Weierstraß-Institut
für Angewandte Analysis und Stochastik

Leibniz-Institut im Forschungsverbund Berlin e. V.

Preprint ISSN 2198-5855

Hierarchical proximal Galerkin: A fast hp-FEM solver for variational problems with pointwise inequality constraints

Ioannis P. A. Papadopoulos

submitted: March 20, 2025

Weierstrass Institute
Mohrenstr. 39
10117 Berlin
Germany
E-Mail: ioannis.papadopoulos@wias-berlin.de

No. 3189
Berlin 2025

2020 *Mathematics Subject Classification.* 35J86, 35R35, 49J40, 49M41, 65K10, 65K15, 65N30.

Key words and phrases. Obstacle problem, gradient-type constraints, latent variable proximal point, hp-FEM.

The author was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689).

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Leibniz-Institut im Forschungsverbund Berlin e. V.
Mohrenstraße 39
10117 Berlin
Germany

Fax: +49 30 20372-303
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Hierarchical proximal Galerkin: A fast hp-FEM solver for variational problems with pointwise inequality constraints

Ioannis P. A. Papadopoulos

Abstract

We leverage the proximal Galerkin algorithm (Keith and Surowiec, Foundations of Computational Mathematics, 2024, DOI: 10.1007/s10208-024-09681-8), a recently introduced mesh-independent algorithm, to obtain a high-order finite element solver for variational problems with pointwise inequality constraints. This is achieved by discretizing the saddle point systems, arising from the latent variable proximal point method, with the hierarchical p-finite element basis. This results in discretized sparse Newton systems that admit a simple and effective block preconditioner. The solver can handle both obstacle-type, $u \leq \varphi$, and gradient-type, $|\nabla u| \leq \varphi$, constraints. We apply the resulting algorithm to solve obstacle problems with hp-adaptivity, a gradient-type constrained problem, and the thermoforming problem, an example of an obstacle-type quasi-variational inequality. We observe hp-robustness in the number of Newton iterations and only mild growth in the number of inner Krylov iterations to solve the Newton systems. Crucially we also provide wall-clock timings that are faster than low-order discretization counterparts.

1 Introduction

Minimizing the Dirichlet energy with an obstacle- or gradient-type pointwise constraint is known as the *obstacle problem* or *generalized elastic-plastic torsion problem*, respectively. The obstacle problem models the deflection of an elastic membrane in contact with an obstacle [37, 88] and due to its foundational nature, has wide-ranging applications in various fields including optimal control [32, 58, 82], topology optimization [20, 23, 75], and elasticity theory [49, 37] among many others. In turn, many PDE-constrained optimization problems are thoroughly reliant on an efficient obstacle problem solver. Gradient-type constraints arise in quasi-variational inequalities that model sandpile growth, river networks, semiconductors, strain-limited elastic material, and number of other applications cf. [8, 94, 90, 26, 80, 79, 7, 65, 35, 84, 91, 30, 83].

In this paper we discretize the latent variable proximal point method (LVPP) of Keith and Surowiec [57] with the hierarchical p-finite element method (p-FEM) to construct the hierarchical proximal Galerkin algorithm (hpG), a solver that enjoys the following properties:

(i) h- and p-independent number of nonlinear iterations until convergence;

(ii) arbitrarily fast convergence to solutions of both the obstacle problem and minimization problems with gradient-type constraints;

(iii) high-order discretizations (see Section 7.4 for a discretization with polynomials up to total degree 164 $(p = 82)$ on each quadrilateral element of a 4×4 mesh);

(iv) competitive solve times with lower order methods to reach a prescribed error;

(v) a block-diagonal preconditioner resulting in mild polylogarithmic growth in the inner Krylov method iterations as $h \to 0$ and $p \to \infty$;

DOI 10.20347/WIAS.PREPRINT.3189 Berlin 2025

(vi) fast sparse factorizations for the mass, stiffness, and preconditioner matrices;

(vii) fast quasi-optimal complexity quadrature via fast transforms and the discrete cosine transform (DCT).

1.1 Motivation

For nontrivial constraints, the regularity of a solution u is capped at $u \in H^s(\Omega)$, for $s < 5/2$, where $H^s(\Omega)$ denotes the usual Sobolev spaces $W^{s,2}(\Omega)$ [1]. This is true even if the problem is posed on a smooth domain Ω with smooth data and a smooth obstacle [27, Th. 3][1]. As such low-order discretizations are often favoured due to their simplicity, perceived reduced computational cost due to sparsity, and efficiency in convergence due to low solution regularity. For the obstacle problem, there exist efficient low-order optimal complexity (but mildly mesh dependent) multigrid (MG) techniques [41, 24, 44, 50, 51, 62, 63, 29, 95]. There is a strong correspondence between these MG solvers and active-set strategies [41, Sec. 6] which require that nodal feasibility in the discretization implies pointwise feasibility. In turn this requirement typically restricts a user to the lowest-order continuous FEM basis[2]. A reader may now arrive at the following question:

Why develop a high-order discretization for pointwise-constrained problems?

In this paper, we argue that, per degree of freedom (dof), a high-order discretization may induce an error that is orders of magnitude smaller than the low-order counterpart – particularly in the H^1-norm. Of course, fewer dofs does not imply faster solve times. Matrix assembly overhead may significantly increase as sparsity is often greatly reduced, resulting in a substantial increase in the number of nonzero entries that must be evaluated. Moreover, each individual nonzero entry becomes more expensive to compute. The loss of sparsity also impacts the effectiveness of sparse linear solvers further reducing solver speed. However, in recent years, there have been a number of developments that allow for fast expansions and evaluations of the high-order orthogonal polynomials that form the hierarchical p-FEM basis which, incidently, when used to discretize the LVPP subproblems [57], induces discretized linear systems where the loss of sparsity is heavily mitigated [92, 60, 76]. By leveraging these developments, we will present concrete counterexamples where accurate solutions are cheaper to obtain with high-order discretizations.

Secondly, despite the cap on the regularity of the solution, convergence rates faster than $O((h/p)^{3/2})$ in the H^1-norm are achievable if the mesh aligns with the regions in the solution where the loss of regularity occurs. In our context, these regions are where the pointwise constraints transition from being active to inactive. These are not known beforehand. However, one may develop estimators to locate these regions and, particularly in the case of gradient-type constraints, the location of the transitions may sometimes be inferred from the problem data. In Section 7.1 and Section 7.3 we present two examples where we obtain convergence rates faster than $3/2$ in the regimes of h and p we consider. Note that a discretization with degree $p = 1$ will only ever converge at a rate of $O(h)$.

