

Fast multigrid solvers for conforming and non-conforming multi-patch Isogeometric Analysis

S. Takacs

RICAM-Report 2019-15

Fast multigrid solvers for conforming and non-conforming multi-patch Isogeometric Analysis

Stefan Takacs^a

^a*Johann Radon Institute for Computational and Applied Mathematics (RICAM),
Austrian Academy of Sciences*

Abstract

Isogeometric Analysis is a high-order discretization method for boundary value problems that uses a number of degrees of freedom which is as small as for a low-order method. Standard isogeometric discretizations require a global parameterization of the computational domain. In non-trivial cases, the domain is decomposed into patches having separate parameterizations and separate discretization spaces. If the discretization spaces agree on the interfaces between the patches, the coupling can be done in a conforming way. Otherwise, non-conforming discretizations (utilizing discontinuous Galerkin approaches) are required. The author and his coworkers have previously introduced multigrid solvers for Isogeometric Analysis for the conforming case. In the present paper, these results are extended to the non-conforming case. Moreover, it is shown that the multigrid solves get even more powerful if the proposed smoother is combined with a (standard) Gauss-Seidel smoother.

Keywords: Isogeometric Analysis, Multi-patch domains, Symmetric interior penalty discontinuous Galerkin

1. Introduction

Isogeometric Analysis (IgA), see [18], is a spline based approach for approximating the solution of a boundary value problem (BVP). One of the big strengths of IgA is that it has the approximation power of a high-order method while the number of degrees of freedom behaves basically like for a low-order method. To obtain this behavior, we have to be able to increase the spline degree while we simultaneously increase the smoothness. In IgA, this is typically called k -refinement and leads to spline based discretizations.

Email address: stefan.takacs@ricam.oeaw.ac.at (Stefan Takacs)

In IgA, the computational domain (usually called physical domain) is parameterized using spline or NURBS functions. Since it might be too restrictive to parameterize the whole computational domain using just one global (smooth) geometry function, one typically represents the physical domain as the union of subdomains, in IgA called patches. Then, each of the patches is parameterized by its own geometry function (multi-patch IgA).

On each patch, a space of trial and test functions is introduced. The simplest approach to set up such a function space is to use tensor-product B-splines on the unit square or unit cube (parameter domain) and to use the geometry function to map them onto the physical domain or, in the multi-patch case, onto one of the patches. If we set up the function spaces such that the basis functions on the interfaces between the patches agree, we can use conforming discretizations. Approximation errors (cf. [18, 4, 5, 6, 31, 11, 24, 9, 32] and many others) and multigrid solvers (cf. [12, 10, 17, 16] and others) for such discretizations have been previously discussed. Since we are interested in k -refinement, we need results that are explicit in the spline degree. For the single-patch case, such error estimates have originally been given in [31] and later improved in [24]. In [16], a robust single-patch multigrid solver has been proposed and analyzed based on the error estimates from [31]. In [29], both the approximation error estimates and the multigrid solver have been extended to the conforming multi-patch case. These results are the foundation of the present paper.

If conforming discretizations are not feasible, discontinuous Galerkin (dG) approaches are possible. One standard dG approach is the Symmetric Interior Penalty discontinuous Galerkin (SIPG) method, see [2, 3]. Already in [20, 21], it has been proposed to utilize these approaches to couple patches in IgA. Recently, also the dependence of the approximation error on the spline degree has been analyzed, see [30]. It was not possible to show that the approximation error is robust in the spline degree but it could be proven that it only grows logarithmically.

(Robust) multigrid solvers for such non-conforming discretizations are not known so far. In the present paper, it is shown how the multigrid solver from [29] can be extended to SIPG discretizations; we observe – as in [29] – that the numerical experiments show both robustness in the grid size and the spline degree. For completeness, we also show how to extend the convergence analysis from [29] to SIPG discretizations. It is worth noting that there are alternative solvers for multi-patch IgA, like FETI-type approaches, cf. [19, 15] and others, overlapping Schwarz type methods, cf. [7], or BDDC methods, cf. [8]; most of them, however, have not been worked out for the non-conforming case.

Note that the idea behind the proposed subspace corrected mass smoother is that the boundary value problem on the physical domain (on one patch) can be well approximated by a boundary value problem on the parameter domain. Thus, the tensor-product structure on the parameter domain can be used. This is true if the geometry function is not too distorted. Otherwise, the convergence behavior suffers significantly. The same behavior can be observed by

other fast solvers that are based on the same idea, cf. the fast diagonalization method [25]. Here, the authors have improved their method by incorporating the geometry information into the preconditioner, cf. [23]. For the multigrid setting, it has turned out that one can overcome these problems quite well if the subspace corrected mass smoother is combined with a Gauss-Seidel smoother (hybrid smoother) since both approaches have strengths that seem to be somewhat orthogonal to each other (robustness in spline degree vs. robustness in the geometry), cf. also [28].

In the present paper, we illustrate our findings with numerical experiments. All presented numerical experiments are available in the G+Smo library [22].

This paper is organized as follows. We give the model problem and a conforming discretization in Section 2. Then, in Section 3, we discuss why a non-conforming discretization might be of interest. Moreover, we propose a discontinuous Galerkin approach that fits our needs. We proceed to multigrid solvers: In Section 4, we discuss Gauss-Seidel smoothers and their performance. Motivated by that section, we introduce a subspace corrected mass smoother in Section 5 and finally a hybrid smoother in Section 6. In Section 7, we conclude and give some outlook. The Appendix finally contains the proofs of the theorems stated in the paper.

2. Model problem and standard Galerkin discretization

Let $\Omega \subset \mathbb{R}^d$ with $d \in \{2, 3\}$ be an open and simply connected Lipschitz domain. Most of the numerical experiments are done for the two-dimensional domains shown in Figure 1.

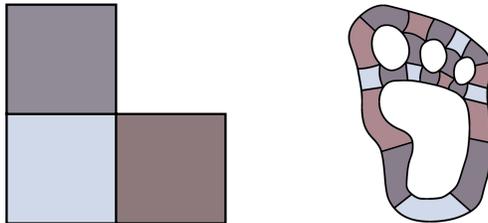


Figure 1: The computational domains: L-shaped domain and Yeti footprint

The first domain is an L-shaped domain consisting of three quadrilaterals. Here, the geometry function is just the identity or a translation. On the coarsest grid level $\ell = 0$, each patch consists only of one element, i.e., the local basis functions are polynomials only. The second domain is the Yeti footprint which consists of the 21 patches depicted in Figure 1. Here, the grid on the coarsest grid level is as follows. The five patches at the bottom consist of two elements each, which are constructed by subdividing the patches on their longer sides. The remaining

patches consist only of one element each. The grid levels $\ell = 1, 2, \dots$, are obtained by uniform refinement.

Consider the following *Poisson model problem with Dirichlet boundary conditions*. Find $u \in H^1(\Omega)$ such that

$$-\Delta u = f \quad \text{in } \Omega, \quad u = g \quad \text{on } \partial\Omega, \quad (1)$$

where $f \in L_2(\Omega)$ and $g \in H^2(\Omega)$ are given functions. Here and in what follows, $L_2(\Omega)$, $H^r(\Omega)$ and $H_0^r(\Omega)$ are the standard Lebesgue and Sobolev spaces.

The experiments are performed for the choice $g(x, y) := \sin(\pi x) \sin(\pi y)$ and $f := -\Delta g$; note that g is the exact solution of the problem.

After homogenization ($u := u - g$, $f := f + \Delta g$), the problem reads in variational form as follows. Find $u \in V := H_0^1(\Omega)$ such that

$$(\nabla u, \nabla v)_{L_2(\Omega)} = (f, v)_{L_2(\Omega)} \quad \text{for all } v \in H_0^1(\Omega). \quad (2)$$

The computational domain Ω is a standard multi-patch domain. Thus, we assume that Ω is composed of K non-overlapping patches Ω_k :

$$\overline{\Omega} = \bigcup_{k=1}^K \overline{\Omega_k} \quad \text{with} \quad \Omega_k \cap \Omega_l = \emptyset \quad \text{for } k \neq l, \quad (3)$$

where each patch is represented by a sufficiently smooth bijective geometry function

$$G_k : \widehat{\Omega} := (0, 1)^d \rightarrow \Omega_k := G_k(\widehat{\Omega}) \subset \mathbb{R}^d \quad (4)$$

which can be continuously extended to $\overline{\widehat{\Omega}}$, the closure of $\widehat{\Omega}$. Moreover, we assume that the mesh introduced by the patches satisfies the following condition.

Assumption 1. *For any $k \neq l$, the intersection $\overline{\Omega_k} \cap \overline{\Omega_l}$ is either (a) empty, (b) one common vertex, (c) the closure of one common edge, or – for $d = 3$ – (d) the closure of one common face.*

For each of the patches, we assume to have a hierarchy of grids with levels $\ell = 0, 1, \dots, L$ obtained by uniform refinement, which we denote by

$$V_{k,\ell} := \{v \in L_2(\Omega_k) : v \circ G_k \in \bigotimes_{\delta=1}^d S_{p,h_\ell}\} = \text{span} \{\varphi_{k,\ell}^{(i)}\}_{i=1}^{N_{k,\ell}}, \quad (5)$$

where $\bigotimes_{\delta=1}^d S_{p,h_\ell}$ is the space of tensor-product splines of degree p , smoothness $H^p(\widehat{\Omega})$ (or, equivalently, $C^{p-1}(\widehat{\Omega})$) and grid size $h_\ell = 2^\ell h_0$ on the parameter domain $\widehat{\Omega}$. Note that the grid can be non-uniform and both the spline degree and the grid size can depend on the spatial direction and of the patch number; for simplicity, we do not write down this dependence explicitly. Note that the grid needs to be quasi-uniform, i.e., there needs to be a constant $c > 0$ such that

all knot spans on grid level ℓ are bounded from below by ch_ℓ . The functions $\varphi_{k,\ell}^{(i)}$ are assumed to form a (standard) B-spline or NURBS basis of $V_{k,\ell}$.

