2 \frac{h_k}{p} \Delta^{(k,\ell,i)} \right).$$

Since $\Delta^{(k,\ell,i)} = 0$ for Algorithms A and C, this finishes the proof for this case. Now, consider Algorithm B. Lemma 7 and the observation that $\ell \in \mathcal{N}_\Gamma(k) \Leftrightarrow k \in \mathcal{N}_\Gamma(\ell)$ yield

$$\mathcal{A} \lesssim \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{\delta p^2}{h_{k\ell}} \left(\|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \Lambda \frac{h_k + h_\ell}{p} |\mathcal{I}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 + \frac{ph_k}{h_{k\ell}} \|u^{(\ell)} - u^{(\ell,k)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \right).$$

Then $h_{k\ell} = \min\{h_k, h_\ell\}$ and $\ell \in \mathcal{N}_\Gamma(k) \Leftrightarrow k \in \mathcal{N}_\Gamma(\ell)$ finish the proof. $\square$

Now we are able to prove the bound for the condition number of the preconditioned dG IETI method as stated in Theorem 1.

Proof of Theorem 1. Theorem 22 in [26] states that

$$\kappa(M_{\text{SD}}F) \leq \sup_{\underline{u} \in \bar{W}} \frac{\|B_D^\top B \underline{u}\|_S^2}{\|\underline{u}\|_S^2}, \quad (4.9)$$

where $\underline{u}$ is the coefficient vector associated to the function $u = (u_e^{(1)}, \dots, u_e^{(K)}) = ((u^{(1)}, (u^{(1,\ell)})_{\ell \in \mathcal{N}_\Gamma(1)}), \dots)$. So let u be arbitrary but fixed, and let the function $w = (w_e^{(1)}, \dots, w_e^{(K)}) = ((w^{(1)}, (w^{(1,\ell)})_{\ell \in \mathcal{N}_\Gamma(1)}), \dots)$ with coefficient vector $\underline{w}$ be such that $\underline{w} = B_D^\top B \underline{u}$. Let $v_e^{(k)} \in V_e^{(k)}$ be an arbitrary extension of $w_e^{(k)}$. Lemma 2 yields

$$\begin{aligned} \|B_D^\top B \underline{u}\|_S^2 &= \|\underline{w}\|_S^2 = \sum_{k=1}^K \|\mathcal{I}_A^{(k)} w_e^{(k)}\|_{a_e^{(k)}}^2 \\ &= \sum_{k=1}^K \inf_{w_0^{(k)} \in V_0^{(k)}} \|v_e^{(k)} - (w_0^{(k)}, 0)\|_{a_e^{(k)}}^2 \approx \sum_{k=1}^K \inf_{w_0^{(k)} \in V_0^{(k)}} \|v_e^{(k)} - (w_0^{(k)}, 0)\|_{d_e^{(k)}}^2 \\ &= \sum_{k=1}^K \inf_{w_0^{(k)} \in V_0^{(k)}} \left(|v^{(k)} - w_0^{(k)}|_{H^1(\Omega^{(k)})}^2 + \frac{\delta p^2}{h_{k\ell}} \sum_{\ell \in \mathcal{N}_\Gamma(k)} \|w^{(k)} - w^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \right). \end{aligned}$$

Note that the second sum does not depend on $w_0^{(k)}$. Thus, the infimum refers only to the H^1 -seminorm, which means that the seminorm coincides with the seminorm of the (standard) discrete harmonic extension. Hence, we have

$$\|B_D^\top B \underline{u}\|_S^2 \approx \underbrace{\sum_{k=1}^K |\mathcal{I}_h^{(k)} w^{(k)}|_{H^1(\Omega^{(k)})}^2}_{=: \mathcal{A}} + \underbrace{\sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{\delta p^2}{h_{k\ell}} \|w^{(k)} - w^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2}_{=: \mathcal{B}}. \quad (4.10)$$

First, we estimate $\mathcal{A}$. Due to [32, Theorem 4.2] and [32, Lemma 4.15], we get

$$\mathcal{A} \leq p \sum_{k=1}^K |w^{(k)}|_{H^{1/2}(\partial\Omega^{(k)})}^2 \lesssim p \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} (|w^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + \Lambda |w^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2).$$