Aside from solver speed and convergence rates, high-order discretizations often offer other benefits. They minimize numerical artefacts in fluid dynamics and elasticity [54], e.g. in two dimensions, locking in linear elasticity does not occur if $p \geq 4$ [2]. As pointwise constraints naturally occur in models for

[1] For a constant gradient-type constraint, the fact that $u \in H^s(\Omega)$, for $s < 5/2$, follows by rewriting the gradient-type constrained problem as an obstacle problem with a Lipschitz continuous obstacle [91, Th. 2.2]. The result in [27, Th. 3] then applies.

[2] A recent study has shown success with active-set strategies and a carefully chosen quadratic or cubic FEM basis [59].

such settings (e.g. the topology optimization of linear elasticity [20, 74] and fluids [23, 77, 73]) then utilizing a solver that is then confined to the lowest-order (continuous) FEM discretization is limiting. High-order methods are also amenable to parallelization making them suitable for modern computing architectures cf. [60, 6].

The proximal Galerkin (pG) algorithm arises after a Galerkin discretization of LVPP which was first introduced in the context of the obstacle problem by Keith and Surowiec [57] and generalized to a number of other problems in a recent review [34]. LVPP is formulated on the infinite-dimensional level and not tied to any particular choice of discretization. The main component of the algorithm is the repeated solve of a coupled nonlinear system of PDEs. Although not yet proven, experiments show that the number of iterations of the pG solver is mesh (h) and degree (p) independent. This, among other reasons, is a motivating factor for choosing a variation of the pG solver in this work. A second reason is that after a Newton linearization of the pG nonlinear system, one arrives at a saddle point problem with one nontrivial term. Hence, with a sparsity promoting discretization, the density of a high-order method is localized to the nontrivial term which in turn is effectively handled by a simple block preconditioning strategy.

In principle, a high-order FEM discretization of a path-following penalty method can be used to enforce the pointwise constraints. In exact arithmetic, this would lead to mesh independent iteration counts [45, 48]. However, in order to extract optimal convergence rates for the obstacle problem when $p > 1$, one must scale the penalty parameter as $O(h^{-3})$ which also scales the condition number of the arising discretized linear systems by the same factor [42, Sec. 3.1]. Ill-conditioning may quickly cause mesh dependent effects and a potential blowup in the required number of nonlinear iterations. Moreover, it is unclear how to handle the loss of sparsity as $p \to \infty$.

We are not the first to discuss high-order discretizations for the obstacle problem [78]. Keith and Surowiec, in the seminal pG paper, solve a 2D obstacle problem with a mesh consisting of five cells and a degree $p = 12$ basis [57, Fig. 11]. They were capped at $p = 12$ by their choice of FEM software and their chosen basis induces increasingly dense stiffness and mass matrices as $p \to \infty$. A recent paper includes an obstacle problem example with $p = 48$ [34, Sec. 3.1] although the discretization is a spectral method limited to a single-cell disk mesh. Other works include [64, 43] as well as a sequence of successful papers by Banz, Schröder, and coauthors [18, 19, 16, 17]. In these works the choice of FEM basis also leads to increasingly denser stiffness and mass matrices as $p \to \infty$ and the outer nonlinear solver is mesh dependent, i.e. the number of nonlinear iterations grow as $h \to 0$ and $p \to \infty$. However, in [18], Banz and Schröder successfully consider examples with $p = 30$ and implement a fully h- and p-adaptive scheme in two dimensions, complete with a posteriori estimators, leading to highly accurate approximations. Also of significant note is a hp-FEM discretization to a problem involving gradient-type constraints in [15], where Bammer, Banz, and Schröder considered a discretization with $p = 50$.

1.2 Contributions

The main contribution of this paper is the investigation of a fast sparsity-preserving hp-FEM discretization of the subproblems of the LVPP algorithm applied to the obstacle and generalized elastic-plastic torsion problems, with a view of providing a bedrock for its potential in applications. We develop the first preconditioners for the Newton systems arising in a pG method and observe them to be exceptionally effective. We notice that the quadrature for the assembly of the matrices at each Newton step can be confined to a discontinuous basis consisting of Legendre polynomials which admit fast evaluation and expansion transforms. We then leverage an atypical but fast quadrature scheme. We also see

that the method may adopt error estimators already developed in the literature to guide hp-refinement. Altogether this allows us to consider examples with the highest discretization degree that have been reported in the literature to date. Moreover, since the novel hpG solver retains sparsity, requires little computational overhead, and exhibits hp-independent number of nonlinear iterations, we are able to provide the first reported competitive wall-clock timing comparisons with other solvers featuring low-order discretizations. In fact in Section 7, we observe up to a 24 and 100 times speed up, per linear solve, to reach the same error for an obstacle problem and generalized elastic-plastic torsion problem, respectively (cf. Sections 7.2 and 7.3).

We begin by introducing the Dirichlet minimization problem with obstacle- and gradient-type constraints, together with the pG algorithm in Section 2, followed the construction of the hierarchical p-FEM basis in Section 3. Next we provide a detailed investigation of the discretized Newton linearization of the pG subsystems in Section 4 and the design the preconditioning strategy in Section 4.4. hp-adaptivity is discussed in Section 5 and we outline four algorithmic techniques that significantly speed up the algorithm when $p \gg 1$ in Section 6.

We consider four numerical examples in Section 7. Section 7.1 tackles a one-dimensional obstacle problem with an oscillatory forcing-term where hp-adaptivity is fully explored. Section 7.2 handles a two-dimensional obstacle problem where it is clear that high-order discretizations give smaller errors both with respect to the dofs *and* wall-clock time (see Figure 6). The penultimate problem in Section 7.3 investigates the effectiveness of the solver applied to a gradient-type constrained problem. We conclude the examples in Section 7.4 by computing the approximate solution of a two-dimensional thermoforming problem, an obstacle-type quasi-variational inequality, with a discretization consisting of 16 cells and multivariate polynomials up to total degree 164 on each element. Finally we give our conclusions in Section 8.