To be able to set up a conforming discretization, we need to assume that the function spaces are fully matching on the interfaces, cf. [29, Assumption 2.4]. For tensor-product B-spline bases, the following assumption characterizes fully matching discretizations.

Assumption 2. *On each interface between two patches, the geometry functions, the knot vector in tangential direction, and the spline degree in tangential direction agree.*

Assuming a fully matching discretization, we define the conforming discretization space by

$$V_\ell^c := \{v \in V : v|_{\Omega_k} \in V_{k,\ell} \text{ for } k = 1, \dots, K\}. \quad (6)$$

A basis for this space is visualized in Figure 2 (left), where all basis functions are represented by their Greville point. The support of the basis functions with Greville point in the interior of a patch is completely contained in that patch. The basis functions with Greville points on the interfaces are combinations of the matching patch-local basis functions. Their support extends to the vertices if and only if the Greville point is located on the vertex.

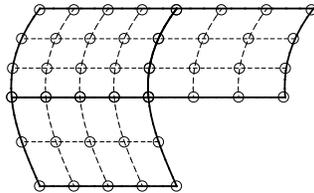


Figure 2: Degrees of freedom (represented by Greville points) in conforming case

The conforming discretization of the model problem is obtained using the standard Galerkin principle: Find $u_\ell \in V_\ell^c$ such that

$$(u_\ell, v_\ell)_{H^1(\Omega)} = (f, v_\ell)_{L_2(\Omega)} \quad \text{for all } v_\ell \in V_\ell^c. \quad (7)$$

Using the abovementioned basis for the space V_ℓ^c , we obtain a standard matrix-vector problem: Find $\underline{u}_\ell \in \mathbb{R}^{N_\ell}$ such that

$$A_\ell \underline{u}_\ell = \underline{f}_\ell, \quad (8)$$

where A_ℓ is the stiffness matrix, the vector \underline{u}_ℓ is the representation of u_ℓ with respect to the chosen basis and the load vector \underline{f}_ℓ is obtained by testing the function f with the basis functions.

3. Symmetric interior penalty discontinuous Galerkin (SIPG) discretization

Following [20, 21, 30], we use a conforming isogeometric discretization for each patch and couple the patches using discontinuous Galerkin. We assume that the domain Ω is again subdivided into patches such that (3), (4) and Assumption 1 are satisfied. We assume again to have patch-local spaces $V_{k,\ell}$ as in (5), which are combined in a non-conforming (i.e., discontinuous) way, i.e., we just define

$$V_\ell^n := \{v \in L_2(\Omega) : v|_{\Omega_k} \in V_{k,\ell} \text{ for } k = 1, \dots, K \text{ and } v|_{\partial\Omega} = 0\}. \quad (9)$$

This allows us to drop Assumption 2. Note that we strongly enforce the Dirichlet boundary conditions in our example; alternatively, one could use the SIPG method also to enforce the Dirichlet boundary conditions.

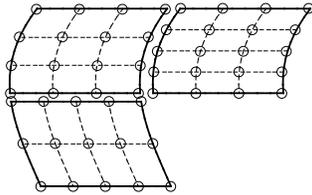


Figure 3: Degrees of freedom (represented by Greville points) in non-conforming case

Since we have a discontinuous function space, we can visualize the degrees of freedom by tearing apart the individual patches, cf. Figure 3. Here, neither the Greville points nor the basis functions need to agree on the interfaces; the support of each basis function is contained in one single patch.

Since $V_\ell^n \not\subset V$, it is not feasible to use the standard Galerkin principle for discretization. Thus, we couple the patches using the Symmetric Interior Penalty discontinuous Galerkin (SIPG) method. First, we define

$$\mathcal{N} := \{(k, j) : k < j \text{ with } \Omega_k \text{ and } \Omega_l \text{ have a common edge.}\}$$

to be the set of interface-indices. For each interface $I_{k,j}$ with $(k, l) \in \mathcal{N}$, we define the following symbols.

- \mathbf{n} is the outer normal vector of Ω_k . (Thus, $-\mathbf{n}$ is the outer normal vector of Ω_l .)
- $[[\cdot]]$ is the jump operator: $[[u]] := u|_{\Omega_k} - u|_{\Omega_l}$.
- $\{\cdot\}$ is the averaging operator: $\{u\} := \frac{1}{2}(u|_{\Omega_k} + u|_{\Omega_l})$.

Now, we can formulate the SIPG discretization as follows: Find $u_\ell \in V_\ell^n$ such that

$$(u_\ell, v_\ell)_{A_\ell} = (f, v_\ell)_{L_2(\Omega)} \quad \text{for all } v_\ell \in V_\ell^n, \quad (10)$$

where we define

$$\begin{aligned}
(u, v)_{A_\ell} &:= (u, v)_{Q_\ell} - (u, v)_{B_\ell} - (v, u)_{B_\ell}, & (u, v)_{Q_\ell} &:= (u, v)_{K_\ell} + \frac{\sigma p^2}{h_L} (u, v)_{J_\ell}, \\
(u, v)_{K_\ell} &:= \sum_{k=1}^K (\nabla u, \nabla v)_{L_2(\Omega_k)}, & (u, v)_{J_\ell} &:= \sum_{(k,l) \in \mathcal{N}} (\llbracket u \rrbracket, \llbracket v \rrbracket)_{L_2(I_{k,l})}, \\
(u, v)_{B_\ell} &:= \sum_{(k,l) \in \mathcal{N}} (\llbracket u \rrbracket, \{\nabla v\} \cdot \mathbf{n})_{L_2(I_{k,l})}.
\end{aligned} \tag{11}$$

There is some $\sigma_0 > 0$ independent of the grid size, the spline degree and the number of patches such that for all $\sigma \geq \sigma_0$, the bilinear form $(\cdot, \cdot)_{A_\ell}$ is coercive, cf. [30, Theorems 8 and 9]. Thus, for $\sigma \geq \sigma_0$, the Theorem of Lax Milgram states that the problem (10) has exactly one solution. The combination of Ceá's Lemma and a naive approximation error estimate yields a discretization error estimate of the form

$$|u - u_L|_{Q_L}^2 \leq c p^2 h_L^2 |u|_{H^2(\Omega)}^2,$$

cf. [30]. By doing a more careful analysis, we obtain estimates of the form

$$|u - u_L|_{Q_L}^2 \leq c (\log p)^4 h_L^2 |u|_{H^2(\Omega)}^2,$$

see [30, eq. (15)]. This significantly decreases the influence of the spline degree.

Note that the penalization term has the form

$$\frac{\sigma p^2}{h_L},$$

i.e., it depends on the grid size on the finest grid h_L . This follows the ideas from [13]. The idea behind that is that

$$(u_\ell, v_\ell)_{A_\ell} = (u_\ell, v_\ell)_{A_{\ell+1}} \quad \text{and} \quad (u_\ell, v_\ell)_{Q_\ell} = (u_\ell, v_\ell)_{Q_{\ell+1}} \tag{12}$$

hold, i.e., we obtain a multigrid solver with conforming coarse-grid correction. This means that – on the coarse grid levels – the discretization is over penalized by a factor of $2^{L-\ell}$, i.e.,

$$\begin{aligned}
\underbrace{\frac{\sigma p^2}{h_L}}_{\tilde{\Sigma}_\ell} &= 2^{L-\ell} \underbrace{\frac{\sigma p^2}{h_\ell}}_{\Sigma_\ell}, \\
\tilde{\Sigma}_\ell &:= \Sigma_\ell :=
\end{aligned}$$

where Σ_ℓ is the canonical parameter and $\tilde{\Sigma}_\ell$ is the chosen one. We will see that this does not cause any problems for the examples we consider; following [13], convergence theory only holds if the number of smoothing steps is sufficiently increased for the coarser grid levels, cf. Remark 1.

Using a basis for the space V_ℓ^n , we obtain a standard matrix-vector problem: Find $\underline{u}_\ell \in \mathbb{R}^{N_\ell}$ such that

$$A_\ell \underline{u}_\ell = \underline{f}_\ell. \tag{13}$$

4. Multigrid solvers with Gauss-Seidel smoothers

In this and the following sections, we discuss several possible choices of multigrid smoothers, illustrate their convergence behavior with numerical experiments, and comment on the convergence theory.

We consider conforming discretizations and non-conforming discretizations which are set up as discussed in the last two sections. As we have nested spaces in all cases, the matrix $I_{\ell-1}^\ell$ is always the canonical embedding from $V_{\ell-1}$ into V_ℓ and the restriction matrix $I_\ell^{\ell-1}$ is its transpose. The chosen method is presented as pseudo-code as Algorithm 1, where we choose $\mu = 1$ for the V-cycle method or $\mu = 2$ for the W-cycle method.

Algorithm 1 Multigrid algorithm

```

MULTIGRID( $\ell, \underline{f}_\ell, \underline{u}_\ell$ )
  // Pre-Smoothing
  for  $n = 1, \dots, \nu_\ell$ 
     $\underline{u}_\ell \leftarrow \underline{u}_\ell + L_\ell^{-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$ 
  // Coarse-grid correction
  if  $\ell = 1$ 
     $\underline{u}_\ell \leftarrow \underline{u}_\ell + I_{\ell-1}^\ell A_{\ell-1}^{-1} I_\ell^{\ell-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$  // Direct solver
  else
    for  $n = 1, \dots, \mu$ 
       $\underline{u}_\ell \leftarrow \underline{u}_\ell + I_{\ell-1}^\ell \text{MULTIGRID}(\ell - 1, I_\ell^{\ell-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell), 0)$ 
  // Post-Smoothing
  for  $n = 1, \dots, \nu_\ell$ 
     $\underline{u}_\ell \leftarrow \underline{u}_\ell + L_\ell^{-\top} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$ 
  return  $\underline{u}_\ell$ 

```

In the finite element world, Gauss-Seidel smoothers are known to be very efficient smoothers; thus, as a first attempt, we consider such a smoother. One forward Gauss-Seidel sweep can be represented by

$$\underline{u}_\ell \leftarrow \underline{u}_\ell + L_\ell^{-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell),$$

where L_ℓ is a lower-triangular matrix containing the coefficients of the stiffness matrix A_ℓ , i.e., it is given by

$$(L_\ell)_{i,j} = \begin{cases} (A_\ell)_{i,j} & \text{if } i \geq j \\ 0 & \text{if } i < j \end{cases}.$$

To be able to use our multigrid solver as preconditioner for a conjugate gradient solver, the post-smoothing procedure uses the transposed matrix L_ℓ^\top , which represents a backward-Gauss-Seidel sweep.