Using $\Lambda \geq 1$ and Lemma 8, we obtain further

$$\begin{aligned} \mathcal{A} &\leq p\Lambda \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} (|u^{(k)}|_{H^{1/2}(\Gamma^{(k,\ell)})}^2 + |u^{(k)}|_{L_\infty^0(\Gamma^{(k,\ell)})}^2) \\ &\quad + p\Lambda \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{p^2}{h_{k\ell}} \|u^{(k,\ell)} - u^{(k)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + p\Lambda^2 \sum_{k=1}^K |\mathcal{I}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2. \end{aligned}$$

Using [32, Lemma 4.15 and Theorem 4.2] and $|\mathcal{N}_\Gamma(k)| \leq 4$, we further estimate

$$\begin{aligned} \mathcal{A} &\lesssim p\Lambda^2 \sum_{k=1}^K |\mathcal{H}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 + p\Lambda \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} |u^{(k)}|_{L_{\infty}^0(\Gamma^{(k,\ell)})}^2 \\ &\quad + p\Lambda \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{p^2}{h_{k\ell}} \|u^{(k,\ell)} - u^{(k)}\|_{L_2(\Gamma^{(k,\ell)})}^2. \end{aligned}$$

We estimate the second sum using [32, Lemma 4.14], the Poincaré inequality, Lemma 1, and $|\mathcal{N}_\Gamma(k)| \lesssim 1$ and obtain

$$\mathcal{A} \lesssim p\Lambda^2 \sum_{k=1}^K \left(|\mathcal{H}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 + \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{p^2}{h_{k\ell}} \|u^{(k,\ell)} - u^{(k)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \right). \quad (4.11)$$

To estimate the second sum in (4.10), we use Lemma 9 to get

$$\mathcal{B} \lesssim \Sigma \sum_{k=1}^K \sum_{\ell \in \mathcal{N}_\Gamma(k)} \left(\frac{\delta p^2}{h_{k\ell}} \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 + \Lambda |\mathcal{H}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 \right), \quad (4.12)$$

where Σ is as in (4.8). Then $\delta \geq 1$, $\Lambda \geq 1$, $\Sigma \geq 1$, (4.10), (4.11), and (4.12) yield

$$\begin{aligned} \|B_D^T B \underline{u}\|_S^2 &\lesssim (p\Lambda^2 + \Sigma\Lambda) \sum_{k=1}^K \left(|\mathcal{H}_h^{(k)} u^{(k)}|_{H^1(\Omega^{(k)})}^2 + \sum_{\ell \in \mathcal{N}_\Gamma(k)} \frac{\delta p^2}{h_{k\ell}} \|u^{(k)} - u^{(k,\ell)}\|_{L_2(\Gamma^{(k,\ell)})}^2 \right) \\ &= (p\Lambda^2 + \Sigma\Lambda) \|\underline{u}\|_d^2 \approx (p\Lambda^2 + \Sigma\Lambda) \|\underline{u}\|_S^2 \lesssim \max\{p, \Sigma\} \Lambda^2 \|\underline{u}\|_S^2. \end{aligned}$$

The combination of this estimate and (4.9) finishes the proof. $\square$

5 Numerical results

In this section, we present the results of our numerical experiments that illustrate the presented convergence theory. We consider the Poisson problem

$$\begin{aligned} -\Delta u(x, y) &= 2\pi^2 \sin(\pi x) \sin(\pi y) && \text{for } (x, y) \in \Omega, \\ u &= 0 && \text{on } \partial\Omega, \end{aligned}$$

where the computational domain Ω is one of the domains depicted in Figure 7.

Figure 7: Computational domains and the decomposition into patches

The first domain (Figure 7 (a)) is a circular ring consisting of 12 patches. Each of them is the image of a NURBS mapping of degree 2. The second domain (Figure 7 (b)) is the Yeti-footprint which is composed of 84 patches. In this case, all of the patches are parameterized using B-spline functions, again of degree 2.