2 Problem setup and latent variable proximal point

Consider an open, convex, and bounded Lipschitz domain $\Omega \subset \mathbb{R}^d$, $d \in \{1, 2\}$. Let $\langle \cdot, \cdot \rangle_{X^*,X}$ denote the (topological) duality pairing between a Banach space X and its dual space X^*. The inner product of a Hilbert space H is denoted by $(\cdot, \cdot)_H$. Let $W^{s,p}(\Omega)$, $s > 0$, $p \geq 1$, denote the usual Sobolev spaces and let $L^p(\Omega)$, $p \geq 1$, denote the Lebesgue spaces [1]. Consider the Hilbert spaces $H^s(\Omega) \coloneqq W^{s,2}(\Omega)$, $s > 0$ and denote by $H^1_0(\Omega) \coloneqq \{v \in H^1(\Omega) : v|_{\partial\Omega} = 0\}$ where $|_{\partial\Omega} : W^{1,p}(\Omega) \to W^{1-\frac{1}{p},p}(\partial\Omega)$ denotes the standard boundary trace operator [40]. Consider the Dirichlet energy functional

$$J(v) \coloneqq \frac{1}{2}\|\nabla v\|^2_{L^2(\Omega)} - (f, v)_{L^2(\Omega)}. \tag{1}$$

We seek the minimizer $u \in H^1_0(\Omega)$ of the Dirichlet energy functional J, with datum $f \in L^2(\Omega)$, while satisfying the obstacle or gradient-type constraint with $\varphi \in H^1(\Omega)$. In other words we consider one of the obstacle- or gradient-type constrained minimization problems:

$$\min_{u \in K} J(u) \quad \text{where} \quad K = \{v \in H^1_0(\Omega) : v \leq \varphi \text{ a.e. in } \Omega\} \qquad \text{(obstacle-type)}, \tag{2a}$$

$$\min_{u \in K} J(u) \quad \text{where} \quad K = \{v \in H^1_0(\Omega) : |\nabla v| \leq \varphi \text{ a.e. in } \Omega\} \qquad \text{(gradient-type)}. \tag{2b}$$

A classical calculus of variations result reveals that the solution $u \in K$ of either (2a) or (2b) also satisfies the variational inequality [36, Ch. 8.4]

$$(\nabla u, \nabla(v - u))_{L^2(\Omega)} \geq (f, v - u)_{L^2(\Omega)} \text{ for all } v \in K. \tag{3}$$

The existence and uniqueness of the minimizer follows by classical results provided the data is compatible, e.g. $\varphi|_{\partial\Omega} > 0$ a.e. on $\partial\Omega$ for the obstacle problem [66].

2.1 LVPP: obstacle problem

LVPP approximates the solution of the obstacle problem (2a), on the infinite-dimensional level, by solving a series of nonlinear mixed saddle point systems where a parameter $\alpha \in \mathbb{R}_+$ and a source term is updated. More specifically, given the datum $f \in L^\infty(\Omega)$, the obstacle $\varphi \in H^1(\Omega)$, the parameter $\alpha_k > 0$, $k \in \mathbb{N}$, a previous latent variable iterate $\psi_{k-1} \in L^\infty(\Omega)$ and setting $\psi_0 \equiv 0$, the pG subproblem seeks $(u_k, \psi_k) \in H^1_0(\Omega) \times L^\infty(\Omega)$ that satisfies, for all $(v, \zeta) \in H^1_0(\Omega) \times L^\infty(\Omega)$ [57, Alg. 3],

$$\begin{aligned} \alpha_k(\nabla u_k, \nabla v)_{L^2(\Omega)} + (\psi_k, v)_{L^2(\Omega)} &= \alpha_k(f, v)_{L^2(\Omega)} + (\psi_{k-1}, v)_{L^2(\Omega)}, \\ (u_k, \zeta)_{L^2(\Omega)} + (\mathrm{e}^{-\psi_k}, \zeta)_{L^2(\Omega)} &= (\varphi, \zeta)_{L^2(\Omega)}. \end{aligned} \tag{4}$$

It was shown that as long as $\sum_{k=1}^N \alpha_k \to \infty$ as $N \to \infty$ then $u_k \to u^*$ strongly in $H^1(\Omega)$ and $\lambda_k \coloneqq (\psi_{k-1} - \psi_k)/\alpha_k \to \lambda^*$ strongly in $H^{-1}(\Omega)$ where u_* is the solution of the obstacle problem and $\lambda^* \coloneqq -\Delta u^* - f$ is its associated Lagrange multiplier [57, Th. 4.13]. By driving α_k larger at a faster-than-geometric rate (e.g. $\alpha_k = k(k+1)(k+2)\cdots(k+m)$ for some $m \in \mathbb{N}$), one can converge to the solution of the obstacle problem with superlinear convergence [57, Cor. A.12]. Moreover, unlike other mesh independent solvers that rely solely on a penalty term, cf. [47, 45], the solver still converges to the solution even if α_k is kept at a fixed value, albeit at a sublinear rate [57, Cor. A.12]. The faster α_k increases, the quicker the convergence to the solution of the original obstacle problem but at the cost of a harder nonlinear problem at each iteration.

2.2 LVPP: gradient-type constraint

Recently LVPP was adapted to handle gradient-type constraints [34, Sec. 4.1] although the convergence properties and choices of stable discretizations have not yet been theoretically justified. Nevertheless, experimentally, we observe very similar behaviours as for the obstacle problem. As before, suppose we are given the datum $f \in L^\infty(\Omega)$, the constraint function $\varphi \in L^\infty(\Omega) \cap H^1(\Omega)$ where $\operatorname{ess\,inf} \varphi \geq c$ for some $c > 0$, the parameter $\alpha_k > 0$, $k \in \mathbb{N}$, and a previous latent variable iterate $\psi_{k-1} \in L^\infty(\Omega)^d$ where if $k = 1$ then $\psi_0 \equiv 0$. Then, the pG gradient-type subproblem for approximating the solution of (2b) is to find $(u_k, \psi_k) \in H^1_0(\Omega) \times L^\infty(\Omega)^d$ that satisfies, for all $(v, \zeta) \in H^1_0(\Omega) \times L^\infty(\Omega)^d$ [34, Eq. (4.3)],

$$\begin{aligned} \alpha_k(\nabla u_k, \nabla v)_{L^2(\Omega)} + (\psi_k, \nabla v)_{L^2(\Omega)} &= \alpha_k(f, v)_{L^2(\Omega)} + (\psi_{k-1}, \nabla v)_{L^2(\Omega)}, \\ (\nabla u_k, \zeta)_{L^2(\Omega)} &= (\varphi\psi_k(1 + |\psi_k|^2)^{-1/2}, \zeta)_{L^2(\Omega)}. \end{aligned} \tag{5}$$

3 Hierarchical p-FEM basis

The hierarchical p-FEM basis was pioneered in the 1980s and 1990s by Babuška, Szabó and coauthors [12, 10, 13, 11, 89]. The basis contains very high-order piecewise polynomials whilst retaining sparsity in the stiffness (weak Laplacian) and mass matrices. More specifically, the basis functions consist of shape and external functions, otherwise known as bubble and hat functions. The shape

functions are translated and scaled weighted orthogonal polynomials whose support are contained within a single cell in the mesh. Hence, shape functions tested against shape functions supported on another cell evaluate to zero. Moreover, thanks to the orthogonality properties, testing many of the basis functions with other basis functions supported on the same cell also evaluate to zero. For more details we refer the reader to [85, 54, 60].