All tables show the number of iterations required until the stopping criterion

$$\frac{\|A_L \underline{u}_L - \underline{f}_L\|_{\ell^2}}{\|\underline{f}_L\|_{\ell^2}} \leq \epsilon := 10^{-8}$$

is satisfied.

As usual, the convergence behavior of the overall solver can be improved if the multigrid method is not just used directly as a solver, but as a preconditioner within a preconditioned conjugate gradient (PCG) solver. Thus, we present results for both possibilities; in the following sections we will restrict ourselves to the more efficient PCG solver. Since the V-cycle and the W-cycle methods yield comparable iteration counts, we present the results for the more efficient V-cycle only. The number of smoothing steps is chosen as $\nu_\ell := 1$ in all cases.

The multigrid solver was implemented in C++ based on the G+Smo library [22]. The tables shown in the remainder of this section are obtained with the following command line instructions, where the values L and p are substituted accordingly.

```
> git clone https://github.com/gismo/gismo.git
> cd gismo
> make
> cd build/bin
> ./multiGrid_example -g domain2d/lldomain.xml -r L -p p
    -s gs -i d // Table 1 (a)
> ./multiGrid_example -g domain2d/yeti_mp2.xml -r L -p p
    -s gs -i d // Table 2 (a)
```

The results for the PCG experiments, presented in Tables 1 (b) and 2 (b), are obtained by replacing the option `-i d` by the option `-i cg`.

	(a) Direct – Conforming							(b) PCG – Conforming						
$L \setminus p$	2	3	4	5	6	7	8	2	3	4	5	6	7	8
4	9	24	74	–	–	–	–	8	15	28	53	–	–	–
5	9	24	73	–	–	–	–	8	15	28	52	–	–	–
6	9	24	72	–	–	–	–	8	15	28	53	–	–	–
7	10	24	72	–	–	–	–	8	15	28	54	–	–	–
8	10	24	72	–	–	–	–	8	15	28	54	–	–	–

Table 1: V-cycle with Gauss-Seidel smoother for the L-shaped domain

In Table 1, we observe that the multigrid solver is certainly robust in the grid size. While this approach is very efficient for small numbers of spline degrees, we observe that the convergence rates deteriorate significantly if the spline degree is increased. On the right side of the table, one can see the iteration counts for a preconditioned conjugate gradient method where one V-cycle of the mentioned

	(a) Direct – Conforming							(b) PCG – Conforming						
$L \setminus p$	2	3	4	5	6	7	8	2	3	4	5	6	7	8
3	13	25	75	–	–	–	–	10	15	28	54	–	–	–
4	14	25	74	–	–	–	–	10	15	28	53	–	–	–
5	15	25	74	–	–	–	–	11	16	28	54	–	–	–
6	15	25	72	–	–	–	–	11	16	29	54	–	–	–
7	17	25	73	–	–	–	–	12	16	29	55	–	–	–

Table 2: V-cycle with Gauss-Seidel smoother for the Yeti footprint

multigrid method is used as a preconditioner. We observe that the iteration counts are significantly smaller than the iteration counts obtained by directly applying the multigrid solver. However, we simultaneously observe that we do not observe any qualitative improvement. In Table 2, we give the iteration counts for the Yeti footprint. We observe that – despite the fact that the geometry function is now non-trivial – the iteration counts are very similar to those of the L-shaped domain.

When one turns to the non-conforming discretizations, it immediately turns out that the multigrid solver utilizing the Gauss-Seidel smoother does not converge well at all.

One can show using standard arguments that the multigrid method converges with rates that are independent of the grid size and of the number of patches. The convergence analysis (for the conforming case) employs estimates that increase exponentially in the spline degree, cf. [12]. The numerical experiments show that this is not only a matter of the proof.

Since we did not obtain convincing results, we are interested in more advanced smoothers that work well also for SIPG discretizations and which do not deteriorate if the spline degree is increased.

5. Multigrid with subspace corrected mass smoother

In this section, we employ the subspace corrected mass smoother as introduced in [16]. That smoother requires that the spline space has a tensor-product structure; in our examples, we have such a structure on each patch but not on the overall domain. The extension of that smoother to conforming discretizations has been discussed in [29] based on a domain-decomposition approach. The key idea was to decompose all degrees of freedom on a per-piece bases. Pieces are the patch-interiors and the interface pieces. In two dimensions, the interface pieces are the edges and the vertices of each of the patches. In three dimensions, the interface pieces are the faces, the edges and the vertices of each of the patches. The decomposition of the degrees of freedom is depicted in Figure 4 (left).

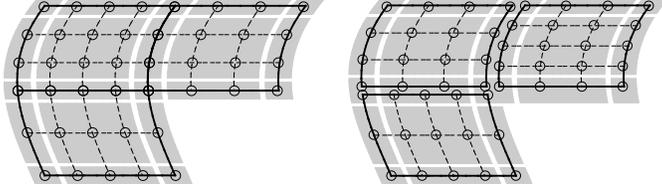


Figure 4: Decomposition of the degrees of freedom (represented by the Greville point)

The piece-local smoothers, which we denote by $L_{\ell,T}^{-1}$, are defined as follows. For T being a patch-interior, we choose $L_{\ell,T}^{-1}$ to be the subspace corrected mass smoother as proposed in [16]. We choose the scaling parameter (which was called σ in [16]) to be $\delta^{-1}h_\ell^{-2}$ for some suitable chosen parameter $\delta > 0$. If T is an interface piece, we choose

$$L_{\ell,T} := P_T^\top A_\ell P_T,$$

where the matrix $P_{\ell,T}$ represents the embedding of the piece T in the whole space. The symbol $L_{\ell,T}^{-1}$ refers to be the application of a direct solver. Applying a direct solver on the interfaces is feasible since the interfaces have much smaller numbers of degrees of freedom than the interiors of the patches. The overall smoother is just an additive composition of the piece-local smoothers, i.e., we choose

$$L_\ell^{-1} := \tau \sum_T P_{\ell,T} L_{\ell,T}^{-1} P_{\ell,T}^\top, \quad (14)$$

where the sum is taken over all pieces T . Here, $\tau > 0$ is some damping parameter to be chosen.

The convergence theory from [29] can be summarized by the following theorem.

Theorem 3. *Assume that $\Omega \subset \mathbb{R}^2$ is such that full elliptic regularity is satisfied (cf. [29, Assumption 3.1]). Consider the conforming discretization and a multigrid solver with the smoother (14). There are constants τ^* , δ^* and θ which are independent of K , h , L and p (but may depend particularly on the geometry functions and the maximum number of neighbors of a patch) such that for*

$$\tau \in (0, \tau^*), \quad \delta \in (0, \delta^*) \quad \text{and} \quad \nu_\ell > \nu_\ell^* := p \frac{\tau^*}{\tau} \frac{\delta^*}{\delta} \theta, \quad (15)$$

the W -cycle multigrid method converges with a convergence rate $q \leq \max_\ell \nu_\ell^ / \nu_\ell$.*

Note that the terms $\frac{\tau^*}{\tau}$ and $\frac{\delta^*}{\delta}$ imply that the convergence degrades if too small values of those parameters are chosen. Thus, it is of interest to choose these parameters in a rather optimal way.

Note that the convergence theorem requires full elliptic regularity (cf. [29, Assumption 3.1]). Thus, it is applicable to the Yeti footprint but it is not directly applicable to the L-shaped domain. Convergence results for the case with full

elliptic regularity often carry over *in practice* to cases where that regularity assumption does not hold. The same behavior can be observed for the numerical experiments we have considered.

Observe that the convergence theorem suggests that the number of smoothing steps should increase with p . As already outlined in [29], this seems to be too pessimistic since the numerical experiments have shown that $\nu_\ell := 1$ in all cases yields good convergence rates.

The numerical experiments are again applied within the same setup as in the last section. We set up a V-cycle multigrid method with 1+1 smoothing steps of the proposed smoother L_ℓ (on all grid levels). The damping parameter τ is chosen as indicated with the option `--MG.Damping` and the scaling parameter δ is chosen as indicated with the option `--MG.Scaling` below. The tables for the conforming case shown in this section are obtained with the following code, where the values L and p are substituted accordingly:

```
> ./multiGrid_example -g domain2d/lldomain.xml -r L -p p
    -s scms --MG.Damping 1 --MG.Scaling .12 -i d // Table 3 (a)
> ./multiGrid_example -g domain2d/yeti_mp2.xml -r L -p p
    -s scms --MG.Damping .25 --MG.Scaling .2 -i cg // Table 5 (a)
```

The results for the PCG experiments presented Table 3 (b) are obtained by replacing the option `-i d` by the option `-i cg`.

	(a) Direct – Conforming							(b) PCG – Conforming						
$L \setminus p$	2	3	4	5	6	7	8	2	3	4	5	6	7	8
4	27	23	22	20	17	16	15	16	14	13	12	11	11	10
5	29	27	27	26	24	24	22	17	16	15	15	14	14	14
6	30	30	28	27	27	27	26	17	17	16	16	15	16	15
7	31	30	29	28	28	27	27	17	17	17	16	16	16	16
8	32	31	30	29	28	28	28	18	17	17	17	16	16	16

Table 3: V-cycle with subspace corrected mass smoother for the L-shaped domain

All numerical experiments show that the proposed method is robust both in the grid size and the spline degree. However, when comparing the results for the Yeti footprint from Table 5 (a) with the corresponding results for the L-shaped domain from Table 3 (b), we see that the multigrid solver suffers from distorted geometry functions.