The numerical experiments are carried out with B-spline discretization spaces of maximum smoothness on grids that are constructed as follows. For the circular ring, the coarsest grid on each patch only consists of one element, i.e., the discretization space (on the parameter domain) for each patch consists only of polynomial functions. For the Yeti-footprint, the coarsest grid for the 20 patches in the bottom of the domain consists of two elements per patch. The grid is constructed by adding an edge that connects the midpoints of the two longer sides of the patch. The other patches of the Yeti-footprint only consist of one element. For all three domains, the finer grids are constructed by refinement. For the first refinement step, i.e., for $r = 1$, we insert one knot into each knot span. The new knot is not located in the center, but at $4/9$ times the length of the knot span if the patch index k is even and at $6/11$ times that length if k is odd. The subsequent refinement steps $r = 2, 3, \dots$ are done uniformly. This refinement procedure yields discretizations that are non-matching at the interfaces. In the numerical experiments underlying Tables 1–8, the patches are not modified when the grid is refined, i.e., the patch diameters H_k remain the same for the respective numerical tests.

For these discretization spaces, we set up a dG IETI discretization as introduced in Section 3. The penalty term is chosen such that $h_{k\ell}$ is an estimate for the size of the smaller side of the smallest element of the physical patches $\Omega^{(k)}$ and $\Omega^{(\ell)}$, and $\delta = 3$ for Algorithms A and C and $\delta = 1$ for Algorithm B. To solve system (3.13), we use a preconditioned conjugate gradient (PCG) method with the proposed scaled Dirichlet preconditioner M_{SD} . The implementation is done using G+Smo [27]. The local subproblems are solved with the sparse direct solver from the PARDISO project¹.

We start the iteration with a randomly sampled vector with entries in the interval $[-1, 1]$. The stopping criterion is chosen as follows. The iteration stops if the l_2 -norm of the residual vector drops below 10^{-6} times the residual of the right-hand side. For each experiment, we show the number of iterations (it) required to reach the stopping criterion and an estimate of the condition number (κ) of the preconditioned system matrix $M_{\text{SD}}F$ that has been derived using the PCG algorithm.

5.1 Tests on the Circular Ring

We start with the numerical experiments for the circular ring (Figure 7 (a)). Table 1 shows the results for Algorithm A (vertex values). We observe that the condition number grows not more than $\log^2 H/h$. Moreover, we also observe a weak growth in the spline degree, which might be smaller than the linear growth predicted by the theory. Table 2 shows the results for Algorithm B (edge averages). Here, the condition numbers are much larger than for Algorithm A, which can be explained as follows. The condition number bound in Theorem 1 depends on δ only for Algorithm B. Estimating $h_{k\ell}$ conservatively, which is necessary in order to obtain a coercive bilinear form, can be interpreted as increasing δ . While this affects the condition number significantly, the iteration counts are not affected as much since the eigenvalues seem to be slightly clustered. The dependence on the grid size and the spline degree seem to be the same as for Algorithm A. Table 3 comprises the data collected for Algorithm C (vertex values + edge averages). As expected, this approach yields the smallest values for the condition numbers. The condition number seems to grow only like $\log H/h$ in the grid size.

5.2 Test on the Yeti-Footprint

Next, we take a look at numerical results on the Yeti-footprint (Figure 7 (b)) if the right-hand side function is less smooth than in the model problem introduced at the beginning of the section. For the Yeti-footprint, we consider a patchwise defined right-hand side that is the zero function on patches with odd index and $2\pi^2 \sin(\pi x) \sin(\pi y)$ on patches with even index. Table 4 reports on the results for Algorithm A. We can again

¹ <https://www.pardiso-project.org/>

r	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 8$	
	it	κ	it	κ	it	κ	it	κ	it	κ	it	κ
3	17	6.66	15	7.05	14	7.56	15	7.98	15	8.39	16	9.09
4	17	7.34	15	8.07	15	8.67	15	9.19	16	9.65	17	10.42
5	18	8.42	16	9.28	16	9.99	16	10.59	16	11.11	18	11.98
6	18	9.69	17	10.69	16	11.48	17	12.15	19	12.74	19	13.70
7	19	11.14	18	12.26	17	13.15	18	13.89	19	14.54	19	15.59
8	20	12.77	18	14.02	19	15.00	20	15.82	20	16.51	20	17.67