In this work we consider both a continuous H^1-conforming hierarchical p-FEM space, which we denote $U_{h,\boldsymbol{p}}$ to discretize u and a discontinuous L^2-conforming space denoted by $\Psi_{h,\boldsymbol{p}}$ to discretize ψ. We first define these spaces on a one-dimensional domain $\Omega = (a, b)$.

Consider the one-dimensional mesh $\mathcal{T}_h = \{x_i\}_{i=1}^m$ where $a = x_0 < x_1 < \cdots < x_m = b$ and $h = \max_i |x_{i+1} - x_i|$. Let $K_i = [x_i, x_{i+1}]$ and consider the set

$$\Psi_{K_i,p} \coloneqq \left\{ P_n(y) : y = \frac{2x - x_i - x_{i+1}}{x_{i+1} - x_i},\ 0 \leq n \leq p \right\}, \tag{6}$$

where $P_n(x)$ denotes the Legendre polynomial of degree n supported on the interval $[-1, 1]$ [67, Sec. 18.3]. For $x \in \mathbb{R}\backslash[-1, 1]$ we define $P_n(x) = 0$. The Legendre polynomials are orthogonal with respect to the L^2-inner product, i.e. $(P_n, P_m)_{L^2(-1,1)} = \frac{2\delta_{nm}}{2n+1}$, where δ_{nm} is the Kronecker delta.

Definition 3.1 (L^2-conforming hierarchical p-FEM space $\Psi_{h,\boldsymbol{p}}$)**.** *Consider the one-dimensional mesh $\mathcal{T}_h = \cup_{i=1}^m K_i$. We define the L^2-conforming hierarchical p-FEM space $\Psi_{h,\boldsymbol{p}}$, $\boldsymbol{p} = (p_1, \ldots, p_m)$ as the union of the cellwise bases:*

$$\Psi_{h,\boldsymbol{p}} \coloneqq \operatorname{span} \bigcup_{i=1}^{m} \Psi_{K_i,p_i}. \tag{7}$$

In other words $\Psi_{h,\boldsymbol{p}}$ consists of scaled and translated Legendre polynomials on cell K_i up to degree p_i where each basis function has a support that is contained in a single cell on the mesh.

Lemma 3.2 (Diagonal mass matrix)**.** *The mass matrix of the discontinuous finite element space $\Psi_{h,\boldsymbol{p}}$ is diagonal, i.e. $(\zeta_i, \zeta_j)_{L^2(\Omega)} = \frac{|K_i|\delta_{ij}}{2i+1}$ for all basis functions $\zeta_i, \zeta_j \in \Psi_{h,\boldsymbol{p}}$ and $K_i \in \mathcal{T}_h$ is the cell such that $\operatorname{supp}(\zeta_i) \subseteq K_i$.*

The result in Lemma 3.2 follows as a direct consequence of the fact that each basis function is only supported on one cell on the mesh as well as the orthogonality of the basis functions supported on the same cell with respect to the L^2-norm.

The H^1-conforming space $U_{h,\boldsymbol{p}}$ consists of the union of the set of standard piecewise linear hat functions with weighted Jacobi polynomials. Given two adjacent cells K_{i-1}, K_i, let $H_i(x)$ denote the usual continuous piecewise linear hat function supported on $K_{i-1} \cup K_i$ [25, Ch. 0.4]. Moreover, consider the polynomials

$$W_n(x) \coloneqq \frac{(1 - x^2)P_n^{(1,1)}(x)}{2(n + 1)}, \tag{8}$$

where $P_n^{(1,1)}(x)$ are the Jacobi polynomials orthogonal with respect to the weight $1-x^2$ on the interval $[-1, 1]$ [67, Sec. 18.3]. The normalization is chosen such that $\frac{\mathrm{d}}{\mathrm{d}x} W_n(x) = -P_{n+1}(x)$ for $n \geq 0$. We define the set of shape functions for the cell K_i as

$$B_{K_i,p} \coloneqq \left\{ W_n(y) : y = \frac{2x - x_i - x_{i+1}}{x_{i+1} - x_i},\ 0 \leq n \leq p - 2 \right\}. \tag{9}$$

Definition 3.3 (H^1-conforming hierarchical p-FEM space $U_{h,\boldsymbol{p}}$)**.** *Consider the one-dimensional mesh $\mathcal{T}_h = \cup_{i=1}^m K_i$. We define the H^1-conforming hierarchical p-FEM space $U_{h,\boldsymbol{p}}$, $\boldsymbol{p} = (p_1, \dots, p_m)$, as:*

$$U_{h,\boldsymbol{p}} \coloneqq \operatorname{span} \left\{ \{H_i\}_{i=1}^{m+1} \cup \bigcup_{i=1}^{m} B_{K_i,p_i} \right\}. \tag{10}$$

We use $U_{0,h,\boldsymbol{p}}$ to denote the space

$$U_{0,h,\boldsymbol{p}} \coloneqq \{v_{hp} \in U_{h,\boldsymbol{p}} : v_{hp}|_{\partial\Omega} = 0\}. \tag{11}$$

We take the tensor product space for two dimensions and label the spaces as $U_{h,\boldsymbol{p}_x,\boldsymbol{p}_y} = U_{h,\boldsymbol{p}_x} \otimes U_{h,\boldsymbol{p}_y}$ and $\Psi_{h,\boldsymbol{p}_x,\boldsymbol{p}_y} = \Psi_{h,\boldsymbol{p}_x} \otimes \Psi_{h,\boldsymbol{p}_y}$. We refer to the degree of the pre-tensor one-dimensional polynomials as the *partial degree* and the degree of the resulting multivariate polynomial as the *total degree*. With a slight abuse of notation, we will often compactify the subscripts such that $U_{hp} = U_{h,\boldsymbol{p}}$ and $\Psi_{hp} = \Psi_{h,\boldsymbol{p}}$ in 1D and $U_{hp} = U_{h,\boldsymbol{p}_x,\boldsymbol{p}_y}$ and $\Psi_{hp} = \Psi_{h,\boldsymbol{p}_x,\boldsymbol{p}_y}$ in 2D.