The numbers for the Yeti footprint seem not to be completely robust in the grid size. Since we have given a convergence theory, we know that the convergence numbers are bounded uniformly. Thus, the observed behavior is pre-asymptotic. The reason for this is that on coarser grid levels, the geometry is not resolved exactly. Let A_ℓ be the original stiffness matrix and \hat{A}_ℓ be the simplified stiffness

matrix obtained by neglecting the geometry function. Then, we have

$$\kappa(\widehat{A}_\ell^{-1}A_\ell) = \sup_{v_\ell \in V_\ell} \frac{|v_\ell|_{H^1(\Omega)}^2}{\sum_{k=1}^K |v_\ell \circ G_k|_{H^1(\widehat{\Omega})}^2} \sup_{v_\ell \in V_\ell} \frac{\sum_{k=1}^K |v_\ell \circ G_k|_{H^1(\widehat{\Omega})}^2}{|v_\ell|_{H^1(\Omega)}^2},$$

which yields

$$\begin{aligned} \kappa(\widehat{A}_0^{-1}A_0) &\leq \dots \leq \kappa(\widehat{A}_{L-1}^{-1}A_{L-1}) \leq \kappa(\widehat{A}_L^{-1}A_L) \\ &\leq \sup_{v \in V} \frac{|v|_{H^1(\Omega)}^2}{\sum_{k=1}^K |v \circ G_k|_{H^1(\widehat{\Omega})}^2} \sup_{v \in V} \frac{\sum_{k=1}^K |v \circ G_k|_{H^1(\widehat{\Omega})}^2}{|v|_{H^1(\Omega)}^2}, \end{aligned} \quad (16)$$

which can be finally bounded by a constant times some power of the quantity $\|\nabla G\|_{L^\infty(\widehat{\Omega})} \|(\nabla G)^{-1}\|_{L^\infty(\widehat{\Omega})}$. Of none of these relations is satisfied by equality. In such a case, the iteration counts are likely to increase if the grid gets refined. For more on this topic, see [27, Section 7.4].

As a next step, we turn towards the non-conforming discretizations. Here, each degree of freedom is assigned to exactly one patch. So, it would be tempting to set up a patch-wise splitting of the degrees of freedom. Unfortunately, numerical experiments have shown that this approach does not work well. So, we follow the approach from [29] also in the non-conforming case and split the degrees of freedom again into pieces T . This means that we avoid breaking the coupling which was enforced by the penalty term. So, the degrees of freedom belonging to one edge (face, vertex) are considered to be one piece, even if the degrees of freedom belong to different patches, see Figure 4 (right).

For this choice, we can give the following convergence theorem.

Theorem 4. *Assume that $\Omega \subset \mathbb{R}^2$ is such that full elliptic regularity holds (cf. [30, Assumption 4]) and assume that the geometry functions (but not necessarily the discretizations) agree on the interfaces (cf. [30, Assumption 2]). Consider the SIPG discretization and a multigrid solver with the smoother (14). There are constants τ^* , δ^* and θ which are independent of K , h , L and p (but may depend particularly on the geometry functions and the maximum number of neighbors of a patch) such that for*

$$\tau \in (0, \tau^*), \quad \delta \in (0, \delta^*) \quad \text{and} \quad \nu_\ell > \nu_\ell^* := 2^{L-\ell} (1+L-\ell)^2 p (\log p)^4 \frac{\tau^*}{\tau} \frac{\delta^*}{\delta} \theta, \quad (17)$$

the W -cycle multigrid method converges with a convergence rate $q \leq \max_\ell \nu_\ell^*/\nu_\ell$.

We give the proof of this theorem in the Appendix; the proof is based on the error estimates from [30].

One might observe that the number of smoothing steps required by this convergence theorem increases like $(1+L-\ell)2^{L-\ell}$. This follows the approach suggested in [13] and is related to the chosen over-penalization discussed in Section 3.

Remark 1. Note that the number of degrees of freedom on the coarser grid levels is smaller by a factor of $2^{d(L-\ell)}$. So, also using these additional smoothing steps, the overall complexity of the multigrid solver is still linear in the number of unknowns on the finest grid level if (a) $d \geq 3$ or (b) the V-cycle is considered. If we consider $d = 2$ and the W-cycle, the choice (17) yields that the computational complexity grows like $N_L L^3$, where N_L is the number of unknowns on the finest grid level. In the numerical experiments, we did not observe that increasing the number of smoothing steps has been required. Analogously to the conforming case, also the stated dependence on p is too pessimistic; thus, we again choose $\nu_\ell := 1$ on all grid levels.

Now, we provide numerical experiments for the SIPG discretization. Theoretically, we could just use exactly the discretization that has been chosen for the conforming case. This, however, yields a (particularly uninteresting) special case since Assumption 2 holds. In this special case, we have $V_\ell^c \subset V_\ell^n$ and the SIPG formulation converges to the conforming discretization for $\sigma \rightarrow \infty$. Instead, we are interested in a discretization such that Assumption 2 does not hold: We modify the setup of the spaces. For one third of the patches, we use the original spline space $S_{p,h_\ell}(\hat{\Omega})$. For one third of the patches, we use the spline space $S_{p+1,2h_\ell}(\hat{\Omega})$. For the last third of the patches, we use the spline space $S_{p,2h_\ell}(\hat{\Omega})$. This particular setting is obtained with the command line option `--NonMatching`. In this way, we obtain a setup where a conforming discretization is not possible.

The tables for the non-conforming case shown in this section are obtained with the following code, where the values L and p are substituted accordingly:

```
> ./multiGrid_example -g domain2d/ldomain.xml -r L -p p --DG
  --NonMatching -s scms --MG.Damping .9 --MG.Scaling .12
  -i d // Table 4 (a)
> ./multiGrid_example -g domain2d/yeti_mp2.xml -r L -p p --DG
  --NonMatching -s scms --MG.Damping .25 --MG.Scaling .2
  -i cg // Table 5 (b)
```

The results for the PCG experiments presented in Table 4 (b) are obtained by replacing the option `-i d` by the option `-i cg`.

The PCG discretizations are presented in Tables 4 (b) and 5 (b); we again obtain robustness in the grid size and the spline degree. Here, for the Yeti footprint, we have again iteration counts that are slightly increasing with the grid size; again, this observation can be explained by the fact that finer grids allow to resolve the geometry function better, cf. (16).

In principle, the method works also if the multigrid solver is applied directly, cf. Table 4 (a). Here, we suffer from numerical instabilities which are amplified with an increasing number of levels. One can avoid these instabilities, e.g., by increasing the number of pre- and post smoothing steps. However, using the

	(a) Direct – Non-conforming							(b) PCG – Non-conforming						
$L \setminus p$	2	3	4	5	6	7	8	2	3	4	5	6	7	8
4	22	28	34	33	23	35	23	17	16	15	14	13	12	12
5	71	48	45	69	35	32	57	19	19	18	17	17	17	16
6	73	71	70	46	69	57	145	21	20	20	19	19	20	19
7	100	106	86	71	92	67	61	22	22	21	21	21	22	22
8	90	94	127	98	291	106	73	23	23	22	22	22	22	23

Table 4: V-cycle with subspace corrected mass smoother for the L-shaped domain

	(a) PCG – Conforming							(b) PCG – Non-conforming						
$L \setminus p$	2	3	4	5	6	7	8	2	3	4	5	6	7	8
3	44	42	41	39	37	36	34	40	38	36	35	34	33	31
4	48	47	45	43	43	40	41	44	44	42	42	40	40	39
5	51	49	48	47	45	45	44	49	47	47	46	46	45	44
6	52	51	49	48	47	46	45	58	57	57	56	55	54	53
7	54	53	51	50	49	48	47	74	73	72	71	71	72	70

Table 5: V-cycle with subspace corrected mass smoother for the Yeti footprint

multigrid method as a preconditioner within a PCG solver is obviously the more efficient approach.

6. Multigrid with hybrid smoother

We have observed that a multigrid method with the subspace corrected mass smoother is robust in the grid size and the spline degree and works well for both conforming and discontinuous Galerkin discretizations. We have also observed that this approach suffers from non-simple geometry functions since it is based on the close connection between the stiffness matrix A_ℓ and the simplified stiffness matrix \hat{A}_ℓ . The results for the Gauss-Seidel smoother are different: the multigrid solver works badly both for large spline degrees and for discontinuous Galerkin discretizations. However, by comparing Table 1 with Table 2, we observe that the method behaves quite robust in the geometry function.

Since the behavior of the two smoothers is somewhat orthogonal, we can hope for a good method if we combine them. Our idea is to use one forward Gauss-Seidel sweep followed by the subspace corrected mass smoother for pre-smoothing and the subspace corrected mass smoother followed by one backward Gauss-Seidel sweep for post-smoothing. The overall method is presented as Algorithm 2.

Algorithm 2 Multigrid algorithm with hybrid smoother

```

MULTIGRID( $\ell, \underline{f}_\ell, \underline{u}_\ell$ )
  // Pre-Smoothing (forward Gauss-Seidel)
   $\underline{u}_\ell \leftarrow \underline{u}_\ell + (L_\ell^{GS})^{-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$ 
  // Pre-Smoothing (subspace corrected mass smoother)
  for  $n = 1, \dots, \nu_\ell$ 
     $\underline{u}_\ell \leftarrow \underline{u}_\ell + (L_\ell^{SCMS})^{-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$ 
  // Coarse-grid correction
  if  $\ell = 1$ 
     $\underline{u}_\ell \leftarrow \underline{u}_\ell + I_{\ell-1}^\ell A_{\ell-1}^{-1} I_\ell^{\ell-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$  // Direct solver
  else
    for  $n = 1, \dots, \mu$ 
       $\underline{u}_\ell \leftarrow \underline{u}_\ell + I_{\ell-1}^\ell \text{MULTIGRID}(\ell - 1, I_\ell^{\ell-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell), 0)$ 
  // Post-Smoothing (subspace corrected mass smoother)
  for  $n = 1, \dots, \nu_\ell$ 
     $\underline{u}_\ell \leftarrow \underline{u}_\ell + (L_\ell^{SCMS})^{-1} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$ 
  // Post-Smoothing (backward Gauss-Seidel)
   $\underline{u}_\ell \leftarrow \underline{u}_\ell + (L_\ell^{GS})^{-\top} (\underline{f}_\ell - A_\ell \underline{u}_\ell)$ 
  return  $\underline{u}_\ell$ 

```

The convergence analysis from Section 5 can be easily carried over to the hybrid smoother. The iteration matrix for the (V or W cycle) multigrid method with the hybrid smoother is given by

$$\widetilde{W}_\ell := (I - (L_\ell^{GS})^{-\top} A_\ell) W_\ell (I - (L_\ell^{GS})^{-1} A_\ell),$$

where W_ℓ is the iteration matrix of the (V or W cycle, respectively) multigrid method with the subspace corrected mass smoother. Since the Gauss-Seidel iteration is stable in the energy norm, we obtain

$$\|\widetilde{W}_\ell\|_{A_\ell} \leq \|I - (L_\ell^{GS})^{-\top} A_\ell\|_{A_\ell} \|W_\ell\|_{A_\ell} \|I - (L_\ell^{GS})^{-1} A_\ell\|_{A_\ell} \leq \|W_\ell\|_{A_\ell}.$$

So, we have using the results from the last section the convergence of the W-cycle multigrid method with hybrid smoother. Thus, we obtain as follows.