Table 1: Iteration counts (it) and condition numbers (κ), Algorithm A, triple ring

r	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 8$	
	it	κ	it	κ	it	κ	it	κ	it	κ	it	κ
3	47	80	46	107	46	133	46	157	48	179	51	241
4	49	83	47	108	46	135	46	164	48	187	51	245
5	49	86	47	111	47	143	48	171	50	200	54	258
6	51	86	48	120	48	145	50	177	50	211	54	272
7	52	95	49	125	49	155	51	184	54	218	56	284
8	53	99	50	130	52	161	53	189	56	233	58	291

Table 2: Iteration counts (it) and condition numbers (κ), Algorithm B, triple ring

r	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 8$	
	it	κ	it	κ	it	κ	it	κ	it	κ	it	κ
3	16	5.80	12	3.57	11	3.00	11	3.09	12	3.28	12	3.62
4	16	5.95	12	3.68	12	3.43	12	3.69	13	3.92	13	4.30
5	17	6.08	13	3.78	13	4.09	13	4.38	13	4.63	14	5.05
6	17	6.10	14	4.43	14	4.82	14	5.14	14	5.41	15	5.86
7	17	6.18	15	5.19	15	5.61	15	5.95	14	6.25	15	6.73
8	18	6.27	16	6.01	16	6.46	16	6.83	15	7.14	16	7.65

Table 3: Iteration counts (it) and condition numbers (κ), Algorithm C, triple ring

r	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 8$	
	it	κ	it	κ	it	κ	it	κ	it	κ	it	κ
2	12	3.00	14	3.63	15	4.19	16	4.65	18	5.07	20	5.80
3	15	3.98	16	4.69	17	5.33	18	5.85	19	6.27	22	7.02
4	17	5.10	19	5.96	19	6.65	21	7.22	21	7.70	23	8.53
5	19	6.38	21	7.35	21	8.11	22	8.74	23	9.31	24	10.18
6	21	7.82	23	8.89	23	9.75	24	10.43	25	11.04	26	12.03
7	24	9.43	24	10.62	26	11.61	26	12.34	27	12.99	29	14.14

Table 4: Iteration counts (it) and condition numbers (κ), Algorithm A, Yeti-footprint

r	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 8$	
	it	κ	it	κ	it	κ	it	κ	it	κ	it	κ
2	25	9.47	30	12.85	33	16.49	39	20.39	44	24.51	54	35.08
3	27	10.29	32	14.15	36	18.22	41	22.50	45	26.85	55	37.14
4	29	11.26	34	15.69	38	20.21	43	25.45	48	29.83	57	41.77
5	31	12.54	36	17.46	41	22.43	46	28.50	51	34.48	60	45.37
6	33	13.95	38	19.33	44	25.02	49	31.59	53	38.00	62	48.45
7	35	15.54	41	21.36	46	27.64	51	34.34	55	40.64	65	54.98

Table 5: Iteration counts (it) and condition numbers (κ), Algorithm B, Yeti-footprint

Figure 8: Condition numbers, Algorithm A, Yeti-footprint

Figure 9: Condition numbers; Algorithm B, Yeti-footprint

r	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		$p = 8$	
	A	C	A	C	A	C	A	C	A	C	A	C
2	0.1	0.1	0.2	0.1	0.2	0.2	0.3	0.2	0.5	0.4	0.9	0.6
3	0.3	0.2	0.5	0.3	0.7	0.5	0.9	0.6	1.2	0.9	2.2	1.5
4	1.4	1.0	2.0	1.3	2.5	1.8	3.4	2.2	3.7	2.5	5.6	3.7
5	5.8	3.8	8.2	5.3	10.1	7.0	13.1	9.1	16.3	11.5	24.7	17.7
6	25.5	17.4	39.3	26.3	53.0	38.4	79.4	53.4	98.6	67.9	132.5	98.0
7	161.9	110.8	233.4	158.5	326.7	218.0	404.1	283.1	485.8	327.6	711.5	495.2