We take the opportunity to introduce a discontinuous basis consisting of Legendre spectral Galerkin polynomials [86]. This basis will later be used in the stabilization term for gradient-type constraints as well as to construct a preconditioner for the obstacle problem. Consider a one-dimensional reference cell $K = [-1, 1]$. We define the spectral Galerkin polynomials, for $n \in \mathbb{N}_0$:

$$Y_n(x) \coloneqq P_n(x) - P_{n+2}(x). \tag{12}$$

Note that $Y_n(\pm 1) = 0$ for all $n \in \mathbb{N}_0$.

Remark 3.4. *In fact each polynomial Y_n is a rescaling of the shape function W_n defined in* (8)*, i.e. $Y_n(x) = c_n W_n(x)$ for some $c_n \in \mathbb{R}$. Nevertheless, we found that the numerical convergence of the linear system iterative solvers proposed later were often better when utilizing the rescaling rather than the shape functions directly. Hence we choose to keep the additional notation Y_n for clarity on where a rescaling is used or not.*

For simplicity we assume that the same degree p is used on each cell. Then, given a one-dimensional mesh $\mathcal{T}_h = \cup_{i=1}^m K_i$ we define the local cellwise basis

$$\Phi_{K_i,p} \coloneqq \left\{ Y_n(y) : y = \frac{2x - x_i - x_{i+1}}{x_{i+1} - x_i},\ 0 \le n \le p \right\}, \tag{13}$$

and gluing the cells together forms the discontinuous 1D global basis

$$\Phi_{h,p} \coloneqq \operatorname{span} \bigcup_{i=1}^{m} \Phi_{K_i,p}. \tag{14}$$

As before, the two-dimensional basis is constructed via a tensor-product, i.e. $\Phi_{h,p_x,p_y} \coloneqq \Phi_{h,p_x} \otimes \Phi_{h,p_y}$ for $p_x, p_y \in \mathbb{N}_0$.

4 Proximal Galerkin, Newton systems & preconditioning

We employ a mixed $H^1 \times L^2$-conforming hierarchical finite element discretization of the LVPP saddle-point problems for (u, ψ) in (4) and (5) where, in particular, the discretization is continuous for u ($u_{hp} \in U_{hp}$) and discontinuous for ψ ($\psi_{hp} \in \Psi_{hp}$ in (4) and $\psi_{hp} \in \Psi_{hp}^d$ in (5)). We refer the discretized nonlinear systems as the hpG subproblems. Below we first discuss the discretization of the obstacle-type LVPP subproblem (4) and detail the differences for the gradient-type LVPP subproblems in Section 4.2.

4.1 Obstacle problem

For the obstacle-type LVPP subproblem (4), we discretize u with (partial) degree $\boldsymbol{p}$ and ψ with (partial) degree $\boldsymbol{p}-2$ for $\boldsymbol{p}\geq 2$, where ± 1 and $\geq$ are to be understood entrywise. With a slight abuse of notation, we denote the finite element functions approximating u and ψ by u_{hp} and ψ_{hp}, respectively.

After a Newton linearization of the FEM discretization of (4), a routine derivation reveals that one must (approximately) solve the following symmetric saddle-point linear system at each Newton step:

$$G \coloneqq \begin{pmatrix} A_\alpha & B \\ B^\top & -D_\psi - E_\beta \end{pmatrix} \begin{pmatrix} \boldsymbol{\delta}_u \\ \boldsymbol{\delta}_\psi \end{pmatrix} = \begin{pmatrix} \mathbf{b}_u \\ \mathbf{b}_\psi \end{pmatrix}, \tag{15}$$

where $\mathbf{b}_u$ and $\mathbf{b}_\psi$ denote the u and ψ components of the residual and $\boldsymbol{\delta}_u$ and $\boldsymbol{\delta}_\psi$ are the coefficient vectors for the Newton updates of u_{hp} and ψ_{hp}, respectively. $A_\alpha \coloneqq \alpha A$ is the α-scaled stiffness matrix for U_{hp}, B is the Gram matrix between $U_{0,hp}$ and Ψ_{hp}, D_ψ is the discretization of the nonlinear term, and E_β is a symmetric positive-definite stabilization matrix such that

$$\begin{aligned} A_{ij} &= (\nabla v_i, \nabla v_j)_{L^2(\Omega)}, & [D_\psi]_{ij} &= (\zeta_i, \mathrm{e}^{-\psi_{hp}}\zeta_j)_{L^2(\Omega)}, \\ B_{ij} &= (v_i, \zeta_j)_{L^2(\Omega)}, & [E_\beta]_{ij} &= \beta(\zeta_i, \zeta_j)_{L^2(\Omega)}, \end{aligned} \tag{16}$$

for each basis function $v_i \in U_{0,hp}$ and $\zeta_i \in \Psi_{hp}$ and user-chosen parameter $\beta \geq 0$. E_β is augmented to the bottom right block in order reduce the condition number of G but is not required for well-posedness. In fact, in most examples in Section 7, we fix $\beta = 0$ which corresponds to $E_\beta \equiv 0$. Note that D_ψ is the only matrix with dependence on the current Newton iterate.