Corollary 5. *Consider the multigrid solver with the hybrid smoother. Under the assumptions of Theorem 3 or 4, respectively, the W-cycle multigrid method converges with a convergence rate $q \leq \max_\ell \nu_\ell^* / \nu_\ell$.*

The tables for the experiments with the hybrid smoother are obtained with the following code, where the values L and p are substituted accordingly:

	(a) Direct – Conforming							(b) PCG – Conforming						
$L \setminus p$	2	3	4	5	6	7	8	2	3	4	5	6	7	8
3	12	15	22	29	35	40	47	9	11	14	17	20	21	22
4	13	16	23	32	38	45	50	10	11	15	19	21	23	25
5	14	16	24	35	41	47	53	11	11	16	19	22	24	26
6	15	16	26	37	45	52	55	11	12	16	20	23	26	27
7	16	17	26	38	47	54	57	12	12	17	21	24	26	28

Table 6: V-cycle with hybrid smoothing strategy for the Yeti footprint

	(a) Direct – Non-conforming							(b) PCG – Non-conforming						
$L \setminus p$	2	3	4	5	6	7	8	2	3	4	5	6	7	8
3	23	18	35	34	37	24	28	15	17	18	19	20	20	21
4	19	23	31	42	32	59	32	16	19	21	22	24	26	26
5	20	25	34	50	46	67	62	17	20	22	25	27	29	31
6	22	27	30	47	48	52	69	17	20	23	25	27	30	32
7	22	29	33	40	58	52	65	17	21	23	26	29	31	33

Table 7: V-cycle with hybrid smoothing strategy for the Yeti footprint

```

> ./multiGrid_example -g domain2d/yeti_mp2.xml -r L -p p
  -s hyb --MG.Damping .25 --MG.Scaling .1 -i d // Table 6 (a)
> ./multiGrid_example -g domain2d/yeti_mp2.xml -r L -p p --DG
  --NonMatching -s hyb --MG.Damping .25 --MG.Scaling .1
  -i d // Table 7 (a)

```

The results for the PCG experiments, presented in Tables 6 (b) and 7 (b), are obtained by replacing the option `-i d` by the option `-i cg`.

For both cases, we obtain that the iteration counts are quite robust in the grid size (even if the maximum number of iterations is not reached for the coarser grid levels). We observe that the number of iterations increases with the spline degree. This is indeed due to the fact that for small values of p , the Gauss-Seidel smoother yields very fast convergence and that convergence behavior is carried over to the hybrid smoother. For larger spline degrees, the hybrid smoother’s convergence behavior degrades mildly; this is due to the fact that the Gauss-Seidel smoother is not completely capable to capture all effects perfectly. Still, keeping in mind that the condition number of the stiffness matrix grows exponentially with the spline degree, the observed behavior is still very satisfactory.

Compared to applying the subspace corrected mass smoother only, the hybrid smoother pays of in most cases. Certainly, applying the hybrid smoother with $\nu_\ell := 1$ means basically that 2 pre- and 2 post-smoothing steps are applied.

Since the Gauss-Seidel smoother is slightly cheaper than the subspace corrected mass smoother, the costs for one such cycle are smaller than the costs of two multigrid cycles with the subspace corrected mass smoother only.

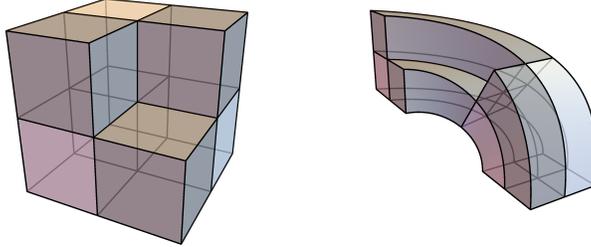


Figure 5: The 3D computational domains: Fichera corner and twisted Fichera corner

Besides the two-dimensional examples considered so far, the proposed methods can be directly extended to three dimensional problems (even if the details of the convergence theory have not been worked out for these cases). We consider the two domains depicted in Figure 5: the Fichera corner and a variant of that domain with non-trivial geometry function, which we call twisted Fichera corner.

The tables for the three dimensional domains are obtained with the following code, where the values L and p are substituted accordingly:

```
> ./multiGrid_example -g domain2d/fichera.xml -r L -p p
  -s hyb --MG.Scaling .12 --MG.Damping 1 -i cg // Table 8 (a)
> ./multiGrid_example -g domain2d/twisted_fichera.xml -r L -p p
  -s hyb --MG.Scaling .12 --MG.Damping .25 -i cg // Table 9 (a)
```

The results for the DG experiments, presented in the Tables 8 (b) and 9 (b), are obtained by adding the command line options `--DG --NonMatching`.

	(a) PCG – Conforming					(b) PCG – Non-conforming				
$L \setminus p$	2	3	4	5	6	2	3	4	5	6
2	6	6	7	8	8	12	13	15	16	19
3	6	7	7	8	8	13	15	16	18	20
4	6	8	9	9	10	14	16	18	19	20
5	6	8	9	10	11	14	16	17	19	20

Table 8: V-cycle with hybrid smoothing strategy for the Fichera corner

Similar to the results for the Yeti footprint, Tables 8 and 9 again show small iteration counts. For the twisted Fichera corner, we observe that the number of iterations increases mildly when the grid gets refined; this is again to be

	(a) PCG – Conforming					(b) PCG – Non-conforming				
$L \setminus p$	2	3	4	5	6	2	3	4	5	6
2	10	13	14	15	17	22	25	28	30	31
3	13	15	18	20	22	29	31	33	36	38
4	14	17	19	22	25	31	34	37	40	42
5	16	17	20	23	26	31	36	40	44	47

Table 9: V-cycle with hybrid smoothing strategy for the twisted Fichera corner

explained by the better resolution of the geometry function. Moreover, we observe a very mild dependence on the spline degree.

7. Conclusions and outlook

We have presented robust multigrid solvers for multi-patch IgA with conforming and non-conforming discretizations. We have given convergence results that exactly state the robustness of the solvers in the grid size. Concerning the dependence on the spline degree, the statements seem to be too pessimistic since the solvers have been completely or (at least) rather robust in practice.

We have addressed another issue, which causes problems for all solvers that use the tensor-product structure on the parameter domain: the dependence on the geometry function. We have proposed a hybrid smoother between our subspace corrected mass smoother and the Gauss-Seidel smoother which seems to reduce the effect on the geometry function. Finding approaches to better incorporate the geometry function into the smoother itself seems to be an interesting topic for further research.

Appendix

In the appendix, we give a proof of Theorem 4 and of some auxiliary results.

Every constant c used within the appendix is assumed to be independent of the grid size, the grid level, the spline degree and the number of patches, but it may depend on the geometry function (cf. [30, Assumption 3]), the number of neighbors of a patch (cf. [29, Assumption 2.3]), the constant in the elliptic regularity assumption (cf. [29, Assumption 3.1]) and the quasi-uniformity of the grid, i.e., the ratio between the largest and the smallest knot span of the knot vectors on one level. We write $A \lesssim B$ if and only if there is a constant c such that $A \leq c B$ and we write $A \approx B$ if and only if $A \lesssim B$ and $B \lesssim A$.

First, we show the following lemma, which is basically a trace inequality.

Lemma 6. Let $S := S^{(1)} \otimes S^{(2)}$ be the space of tensor-product splines of degree p on a quasi-uniform grid with size h on the parameter domain $\widehat{\Omega} := (0, 1)^2$. Then, the estimate

$$|u(0)|^2 \lesssim \left(\log \left(1 + \frac{p^4}{h^2 \theta^2} \right) \right)^2 \left(|u|_{H^1(\widehat{\Omega})}^2 + \theta^2 \|u\|_{L_2(\widehat{\Omega})}^2 \right)$$

holds for all $u \in S$ and all $\theta \geq 1$.