Table 6: Solving times (sec.) for Algorithms A and C, single core, Yeti-footprint

see that the increase in the condition number is like $\log^2 H/h$, and the increase in p is sub-linear, almost like $\sqrt{p}$. Table 5 gives the condition number estimates for Algorithm B, where we observe that the condition number grows almost linearly in the spline degree p (which indicates that the convergence theory might be sharp in this respect). Again, Algorithm B yields the largest condition numbers. The graphs in Figures 8 and 9 visualize the condition number growths for Algorithms A and B on the Yeti-footprint, respectively. Compared to the circular ring, the condition numbers for the Yeti-footprint are smaller, which might be connected to a more regular geometry mapping. Lastly, we compare the solving times between Algorithms A

e	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		Algorithm
	it	κ	it	κ	it	κ	it	κ	it	κ	
1	17	9	16	10	16	11	16	11	16	12	A
2	17	9	15	10	16	11	17	11	17	12	
3	17	9	16	10	16	11	17	11	18	12	
1	49	88	48	123	49	155	53	191	57	227	B
2	52	109	54	161	59	214	63	267	67	331	
3	60	149	62	249	69	354	75	452	80	684	

Table 7: Iteration counts (it) and condition numbers (κ), triple ring

e	$p = 2$		$p = 3$		$p = 4$		$p = 5$		$p = 6$		Algorithm
	it	κ	it	κ	it	κ	it	κ	it	κ	
1	18	6	20	6	21	7	22	8	23	8	A
2	19	6	20	7	21	7	22	8	23	8	
3	20	6	21	7	22	7	22	8	23	9	
1	30	12	36	17	41	23	46	29	51	35	B
2	35	20	42	30	49	41	55	54	61	67	
3	44	38	56	61	66	86	74	115	84	147	

Table 8: Iteration counts (it) and condition numbers (κ), Yeti-footprint

and C in Table 6. The solving times are given in seconds and are measured on a single core on the Radon1 cluster² located in Linz. We see that Algorithm C is considerably faster than Algorithm A despite the fact that the primal problem is larger for Algorithm C (392 dofs) than for Algorithm A (192 dofs).

5.3 Dependence on h_k/h_e

The next experiment shows that the dependence of the condition number on the ratio of the grid sizes of neighboring patches is only present for Algorithm B. For this purpose, we compare Algorithm A with Algorithm B on both computational domains for particular grids that are constructed as follows. Starting from the initial grid introduced above, we applied 5 refinement steps for the ring domain and 4 refinement steps for the Yeti-footprint in the same way mentioned above. Then we uniformly refine the grids on all patches $\Omega^{(k)}$, where k is even, additionally $e \in \{1, 2, 3\}$ times. Thus, the grid sizes between patches with even and odd degrees vary by a factor of 2^e . We observe in Table 7 for the circular ring and in Table 8 for the Yeti-footprint that the condition number is almost independent of e if Algorithm A is used, while it increases like 2^e if Algorithm B is chosen. This means that the condition number grows linearly in the ratio of the grid sizes, which coincides with the prediction of the convergence theory.

5.4 Discretization Errors

In this section, we report the discretization errors and the corresponding rates. Here, we focus on the circular ring depicted in Figure 7 (a). We slightly modify the problem in order to know the exact solution. We consider inhomogeneous Dirichlet boundary conditions $u = u^*$ on $\partial\Omega$, where

$$u^*(x, y) := \sin(\pi x) \sin(\pi y)$$

² <https://www.ricam.oeaw.ac.at/hpc/>

r	$p = 2$		$p = 3$		$p = 4$		$p = 5$	
	$\epsilon_{r,p}$	$\iota_{r,p}$	$\epsilon_{r,p}$	$\iota_{r,p}$	$\epsilon_{r,p}$	$\iota_{r,p}$	$\epsilon_{r,p}$	$\iota_{r,p}$
2	0.9068		0.7795		0.6588		0.6440	
3	0.5852	0.63	0.5841	0.42	0.5577	0.24	0.5298	0.28
4	0.2838	1.04	0.2488	1.32	0.1448	1.94	0.1700	1.64
5	0.0389	2.87	0.0124	4.32	0.0025	5.86	8.3e-4	7.67
6	0.0077	2.33	7.6e-4	4.04	8.8e-5	4.82	9.7e-6	6.42
7	0.0018	2.13	8.0e-5	3.24	4.3e-6	4.36	2.3e-7	5.40
Expected:	2.00		3.00		4.00		5.00	