4.2 Gradient-type constraint

In order to discretize the gradient-type LVPP subproblem (5) we choose a similar discretization. Now we discretize u with (partial) degree $\boldsymbol{p}$ and ψ with (partial) degree $\boldsymbol{p}-1$ for $\boldsymbol{p}\geq 1$. After a Newton linearization, this leads to the same symmetric saddle point structure as (15) that we choose not to relabel. The α-scaled stiffness matrix A_α is identical to the one in the previous subsection. Recall that Φ_{hp} denotes the discontinuous basis of Legendre spectral Galerkin basis defined in (14). Then, for each basis function $v_i \in U_{0,hp}$, $\zeta_i \in \Psi^d_{hp}$, and $\eta_i \in \Phi^d_{hp}$, $d \in \{1,2\}$, the (not relabeled) matrices B, D_ψ, and E_β are defined as:

$$B_{ij} = (\nabla v_i, \zeta_j)_{L^2(\Omega)}, \quad [D_\psi]_{ij} = \langle F'_{\zeta_i}(\psi_{hp}), \zeta_j\rangle, \quad [E_\beta]_{ij} = \beta(\nabla_h \eta_i, \nabla_h \eta_j)_{L^2(\Omega)}, \tag{17}$$

where ∇_h denotes the broken (cellwise) gradient, $F'_{\zeta_i}(\psi_{hp})$ represents the Fréchet derivative of $F_{\zeta_i}(\psi_{hp}) \coloneqq (\zeta_i, \varphi_{hp}\psi_{hp}(1+|\psi_{hp}|^2)^{-1/2})_{L^2(\Omega)}$, and φ_{hp} denotes the FEM approximation of φ. E_β plays the same role as in Section 4.1 and, once again, D_ψ is the only matrix with dependence on the current Newton iterate. The effectiveness of the hpG algorithm hinges on the ability to solve (15) efficiently and in a manner that is robust to choices of the mesh size h, the truncation degree p, α, and ψ_{hp}.

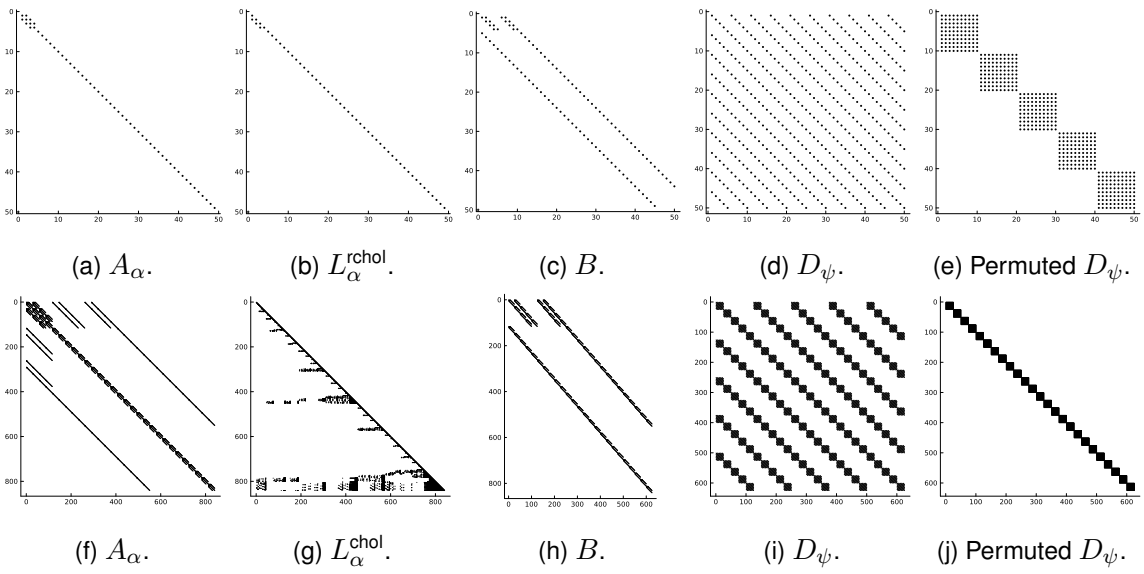
4.3 The matrices A_α, B, D_ψ, and E_β

The matrices A, B, and E_β are independent of any of the parameters of the pG algorithm and sparse with $O(p^d/h^d)$ entries that are nonzero. In particular E_β is block-diagonal after a permutation of

the rows and columns. Hence they can be efficiently assembled, stored and applied to a vector in $O(p^d/h^d)$ flops. Moreover, when $d = 1$, A admits a reverse Cholesky factorization[3] $A = L^\top L$ such that the factor L is also sparse and has $O(p/h)$ nonzero entries [60, Th. 4.2]. Thus the inverse of A_α may be computed and applied with $O(p/h)$ flops. In two dimensions, we observe that the Cholesky factorization of A has only a moderate fill-in and can be computed quickly. In this paper, we always performed this factorization once at the start of a solve. However, we note that an optimal complexity iterative solver also exists for A in 2D [60, 76] via the ADI algorithm [38].

The matrix D_ψ is not as sparse as A_α and B. However, since all the basis functions in Ψ_{hp} are supported on a maximum of one cell, the columns and rows of D_ψ may be permuted to a block diagonal structure although the blocks themselves may be dense, leading to $O((p-j)^{2d}/h^d)$ nonzero entries where $j = 2$ for the obstacle problem and $j = 1$ in the gradient-type constraint case. We note that we can apply the action of D_ψ with quasi-optimal $O((p-j)^d \log^d(p-j)/h^d)$ flops. Since we are able to apply D_ψ in an efficient manner and have a fast inverse for the top left block A_α, this motivates a block preconditioning approach coupled with a matrix-free Krylov method.

In Figure 1 we provide spy plots of the individual blocks in G in both a one and two-dimensional discretization for the pG obstacle subproblem (4). The gradient-type case results in similar sparsity patterns. The discretization involves five cells and $p = 10$ on each cell in 1D and 25 cells and partial degree $p = 5$ on each cell in 2D. Note the sparsity in A_α, its (reverse) Cholesky factor, and B. Moreover, D_ψ has a block diagonal structure (after rearranging the rows and columns) and hence computing its inverse can be performed cellwise and in parallel.



(a) A_α. (b) L_α^{rchol}. (c) B. (d) D_ψ. (e) Permuted D_ψ.

(f) A_α. (g) L_α^{chol}. (h) B. (i) D_ψ. (j) Permuted D_ψ.

Figure 1: The spy plots of the scaled stiffness matrix A_α, its lower Cholesky factor L_α^{chol} or reverse lower Cholesky factor L_α^{rchol}, the Gram matrix B, the exponential block D_ψ, and D_ψ with permuted rows and columns to reveal the hidden block diagonal structure for the Newton linearization of the pG obstacle subproblem (4) when using: (top row) a one-dimensional p-FEM discretization on a uniform mesh with five cells and degree $p = 10$ on each cell and (bottom row) a two-dimensional tensor-product p-FEM discretization with 25 cells and partial degree $p = 5$ on each cell.