PROOF. For $\nu = 1, 2$, let $(\psi_{\nu,i})_{i=1}^{N_\nu}$ be the eigenfunctions of $S^{(\nu)}$, i.e., such that

$$\begin{aligned} (\psi_{\nu,i}, \psi_{\nu,j})_{L_2(0,1)} &= \delta_{i,j} \quad \text{and} \\ (\psi'_{\nu,i}, \psi'_{\nu,j})_{L_2(0,1)} + \theta^2 (\psi_{\nu,i}, \psi_{\nu,j})_{L_2(0,1)} &= \lambda_{\nu,i} \delta_{i,j}, \end{aligned}$$

where $\delta_{i,j}$ is the Kronecker delta and $\lambda_{\nu,1} \leq \lambda_{\nu,2} \leq \dots \leq \lambda_{\nu,N_\nu}$ are the corresponding eigenvalues. Using coercivity of $(\cdot, \cdot)_{L_2(0,1)}$ and a standard inverse estimate, cf. [26, Corollary 3.94], we obtain

$$\theta^2 \leq \lambda_{\nu,1} \quad \text{and} \quad \lambda_{\nu,N_\nu} \lesssim p^4 h^{-2} + \theta^2.$$

We define level sets

$$I_{\nu,m} := \{i : \mu_{m-1} := 2^{m-1} \theta^2 \leq \lambda_{\nu,i} < \mu_m := 2^m \theta^2\}$$

for $m \in \{1, 2, 3, \dots, M\}$, where

$$M := 1 + \max_{\nu \in \{1,2\}} \lceil \log_2(\theta^{-2} \lambda_{\nu,N_\nu}) \rceil \lesssim \log(1 + p^4 h^{-2} \theta^{-2})$$

is the number of level sets. Note that by construction every eigenvalue belongs to exactly one level set. Every function $u \in S$ can be represented as

$$u(x, y) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} u_{i,j} \psi_{1,i}(x) \psi_{2,j}(y) = \sum_{m=1}^M \sum_{n=1}^M \underbrace{\sum_{i \in I_{1,m}} \sum_{j \in I_{2,n}} u_{i,j} \psi_{1,i}(x) \psi_{2,j}(y)}_{w_{m,n}(x, y) :=}$$

A standard trace estimate, cf. [29, Lemma 4.4], yields

$$\begin{aligned} |w_{m,n}(0)|^2 &\lesssim \|w_{m,n}\|_{L_2(\{0\} \times (0,1))} \|w_{m,n}\|_{H^1(\{0\} \times (0,1))} \\ &\lesssim \|w_{m,n}\|_{0,0,1}^{1/2} \|w_{m,n}\|_{1,0,1}^{1/2} \|w_{m,n}\|_{0,1,1}^{1/2} \|w_{m,n}\|_{1,1,1}^{1/2} \end{aligned} \quad (18)$$

where

$$\|w\|_{a,b,\eta}^2 := \left\| \frac{\partial^{a+b}}{\partial x^a \partial x^b} w \right\|_{L_2(\widehat{\Omega})}^2 + \eta^{2(a+b)} \|w\|_{L_2(\widehat{\Omega})}^2$$

for $a, b \in \mathbb{N}_0$. Since $\theta \geq 1$ and since all eigenvalues are in $I_{1,m}$ or $I_{2,n}$, respectively, we obtain

$$\|w\|_{a,b,1}^2 \leq \|w\|_{a,b,\theta}^2 \approx \mu_m^a \mu_n^b \|w_{m,n}\|_{L_2(\widehat{\Omega})}^2.$$

Using (18), we obtain further

$$\begin{aligned} |w_{m,n}(0)|^2 &\lesssim \mu_m \mu_n \|w_{m,n}\|_{L_2(\hat{\Omega})}^2 \approx \|w_{m,n}\|_{1,0,\theta} \|w_{m,n}\|_{0,1,\theta} \\ &\leq \left(|w_{m,n}|_{H^1(\hat{\Omega})}^2 + \theta^2 \|w_{m,n}\|_{L_2(\hat{\Omega})}^2 \right). \end{aligned}$$

Finally, the Cauchy-Schwarz inequality and orthogonality of the basis functions (both in L_2 and H^1) yield

$$\begin{aligned} |u(0)|^2 &\lesssim M^2 \sum_{m=1}^M \sum_{n=1}^M |w_{m,n}(0)|^2 \\ &\lesssim M^2 \sum_{m=1}^M \sum_{n=1}^M \left(|w_{m,n}|_{H^1(\hat{\Omega})} + \theta^2 \|w_{m,n}\|_{L_2(\hat{\Omega})} \right)^2 \\ &= M^2 \left(|u|_{H^1(\hat{\Omega})}^2 + \theta^2 \|u\|_{L_2(\hat{\Omega})}^2 \right), \end{aligned}$$

which finishes the proof. \square

Now, we give bounds on the smoother which allow to show the smoothing property.

Lemma 7. *Provided the assumptions of Theorem 4, the estimate*

$$A_\ell \leq L_\ell \lesssim p(\log p)^2 (1 + L - \ell)^2 2^{L-\ell} \frac{\tau^*}{\tau} \frac{\delta^*}{\delta} \tilde{L}_\ell$$

holds, where $\tilde{L}_\ell := Q_\ell + (1 + 2^{\ell-L} h_\ell^{-2}) M_\ell$ and M_ℓ is the standard mass matrix.

PROOF. The proof of this Lemma requires the notation from [29], i.e., we denote the set of all patch-interiors by \mathbb{K} , the set of all edges by \mathbb{E} and the set of all vertices by \mathbb{V} .

Observe that we have

$$A_\ell \approx Q_\ell \tag{19}$$

for all $\ell = 0, 2, \dots, L$, where Q_ℓ is defined by (11) and (12). For $\ell = L$, this statement directly follows from [30, Theorem 8]. Since [30, Theorem 8] also holds in cases of over-penalization, we can apply that theorem also to the case $\ell < L$ and obtain (19) also in that case.

As first step, we bound L_ℓ from below. (The following arguments are analogous to [29, Lemma 4.3].) The triangle inequality yields

$$A_\ell \lesssim \sum_{T \in \mathbb{K} \cup \mathbb{E} \cup \mathbb{V}} P_{\ell,T} (P_{\ell,T}^\top A_\ell P_{\ell,T}^\top) P_{\ell,T}^\top. \tag{20}$$

For $T \in \mathbb{K}$, [16, Lemma 8] and (19) yield $L_{\ell,T} \gtrsim P_{\ell,T}^\top Q_\ell P_{\ell,T}^\top \gtrsim P_{\ell,T}^\top A_\ell P_{\ell,T}^\top$. For $T \in \mathbb{E} \cup \mathbb{V}$, we have by definition $L_{\ell,T} = P_{\ell,T}^\top A_\ell P_{\ell,T}^\top$. Thus, we obtain from (20)

$$A_\ell \lesssim \sum_{T \in \mathbb{K} \cup \mathbb{E} \cup \mathbb{V}} P_{\ell,T} L_{\ell,T} P_{\ell,T}^\top$$

and for all $\tau \in (0, \tau^*)$ with τ^* small enough further

$$A_\ell \leq \tau^{-1} \sum_{T \in \mathbb{K} \cup \mathbb{E} \cup \mathbb{V}} P_{\ell,T} L_{\ell,T} P_{\ell,T}^\top = L_\ell,$$

which shows the first part of the desired inequality.

Now, we bound L_ℓ from above. We use the decomposition

$$Q_\ell = K_\ell + \frac{\sigma p^2}{h_L} J_\ell,$$

cf. (11). Using (19), we obtain

$$L_{\ell,T} = P_{\ell,T}^\top A_\ell P_{\ell,T} \approx P_{\ell,T}^\top Q_\ell P_{\ell,T} = \underbrace{P_{\ell,T}^\top K_\ell P_{\ell,T}}_{\tilde{K}_{\ell,T} :=} + \frac{\sigma p^2}{h_L} \underbrace{P_{\ell,T}^\top J_\ell P_{\ell,T}}_{\tilde{J}_{\ell,T} :=}$$

for all $T \in \mathbb{E} \cup \mathbb{V}$ and, therefore,

$$L_\ell \approx \tilde{K}_\ell + \frac{\sigma p^2}{h_L} \tilde{J}_\ell, \quad (21)$$

where

$$\tilde{K}_\ell := \tau^{-1} \sum_{T \in \mathbb{K}} P_{\ell,T} L_{\ell,T} P_{\ell,T}^\top + \tau^{-1} \sum_{T \in \mathbb{E} \cup \mathbb{V}} P_{\ell,T} K_{\ell,T} P_{\ell,T}^\top$$

and

$$\tilde{J}_\ell := \tau^{-1} \sum_{T \in \mathbb{E} \cup \mathbb{V}} P_{\ell,T} J_{\ell,T} P_{\ell,T}^\top.$$

Completely analogous to [29, Lemma 4.7], we obtain

$$\tilde{K}_\ell \lesssim p \frac{\tau^*}{\tau} \frac{\delta^*}{\delta} (K_\ell + h_\ell^{-2} M_\ell) \leq p \frac{\tau^*}{\tau} \frac{\delta^*}{\delta} (Q_\ell + h_\ell^{-2} M_\ell) \leq p 2^{L-\ell} \frac{\tau^*}{\tau} \frac{\delta^*}{\delta} \tilde{L}_\ell. \quad (22)$$

So, it remains to bound \tilde{J}_ℓ from above. Since the restriction of J_ℓ to any patch-interior vanishes, the same arguments as in the proof of [29, Lemma 4.7] and the triangle inequality yield

$$\begin{aligned} \sum_{T \in \mathbb{E}} \|P_T P_T^\top \underline{u}_\ell\|_{J_\ell}^2 &= \left\| \sum_{T \in \mathbb{E}} P_T P_T^\top \underline{u}_\ell \right\|_{J_\ell}^2 = \left\| \underline{u}_\ell - \sum_{T \in \mathbb{V}} P_T P_T^\top \underline{u}_\ell \right\|_{J_\ell}^2 \\ &\leq \|\underline{u}_\ell\|_{J_\ell}^2 + \sum_{T \in \mathbb{V}} \|P_T P_T^\top \underline{u}_\ell\|_{J_\ell}^2. \end{aligned}$$

and therefore

$$\sum_{T \in \mathbb{E} \cup \mathbb{V}} P_{\ell,T} \tilde{J}_{\ell,T} P_{\ell,T}^\top \lesssim J_\ell + \sum_{T \in \mathbb{V}} P_{\ell,T} \tilde{J}_{\ell,T} P_{\ell,T}^\top. \quad (23)$$