Table 9: Discretization errors ($\epsilon_{r,p}$) and rates ($\iota_{r,p}$), Algorithm A, triple ring Figure 10: Discretization errors ($\epsilon_{r,p}$), Algorithm A, triple ring

and observe that u^* is the exact solution of the boundary value problem. The refinement procedure and the solvers are as outlined above. We stop the iteration if the ℓ_2 -norm of the residual vector drops below 10^{-8} times the residual of the right-hand side vector. In Table 9 and Figure 10, we present the discretization error $\epsilon_{r,p}$ and the corresponding rate $\iota_{r,p}$, which are defined by

$$\epsilon_{r,p} := \frac{|u_{h_{r,p}} - u^*|_{H^1(\Omega)}}{|u^*|_{H^1(\Omega)}} \quad \text{and} \quad \iota_{r,p} := \log_2 \left(\frac{|u_{h_{r-1,p}} - u^*|_{H^1(\Omega)}}{|u_{h_{r,p}} - u^*|_{H^1(\Omega)}} \right).$$

We give the results for Algorithm A. We observe that the error rates are better than predicted by (2.5). They have the tendency to decrease as the grid is refined. Since the discretization error is almost independent of the method used to solve the linear system, we obtain for Algorithm B and C almost the same results.

5.5 Patch Scaling Tests

Finally, we examine the dependence of the condition number on the number of patches in practice. We use the circular ring with its $K = 12$ patches as initial configuration. This serves as the computational domain with the fewest number of patches. The decompositions into $K = 48$, 192, and 768 patches are obtained by splitting the patches uniformly. On each patch, we consider spline spaces that are obtained by $r = 5$ refinement steps that are done in the same way as mentioned above. Table 10 shows that the condition numbers for Algorithms A and B increase only mildly. In the first step from 12 patches to 48 patches, one obtains the largest increase since in the initial configuration with 12 patches still most of the patches contribute to the Dirichlet boundary.

K	p						Algorithm
	2	3	4	5	6	7	
12	8	9	10	11	11	12	A
48	24	22	24	25	26	27	
192	29	31	33	34	35	37	
768	32	34	36	38	40	41	
12	86	111	143	171	200	234	B
48	155	201	267	307	371	434	
192	167	221	290	346	388	438	
768	180	242	292	346	392	441	

Table 10: Condition numbers (κ) depending on the number of patches K , triple ring

6 Conclusions

We have extended the theory from [32], where we established p -explicit condition number estimates for continuous Galerkin IETI-DP solvers, to symmetric interior penalty discontinuous Galerkin discretizations. These discretizations are of particular interest in the area of Isogeometric Analysis since they allow geometry functions and spline spaces that do not coincide on the interfaces. Again, we have analyzed both the dependence on the grid size and the spline degree. If the vertex values are chosen as primal degrees of freedom (Algorithms A and C), the results are the same as for conforming discretizations. If we use only the edge averages (Algorithm B), the condition number estimate additionally depends on the ratio between the grid sizes of neighboring patches. This can also be observed in the numerical experiments.

Algorithm B does not perform as good as the other options. However, this approach seems to be beneficial if a non-conforming decomposition of the overall domain into patches is considered, like a decomposition with T-junctions. Although the IETI-DP methods also perform well on domains with non-trivial geometry functions, analyzing the dependence of the condition number on the geometry function is an interesting topic for future research.

Acknowledgment: Moreover, the authors want to thank Ulrich Langer for fruitful discussions and help with the study of existing literature. Finally, the authors thank the anonymous reviewers. Their comments have helped to significantly improve the presentation of this paper.

Funding: The first author was supported by the Austrian Science Fund (FWF): S117 and W1214-04. Also, the second author has received support from the Austrian Science Fund (FWF): P31048. This work was also supported by the bilateral project WTZ BG 03/2019 (KP-06-Austria/8/2019), funded by OeAD (Austria) and Bulgarian National Science Fund.