[3] A *reverse* Cholesky factorization $A = L^\top L$ initializes the factorization from the bottom right corner of a symmetric positive-definite matrix A rather than the top left.

4.4 Block preconditioning

Given that the submatrices within G in (15) are relatively sparse, one can assemble the full matrix and perform a sparse LU factorization–a strategy that showed unexpectedly good performance in our examples. However, by employing a straightforward block preconditioning strategy, we can improve the conditioning of the solver and achieve faster solve times, particularly as $h \to 0$ and $p \to \infty$. Block preconditioning reduces the problem to solving a smaller system involving a dense Schur complement matrix. Despite its density, this matrix allows for an efficient matrix-vector product, making it well-suited for solution via Krylov subspace methods. As usual, maintaining an acceptable iteration count depends on the availability of a suitable preconditioner. In this subsection we develop a (up to permutation) block-diagonal preconditioner, whose effectiveness is expected to deteriorate at a worse-case polylogarithmic rate with respect to p and h.

The first step of the block preconditioner is to perform a Schur complement factorization of the matrix in (15). Let $\mathbf{y} = \mathbf{b}_\psi - B^\top A_\alpha^{-1}\mathbf{b}_u$. Then a routine calculation reveals that

$$\boldsymbol{\delta}_\psi = S^{-1}\mathbf{y} \tag{18a}$$

$$\boldsymbol{\delta}_u = A_\alpha^{-1}(\mathbf{b}_u - B\boldsymbol{\delta}_\psi) \tag{18b}$$

where the Schur complement matrix S is $S \coloneqq -(D_\psi + E_\beta + B^\top A_\alpha^{-1} B)$. (18b) already has a fast solve thanks to the sparse (reverse) Cholesky factorization of A_α and fast action of B. Hence a fast solve for the full system hinges on computing the inverse of S quickly in (18a).

As is typical with the Schur complement, S is dense due to the $B^\top A_\alpha^{-1} B$ term. Nevertheless, since the action of S can be applied efficiently (each matrix in the triple matrix product $B^\top A_\alpha^{-1} B$ is applied sequentially), we may solve (18b) via a right-preconditioned GMRES Krylov method. Now the challenge is to construct a preconditioner for S that admits a cheap assembly and factorization.

Remark 4.1 (Other Krylov methods)**.** *The linear system in* (15) *is symmetric and thus amenable to cheaper Krylov methods such as MINRES. In our implementation we utilize a quadrature scheme that does not preserve the symmetry of D_ψ (but provides other advantages such as speed) and therefore required a Krylov method that does not rely on symmetry for convergence. We refer the reader to Section 6 for further details.*

4.5 Obstacle problem: a Schur complement preconditioner

In this subsection we focus on preconditioning the Schur complement as it arises in the context of the pG obstacle subproblem (4). For ease of implementation, we will restrict our search for a preconditioner for S in (18a) that can be assembled and factorized. Therefore, we do not consider preconditioners based on p-multigrid in this work [13, 21, 61, 85].

S maps a vector from the primal to the dual space of Ψ_{hp}. Our goal is to preserve the mapping and approximate B and A_α with the alternative matrices $\hat{B}$ and $\hat{A}_\alpha$ such that the new matrix-triple-product $\hat{B}^\top \hat{A}_\alpha^{-1} \hat{B}$ induces a (up to permutation) block-diagonal matrix. Common choices involve removing or modifying the hat functions in $U_{0,hp}$ [13, Sec. 3]. We opt for removing the hat functions and rescaling the remaining shape functions, i.e. we use the Legendre spectral Galerkin basis Φ_{hp} defined in (14) which as we found it exhibited the lowest iteration counts. We choose the following sparse approximation for S

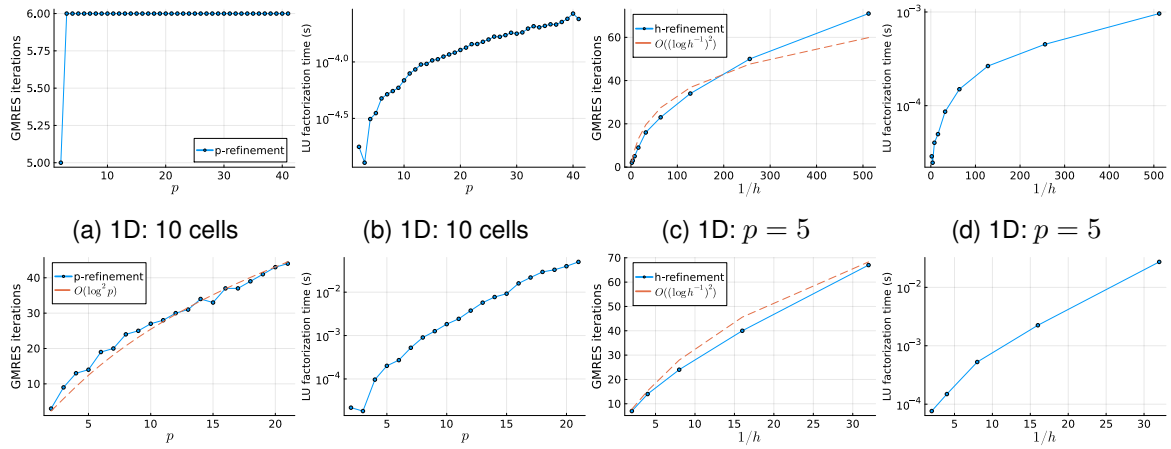
$$\hat{S} = -D_\psi - E_\beta - \hat{B}^\top \hat{A}_\alpha^{-1} \hat{B} \tag{19}$$

where, for each basis function $\eta_i \in \Phi_{hp}$ and $\zeta_i \in \Psi_{hp}$,

$$[\hat{A}_\alpha]_{ij} \coloneqq \alpha(\nabla_h \eta_i, \nabla_h \eta_j)_{L^2(\Omega)} \quad \text{and} \quad \hat{B}_{ij} \coloneqq (\eta_i, \zeta_j)_{L^2(\Omega)}. \tag{20}$$

$\hat{A}_\alpha$ is diagonal when $d = 1$ and block-diagonal (after a permutation of the rows and columns) with 4N blocks (where N is the number of 2D cells in the mesh) when $d = 2$. Thus $\hat{A}$ admits a cheap sparse Cholesky factorization. $\hat{S}$ is also block-diagonal although the blocks themselves are dense and of size $O(p^d \times p^d)$, $d \in \{1, 2\}$.