Note that J_ℓ models jumps and note that these jumps can be bounded from above using the triangle inequality with the function values on both sides. Thus, we obtain

$$\sum_{T \in \mathbb{V}} \|P_T P_T^\top \underline{u}_\ell\|_{J_\ell}^2 \lesssim \sum_{k=1}^K \sum_{T \in \mathbb{V}} (u_\ell|_{\Omega_k}|_T)^2 \|\psi\|_{L_2(0,1)}^2,$$

where $u_\ell|_{\Omega_k}$ is the restriction of u_ℓ to the patch Ω_k and $u_\ell|_{\Omega_k}|_T$ is the evaluation of the continuous extension of that function to the vertex T at that vertex and $\psi(x) = \max\{1 - x/h_\ell, 0\}^p$ is the corresponding basis function. Using $\|\psi\|_{L_2(0,1)}^2 \approx p^{-1}h_\ell$, cf. [29, Eq. (4.16)], Lemma 6 (with $\theta := (1 + h_\ell^{-2}2^{\ell-L})^{1/2}$), and $h_\ell \leq 1$, we further obtain

$$\begin{aligned} \sum_{T \in \mathbb{V}} \|P_T P_T^\top \underline{u}_\ell\|_{J_\ell}^2 &\lesssim \frac{h_\ell}{p} \left(\log \left(1 + \frac{p^4}{h_\ell^2(1 + h_\ell^{-2}2^{\ell-L})^2} \right) \right)^2 \\ &\quad \sum_{k=1}^K \left(|u_\ell \circ G_k|_{H^1(\hat{\Omega})}^2 + (1 + 2^{\ell-L}h_\ell^{-2}) \|u_\ell \circ G_k\|_{L_2(\hat{\Omega})}^2 \right) \\ &\lesssim \frac{h_\ell}{p} (\log p)(1 + L - \ell)^2 \sum_{k=1}^K \left(|u_\ell \circ G_k|_{H^1(\hat{\Omega})}^2 + (1 + 2^{\ell-L}h_\ell^{-2}) \|u_\ell \circ G_k\|_{L_2(\hat{\Omega})}^2 \right). \end{aligned}$$

Using [30, Lemma 6], we obtain

$$\sum_{T \in \mathbb{V}} \|P_T P_T^\top \underline{u}_\ell\|_{J_\ell}^2 \lesssim \frac{h_\ell}{p} (\log p)^2 (1 + L - \ell)^2 \left(|u_\ell|_{H^1(\Omega)}^2 + (1 + 2^{\ell-L}h_\ell^{-2}) \|u_\ell\|_{L_2(\Omega)}^2 \right)$$

and thus

$$\sum_{T \in \mathbb{V}} P_{\ell,T} \tilde{J}_{\ell,T} P_{\ell,T}^\top \lesssim \frac{h_\ell}{p} (\log p)^2 (1 + L - \ell)^2 (K_\ell + (1 + 2^{\ell-L}h_\ell^{-2}) M_\ell).$$

This shows together with (23) and $h_L \approx 2^{\ell-L}h_\ell$

$$\begin{aligned} \frac{\sigma p^2}{h_L} \tilde{J}_\ell &= \tau^{-1} \frac{\sigma p^2}{h_L} \sum_{T \in \mathbb{E} \cup \mathbb{V}} P_{\ell,T} \tilde{J}_{\ell,T} P_{\ell,T}^\top \\ &\lesssim \tau^{-1} \left(\frac{\sigma p^2}{h_L} J_\ell + p(\log p)^2 (1 + L - \ell)^2 2^{L-\ell} (K_\ell + (1 + 2^{\ell-L}h_\ell^{-2}) M_\ell) \right) \\ &\lesssim \tau^{-1} p(\log p)^2 (1 + L - \ell)^2 2^{L-\ell} \tilde{L}_\ell. \end{aligned}$$

Since $\delta \in (0, \delta^*)$ and since $\tau^* \approx 1$, we obtain

$$\frac{\sigma p^2}{h_L} \tilde{J}_\ell \lesssim p(\log p)^2 (1 + L - \ell)^2 2^{L-\ell} \frac{\tau^* \delta^*}{\tau \delta} \tilde{L}_\ell,$$

which finishes together with (21) and (22) the proof. \square

Lemma 8. Let $\|v\|_{Q_\ell^+}^2 := |v|_{H^1(\Omega)}^2 + \sigma^{-2}p^{-4}4^{\ell-L}h_\ell^2|v|_{H^2(\Omega)}^2$. The estimate

$$\inf_{v_\ell \in V_\ell^n} \|u - v_\ell\|_{Q_\ell^+}^2 \lesssim |u|_{H^2(\Omega)}^2$$

holds for all $u \in H^2(\Omega)$.

PROOF. Let

$$W := \{u \in H^1(\Omega) : u \circ G_k \in S_{1,1}(\widehat{\Omega}) \text{ for all } k = 1, \dots, K\},$$

be the set of all globally continuous functions which are linear locally. Observe that $W \subseteq V_\ell^n$. Using u and w being continuous, we obtain

$$\|u - w\|_{Q_\ell^+}^2 = |u - w|_{H^1(\Omega)}^2 + \frac{h_\ell^2}{\sigma^2 p^4 4^{L-\ell}} |u - w|_{H^2(\Omega)}^2.$$

For the choice $w \in H^1(\Omega)$ with $w|_{\Omega_k} := w_k = \widehat{w}_k \circ G_k^{-1}$, where

$$\widehat{w}_k(x, y) := \sum_{i=0}^1 \sum_{j=0}^1 \widehat{\phi}_i(x) \phi_j(y) \widehat{u}_k(i, j) \quad \text{and} \quad \widehat{\phi}_0(t) := 1 - t \quad \text{and} \quad \widehat{\phi}_1(t) := t,$$

we further obtain using standard approximation error estimates and [30, Lemma 6]

$$\inf_{v_\ell \in V_\ell^n} \|u - v_\ell\|_{Q_\ell^+}^2 \leq \|u - w\|_{Q_\ell^+}^2 \lesssim \left(1 + \frac{h_\ell^2}{\sigma^2 p^4 4^{L-\ell}}\right) |u|_{H^2(\Omega)}^2.$$

Using $h_\ell \leq 1$, $\sigma \geq 1$, $p \geq 2$, and $L \geq \ell$, we obtain the desired result.

Lemma 9. Provided the assumptions of Theorem 4, the estimate

$$\|(I - I_{\ell-1}^\ell A_{\ell-1}^{-1} I_\ell^{\ell-1} A_\ell) A_\ell^{-1} \widetilde{L}_\ell\|_{\widetilde{L}_\ell} \lesssim (\log p)^2$$

holds, where \widetilde{L}_ℓ is as in Lemma 7.

PROOF. Let $u_\ell \in V_\ell^n$ be arbitrary but fixed. Let $f_\ell \in V_\ell^n$ be such that

$$(u_\ell, v_\ell)_{A_\ell} = (f_\ell, v_\ell)_{L_2(\Omega)} \quad \text{for all } v_\ell \in V_\ell^n.$$

Let $u_{\ell-1} \in V_{\ell-1}$ and $u \in V$ be such that

$$\begin{aligned} (u_{\ell-1}, v_{\ell-1})_{A_\ell} &= (f_\ell, v_{\ell-1})_{L_2(\Omega)} \quad \text{for all } v_{\ell-1} \in V_{\ell-1}, \\ (\nabla u, \nabla v)_{L_2(\Omega)} &= (f_\ell, v)_{L_2(\Omega)} \quad \text{for all } v \in V. \end{aligned}$$

Using $f_\ell \in L_2(\Omega)$ and full elliptic regularity, cf. [29, Assumption 3.1], we obtain $u \in H^2(\Omega)$ and

$$|u|_{H^2(\Omega)} \lesssim \|f_\ell\|_{L_2(\Omega)} = \sup_{v_\ell \in V_\ell^n} \frac{(f_\ell, v_\ell)_{L_2(\Omega)}}{\|v_\ell\|_{L_2(\Omega)}} = \sup_{v_\ell \in V_\ell^n} \frac{(u_\ell, v_\ell)_{A_\ell}}{\|v_\ell\|_{L_2(\Omega)}} = \|u_\ell\|_{A_\ell M_\ell^{-1} A_\ell}. \quad (24)$$

[30, Theorems 12 and 13] and Lemma 8 yield

$$\begin{aligned}\|u - u_\ell\|_{Q_\ell}^2 &\lesssim \min\{1, (\log \sigma_\ell)^2 \sigma_\ell^{1/(2p-1)} h_\ell^2\} |u|_{H^2(\Omega)}^2, \\ \|u - u_{\ell-1}\|_{Q_\ell}^2 &\lesssim \min\{1, (\log \sigma_{\ell-1})^2 \sigma_{\ell-1}^{1/(2p-1)} h_{\ell-1}^2\} |u|_{H^2(\Omega)}^2,\end{aligned}$$

where $\sigma_\ell = 2^{L-\ell} p^2 \sigma$. Using the triangle inequality, $\sigma \approx 1$, $\log(ab) \lesssim \log a \log b$, $h_\ell \approx h_{\ell-1}$, $p \geq 2$, and we obtain further

$$\begin{aligned}\|u_\ell - u_{\ell-1}\|_{Q_\ell}^2 &\lesssim \min\{1, (\log \sigma_\ell)^2 \sigma_\ell^{1/(2p-1)} h_\ell^2\} |u|_{H^2(\Omega)}^2 \\ &\lesssim \min\{1, (\log p)^2 p^{2/(2p-1)} (1+L-\ell)^2 2^{(L-\ell)(1/(2p-1))} h_\ell^2\} |u|_{H^2(\Omega)}^2 \\ &\lesssim \min\{1, (\log p)^2 2^{L-\ell} h_\ell^2\} |u|_{H^2(\Omega)}^2 \lesssim (\log p)^2 (1+2^{\ell-L} h_\ell^{-2})^{-1} |u|_{H^2(\Omega)}^2\end{aligned}$$

Using (19), (24) and the definition of $u_{\ell-1}$, we obtain further

$$\|(I - I_{\ell-1}^\ell A_{\ell-1}^{-1} I_\ell^{\ell-1} A_\ell) \underline{u}_\ell\|_{A_\ell}^2 \lesssim (\log p)^2 (1+2^{\ell-L} h_\ell^{-2})^{-1} \|\underline{u}_\ell\|_{A_\ell M_\ell^{-1} A_\ell}^2.$$