References

- [1] R. A. Adams and J. J. F. Fournier, *Sobolev Spaces*, Elsevier/Academic, Amsterdam, 2003.
- [2] M. Ainsworth, A preconditioner based on domain decomposition for h - p finite-element approximation on quasi-uniform meshes, *SIAM J. Numer. Anal.* **33** (1996), no. 4, 1358–1376.
- [3] P. F. Antonietti and B. Ayuso, Schwarz domain decomposition preconditioners for discontinuous Galerkin approximations of elliptic problems: Non-overlapping case, *ESAIM Math. Model. Numer. Anal.* **41** (2007), no. 1, 21–54.
- [4] P. F. Antonietti and P. Houston, A class of domain decomposition preconditioners for hp -discontinuous Galerkin finite element methods, *J. Sci. Comput.* **46** (2011), no. 1, 124–149.
- [5] D. N. Arnold, An interior penalty finite element method with discontinuous elements, *SIAM J. Numer. Anal.* **19** (1982), no. 4, 742–760.

- [6] D. N. Arnold, F. Brezzi, B. Cockburn and L. D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, *SIAM J. Numer. Anal.* **39** (2001/02), no. 5, 1749–1779.
- [7] Y. Bazilevs, L. Beirão da Veiga, J. A. Cottrell, T. J. R. Hughes and G. Sangalli, Isogeometric analysis: Approximation, stability and error estimates for h -refined meshes, *Math. Models Methods Appl. Sci.* **16** (2006), no. 7, 1031–1090.
- [8] C. Canuto, L. F. Pavarino and A. B. Pieri, BDDC preconditioners for continuous and discontinuous Galerkin methods using spectral/ hp elements with variable local polynomial degree, *IMA J. Numer. Anal.* **34** (2014), no. 3, 879–903.
- [9] J. A. Cottrell, T. J. R. Hughes and Y. Bazilevs, *Isogeometric Analysis—Toward Integration of CAD and FEA*, John Wiley & Sons, Chichester, 2009.
- [10] L. Diosady and D. Darmofal, BDDC for higher-order discontinuous Galerkin discretizations, in: *Domain Decomposition Methods in Science and Engineering 20*, Lect. Notes Comput. Sci. Eng. 91, Springer, Heidelberg (2013), 559–567.
- [11] M. Dryja, J. Galvis and M. Sarkis, BDDC methods for discontinuous Galerkin discretization of elliptic problems, *J. Complexity* **23** (2007), no. 4–6, 715–739.
- [12] M. Dryja, J. Galvis and M. Sarkis, A FETI-DP preconditioner for a composite finite element and discontinuous Galerkin method, *SIAM J. Numer. Anal.* **51** (2013), no. 1, 400–422.
- [13] J. A. Evans and T. J. R. Hughes, Explicit trace inequalities for isogeometric analysis and parametric hexahedral finite elements, *Numer. Math.* **123** (2013), no. 2, 259–290.
- [14] C. Farhat and F.-X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, *Internat. J. Numer. Methods Engrg.* **32** (1991), no. 6, 1205–1227.
- [15] B. Guo and W. Cao, Additive Schwarz methods for the h - p version of the finite element method in two dimensions, *SIAM J. Sci. Comput.* **18** (1997), no. 5, 1267–1288.
- [16] C. Hofer, Analysis of discontinuous Galerkin dual-primal isogeometric tearing and interconnecting methods, *Math. Models Methods Appl. Sci.* **28** (2018), no. 1, 131–158.
- [17] C. Hofer and U. Langer, Dual-primal isogeometric tearing and interconnecting solvers for multipatch dG-IgA equations, *Comput. Methods Appl. Mech. Engrg.* **316** (2017), 2–21.
- [18] C. Hofer and U. Langer, Dual-primal isogeometric tearing and interconnecting methods, in: *Contributions to Partial Differential Equations and Applications*, Comput. Methods Appl. Sci. 47, Springer, Cham (2019), 273–296.
- [19] C. Hofer, U. Langer and I. Touloupoulos, Isogeometric analysis on non-matching segmentation: Discontinuous Galerkin techniques and efficient solvers, *J. Appl. Math. Comput.* **61** (2019), no. 1–2, 297–336.
- [20] T. J. R. Hughes, J. A. Cottrell and Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Comput. Methods Appl. Mech. Engrg.* **194** (2005), no. 39–41, 4135–4195.
- [21] A. Klawonn, L. F. Pavarino and O. Rheinbach, Spectral element FETI-DP and BDDC preconditioners with multi-element subdomains, *Comput. Methods Appl. Mech. Engrg.* **198** (2008), no. 3–4, 511–523.
- [22] S. K. Kleiss, C. Pechstein, B. Jüttler and S. Tomar, IETI—iseogeometric tearing and interconnecting, *Comput. Methods Appl. Mech. Engrg.* **247/248** (2012), 201–215.
- [23] V. G. Korneev and U. Langer, *Dirichlet–Dirichlet Domain Decomposition Methods for Elliptic Problems*, World Scientific, Hackensack, 2015.
- [24] U. Langer, A. Mantzaflaris, S. E. Moore and I. Touloupoulos, Multipatch discontinuous Galerkin isogeometric analysis, in: *Isogeometric Analysis and Applications 2014*, Lect. Notes Comput. Sci. Eng. 107, Springer, Cham (2015), 1–32.
- [25] U. Langer and I. Touloupoulos, Analysis of multipatch discontinuous Galerkin IgA approximations to elliptic boundary value problems, *Comput. Vis. Sci.* **17** (2015), no. 5, 217–233.
- [26] J. Mandel, C. R. Dohrmann and R. Tezaur, An algebraic theory for primal and dual substructuring methods by constraints, *Appl. Numer. Math.* **54** (2005), no. 2, 167–193.
- [27] A. Mantzaflaris, R. Schneckleitner, S. Takacs, G+Smo (Geometry plus Simulation modules), <http://github.com/gismo>, 2020.
- [28] L. F. Pavarino, BDDC and FETI-DP preconditioners for spectral element discretizations, *Comput. Methods Appl. Mech. Engrg.* **196** (2007), no. 8, 1380–1388.
- [29] L. F. Pavarino and O. B. Widlund, A polylogarithmic bound for an iterative substructuring method for spectral elements in three dimensions, *SIAM J. Numer. Anal.* **33** (1996), no. 4, 1303–1335.
- [30] C. Pechstein, *Finite and Boundary Element Tearing and Interconnecting Solvers for Multiscale Problems*, Springer, Heidelberg, 2013.
- [31] B. Rivière, *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations*, Society for Industrial and Applied Mathematics, Philadelphia, 2008.
- [32] R. Schneckleitner and S. Takacs, Condition number bounds for IETI-DP methods that are explicit in h and p , *Math. Models Methods Appl. Sci.* **30** (2020), no. 11, 2067–2103.
- [33] R. Schneckleitner and S. Takacs, IETI-DP for conforming multi-patch Isogeometric Analysis in three dimensions, preprint (2021), <https://arxiv.org/abs/2103.04801>.
- [34] R. Schneckleitner and S. Takacs, Towards a IETI-DP solver on non-matching multi-patch domains, preprint (2021), <https://arxiv.org/abs/2103.02536>.

- [35] J. Schöberl and C. Lehrenfeld, Domain decomposition preconditioning for high order hybrid discontinuous Galerkin methods on tetrahedral meshes, in: *Advanced Finite Element Methods and Applications*, Lect. Notes Appl. Comput. Mech. 66, Springer, Heidelberg (2013), 27–56.
- [36] J. Schöberl, J. M. Melenk, C. G. A. Pechstein and S. C. Zaglmayr, Schwarz preconditioning for high order simplicial finite elements, in: *Domain Decomposition Methods in Science and Engineering 26*, Lect. Notes Comput. Sci. Eng. 55, Springer, Berlin (2007), 139–150.
- [37] C. Schwab, *p- and hp-Finite Element Methods: Theory and Applications in Solid and Fluid Mechanics*, The Clarendon, New York, 1998.
- [38] S. Takacs, A quasi-robust discretization error estimate for discontinuous Galerkin isogeometric analysis, preprint (2019), <https://arxiv.org/abs/1901.03263>.
- [39] A. Toselli and O. Widlund, *Domain Decomposition Methods—Algorithms and Theory*, Springer, Berlin, 2005.