This yields

$$\|A_\ell^{1/2} (I - I_{\ell-1}^\ell A_{\ell-1}^{-1} I_\ell^{\ell-1} A_\ell) A_\ell^{-1} M_\ell^{1/2}\|^2 \lesssim (\log p)^2 (1+2^{\ell-L} h_\ell^{-2})^{-1}$$

and thus

$$\|(I - I_{\ell-1}^\ell A_{\ell-1}^{-1} I_\ell^{\ell-1} A_\ell) \underline{u}_\ell\|_{(1+2^{\ell-L} h_\ell^{-2}) M_\ell}^2 \lesssim (\log p)^2 \|\underline{u}_\ell\|_{A_\ell}^2 \quad \text{for all } \underline{u}_\ell \in \mathbb{R}^{N_\ell}.$$

Using this estimate and the stability of the A_ℓ -orthogonal projection, we obtain

$$\|(I - I_{\ell-1}^\ell A_{\ell-1}^{-1} I_\ell^{\ell-1} A_\ell) \underline{u}_\ell\|_{\tilde{L}_\ell}^2 \lesssim (\log p)^2 \|\underline{u}_\ell\|_{A_\ell}^2 \quad \text{for all } \underline{u}_\ell \in \mathbb{R}^{N_\ell}$$

and further

$$\|\tilde{L}_\ell^{1/2} (I - P A_{\ell-1}^{-1} P^\top A_\ell) A_\ell^{-1} \tilde{L}_\ell^{-1/2}\| \lesssim \log p.$$

Using the identity $\|A^\top A\| \leq \|A\|^2$, we finally obtain the desired result. \square

Finally, we can show Theorem 4. Here, we follow the classical approach as introduced by Hackbusch, cf. [14].

PROOF (OF THEOREM 4). Lemma 7 yields $A_\ell \leq L_\ell$. Using standard arguments, cf. [17, Lemma 2] or [14], the smoothing property

$$\|L_\ell^{-1} A_\ell (I - L_\ell^{-1} A_\ell)^\nu\|_{L_\ell} \leq \frac{1}{\nu+1} \leq \frac{1}{\nu}$$

follows. Using Lemma 7, we obtain further

$$\|\tilde{L}_\ell^{-1} A_\ell (I - L_\ell^{-1} A_\ell)^\nu\|_{\tilde{L}_\ell} \lesssim \frac{p(\log p)^2 (1+L-\ell)^2 2^{L-\ell} \tau^* \delta^*}{\nu} \frac{\tau^* \delta^*}{\tau \delta},$$

which shows together Lemma 9

$$\begin{aligned}
& \|(I - L_\ell^{-1}A_\ell)^\nu(I - PA_{\ell-1}^{-1}P^\top A_\ell)(I - L_\ell^{-1}A_\ell)^\nu\|_{A_\ell} \\
& \leq \|(I - PA_{\ell-1}^{-1}P^\top A_\ell)(I - L_\ell^{-1}A_\ell)^\nu\|_{A_\ell} \\
& \leq \|(I - PA_{\ell-1}^{-1}P^\top A_\ell)(I - L_\ell^{-1}A_\ell)^\nu\|_{\tilde{L}_\ell} \\
& \leq \|(I - PA_{\ell-1}^{-1}P^\top A_\ell)A_\ell^{-1}\tilde{L}_\ell\|_{\tilde{L}_\ell}\|\tilde{L}_\ell^{-1}A_\ell(I - L_\ell^{-1}A_\ell)^\nu\|_{\tilde{L}_\ell} \\
& \lesssim \frac{p(\log p)^4(1 + L - \ell)^2 2^{L-\ell} \tau^* \delta^*}{\nu \tau \delta}.
\end{aligned}$$

This statement shows convergence of the two-grid method if ν is large enough. Standard arguments, cf. [14], allow to extend the analysis to the W-cycle multi-grid method.

Acknowledgments

The author was supported by the Austrian Science Fund (FWF): grant P31048, and by the bilateral project DNTS-Austria 01/3/2017 (WTZ BG 03/2017), funded by Bulgarian National Science Fund and OeAD (Austria).

References

- [1] R. Adams and J. Fournier, *Sobolev Spaces*, Academic Press, 2008, 2nd ed.
- [2] D. Arnold, *An interior penalty finite element method with discontinuous elements*, SIAM J. Numer. Anal. **19** (1982), no. 4, 742 – 760.
- [3] D. Arnold, F. Brezzi, B. Cockburn, and L. Marini, *Unified analysis of discontinuous Galerkin methods for elliptic problems*, SIAM J. Numer. Anal. **39** (2002), no. 5, 1749 – 1779.
- [4] 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. 07, 1031 – 1090.
- [5] L. Beirão da Veiga, A. Buffa, J. Rivas, and G. Sangalli, *Some estimates for h-p-k-refinement in isogeometric analysis*, Numer. Math. **118** (2011), no. 2, 271 – 305.
- [6] L. Beirão da Veiga, A. Buffa, G. Sangalli, and R. Vázquez, *Mathematical analysis of variational isogeometric methods*, Acta Numer. **23** (2014), 157 – 287.
- [7] L. Beirão da Veiga, D. Cho, L. Pavarino, and S. Scacchi, *Overlapping Schwarz methods for isogeometric analysis*, SIAM J. Numer. Anal. **50** (2012), no. 3, 1394 – 1416.

- [8] ———, *BDDC preconditioners for isogeometric analysis*, Math. Models Methods Appl. Sci. **23** (2013), no. 6, 1099 – 1142.
- [9] Á. Pé de la Riva, C. Rodrigo, and F. J. Gaspar, *An efficient multigrid solver for isogeometric analysis*, [arXiv:1806.05848], 2018.
- [10] M. Donatelli, C. Garoni, C. Manni, S. Serra-Capizzano, and H. Speleers, *Robust and optimal multi-iterative techniques for IgA Galerkin linear systems*, Comput. Methods. Appl. Mech. Eng. **284** (2015), 230 – 264.
- [11] M. Floater and E. Sande, *Optimal spline spaces of higher degree for L_2 n -widths*, J. Approx. Theor. **216** (2017), 1 – 15.
- [12] K. P. S. Gahalaut, J. K. Kraus, and S. K. Tomar, *Multigrid methods for isogeometric discretization*, Comput. Methods. Appl. Mech. Eng. **253** (2013), 413 – 425.
- [13] J. Gopalakrishnan and G. Kanschat, *A multilevel discontinuous Galerkin method*, Numer. Math. **95** (2003), no. 3, 527 – 550.
- [14] W. Hackbusch, *Multi-Grid Methods and Applications*, Springer, Berlin, 1985.
- [15] C. Hofer and U. Langer, *Dual-primal isogeometric tearing and interconnecting solvers for multipatch dG-IgA equations*, Comput. Methods. Appl. Mech. Eng. **316** (2017), 2 – 21.
- [16] C. Hofreither and S. Takacs, *Robust multigrid for isogeometric analysis based on stable splittings of spline spaces*, SIAM J. Numer. Anal. **4** (2017), no. 55, 2004 – 2024.
- [17] C. Hofreither, S. Takacs, and W. Zulehner, *A robust multigrid method for isogeometric analysis in two dimensions using boundary correction*, Comput. Methods Appl. Mech. Eng. **316** (2017), 22 – 42.
- [18] 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. Eng. **194** (2005), no. 39-41, 4135 – 4195.
- [19] S. K. Kleiss, C. Pechstein, B. Jüttler, and S. Tomar, *IETI – Isogeometric tearing and interconnecting*, Comput. Methods Appl. Mech. Eng. **247–248** (2012), 201 – 215.
- [20] U. Langer, A. Mantzaflaris, S. Moore, and I. Touloupoulos, *Multipatch discontinuous Galerkin Isogeometric Analysis*, Isogeometric Analysis and Applications 2014 (B. Jüttler and B. Simeon, eds.), Springer International Publishing, 2015, pp. 1 – 32.
- [21] U. Langer and I. Touloupoulos, *Analysis of multipatch discontinuous galerkin iga approximations to elliptic boundary value problems*, Computing and Visualization in Science **17** (2015), no. 5, 217 – 233.

- [22] A. Mantzaflaris, S. Takacs, et al., *G+Smo*, <https://github.com/gismo/gismo/wiki>, 2019.
- [23] M. Montardini, G. Sangalli, and M. Tani, *Robust isogeometric preconditioners for the stokes system based on the fast diagonalization method*, *Comput. Methods. Appl. Mech. Eng.* **338** (2018), 162 – 185.
- [24] E. Sande, C. Manni, and H. Speleers, *Sharp error estimates for spline approximation: explicit constants, n-widths, and eigenfunction convergence*, [arXiv: 1810.13418], 2018.
- [25] G. Sangalli and M. Tani, *Isogeometric preconditioners based on fast solvers for the Sylvester equation*, *SIAM J. Sci. Comput.* **38** (2016), no. 6, A3644 – A3671.
- [26] C. Schwab, *p- and hp-finite element methods: Theory and applications in solid and fluid mechanics*, *Numerical Mathematics and Scientific Computation*, Clarendon Press, Oxford, 1998.
- [27] J. Sogn, *Schur complement preconditioners for multiple saddle point problems and applications*, Ph.D. thesis, Johannes Kepler University Linz, Institute of Computational Mathematics, 2018.
- [28] J. Sogn and S. Takacs, *Robust multigrid solvers for the biharmonic problem in isogeometric analysis*, *Comput. Math. Appl.* **77** (2018), no. 1, 105 – 124.
- [29] S. Takacs, *Robust approximation error estimates and multigrid solvers for isogeometric multi-patch discretizations*, *Math. Models Methods Appl. Sci.* **28** (2018), no. 10, 1899 – 1928.
- [30] ———, *A quasi-robust approximation error estimate for discontinuous Galerkin Isogeometric Analysis*, [arXiv:1901.03262], 2019.
- [31] S. Takacs and T. Takacs, *Approximation error estimates and inverse inequalities for B-splines of maximum smoothness*, *Math. Models Methods Appl. Sci.* **26** (2016), no. 07, 1411 – 1445.
- [32] R. Tielen, M. Möller, D. Göttsche, and C. Vuik, *Efficient p-multigrid methods for isogeometric analysis*, [arXiv:1901.01685], 2019.