In Figure 2 we consider (15) as constructed for both a 1D and 2D discretization and measure the growth of $\hat{S}$-left-preconditioned GMRES iterations to solve $S\mathbf{x} = \mathbf{1}$ to a relative error of 10^{-6} with respect to partial degree p and $1/h$. We fix $\alpha = 1$, $\beta = 0$, and $D_\psi \equiv 0$ which constitutes the worst case scenario, i.e. where S is the most singular. In 1D, when h is fixed, we see that the GMRES iteration counts remain constant as $p \to \infty$. In all other cases, we observe polylogarithmic growth in the number of GMRES iterations. We also provide timings for the sparse LU factorization of $\hat{S}$.



(a) 1D: 10 cells (b) 1D: 10 cells (c) 1D: $p = 5$ (d) 1D: $p = 5$

(e) 2D: 16 quad. cells (f) 2D: 16 quad. cells (g) 2D: $p = 5$ (h) 2D: $p = 5$

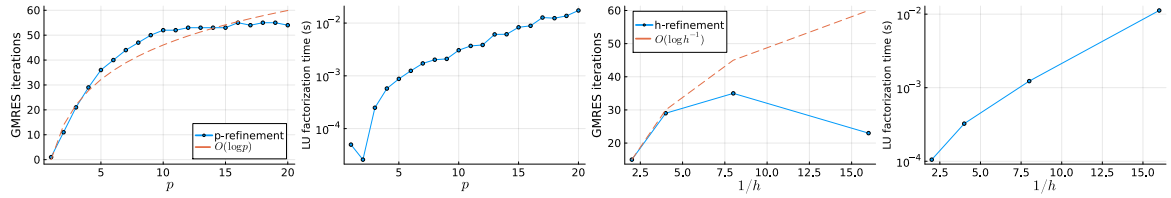
Figure 2: (Obstacle problem). We consider $d \in \{1, 2\}$ and fix $\alpha = 1$, $\beta = 0$, $D_\psi \equiv 0$ and consider the dense Schur complement S as defined (18a), in the context of the obstacle subproblem (4), and its sparse preconditioner $\hat{S}$ as defined in (19). We measure the growth in the number of $\hat{S}$-left-preconditioned GMRES iterations to solve $S\mathbf{x} = \mathbf{1}$ to a relative error of 10^{-6} (i.e. $\|\mathbf{r}_k\|_{\ell^2}/\|\mathbf{r}_0\|_{\ell^2} \leq 10^{-6}$) with respect to partial degree p in (a) and (e) and $1/h$ in (c) and (g). We observe no growth in 1D with respect to p and polylogarithmic growth in the other cases. We also measure the sparse LU factorization time for $\hat{S}$ with respect to partial degree p in (b) and (f) and $1/h$ in (d) and (h).

4.6 Gradient-type constraints: a Schur complement preconditioner

We choose a simpler preconditioner in the case of the pG gradient-type subproblem (5). More specifically, we drop the triple-matrix-product in S and choose:

$$\hat{S} = -D_\psi - E_\beta. \tag{21}$$

This provided robust preconditioning as evidenced in Figure 3. We perform the same experiment as for the obstacle-type case and examine the number of left-preconditioned GMRES iterations to solve $S\mathbf{x} = \mathbf{1}$ to a relative tolerance of 10^{-6} with respect to increasing p and $1/h$ in 2D. We fix $\alpha = 1$, $D_\psi \equiv 0$ but select $\beta = 10^{-5}$ as otherwise the Schur complement S is nearly singular.



(a) 16 quad. cells (b) 16 quad. cells (c) $p = 4$ (d) $p = 4$

Figure 3: (Gradient-type constraint). We consider $d = 2$ and fix $\alpha = 1$, $\beta = 10^{-5}$, $D_\psi \equiv 0$ and consider the dense Schur complement S as defined (18a), in the context of the pG gradient-type subproblem (5), and its sparse preconditioner $\hat{S}$ as defined in (21). We measure the growth in the number of $\hat{S}$-left-preconditioned GMRES iterations to solve $S\mathbf{x} = \mathbf{1}$ to a relative error of 10^{-6} with respect to partial degree p in (a) and $1/h$ in (c). We observe slower than logarithmic growth in both cases. We also measure the sparse LU factorization time for $\hat{S}$ with respect to partial degree p in (b) and $1/h$ in (d).

5 Adaptive hp-refinement

As the regularity of the solution is almost always capped at $H^s(\Omega)$, $s < 5/2$, a uniform p-refinement will often fail to reduce the error faster than h-refinement, particularly if the mesh is under-resolved in transition regions from obstacle contact to no contact. This is a well-studied phenomenon and estimators have been designed to guide hp-refinement. There are two questions to consider:

1 How does one estimate the error on each cell in the mesh?

2 How does one choose whether to h and/or p-refine a cell?

In this study, we focus exclusively on hp-adaptivity for the one-dimensional obstacle problem. Achieving a fast and efficient implementation in higher dimensions is intricate and falls outside the scope of this work. By design, p-adaptivity leads to a discretization where the degree p varies from cell to cell in the mesh. This may provide the smallest error per dof but there is an overhead associated with such an implementation. In particular certain speedups for the fast transforms in 2D are lost in an implementation that allows for varying degree and may lead to slower solve times. Moreover, the current tensor-product structure of the discretization for $d \geq 2$ introduces additional challenges. Our focus on the obstacle problem stems from the availability of the provably reliable and efficient hp-a posteriori error estimators developed by Banz and Schröder in [18, Sec. 4].

We first consider (2) and utilize the strategy as found in the work of Houston and Süli [52, Alg. 1]. Suppose that $d = 1$. Given a cell $K \in \mathcal{T}_h$ and the discrete solution u_{hp}, we recover the coefficients of the Legendre expansion on the cell K: $u_{hp}(x)|_K = \sum_{j=0}^{p+1} a_j P_j^K(x)$, where P_j^K denotes the Legendre polynomial of degree j affine-transformed to the cell K. In other words we re-expand u_{hp} in the L^2-conforming p-FEM space[4]. Next we find the ℓ^2-norm least squares solution (m, b) to $jm+b = |\log|a_j||$ for $j = 0, \ldots, p+1$. e^{-m} is a measurement of the analyticity of u_{hp} in K cf. [52, Sec. 2.4.1]. Thus for a user-chosen parameter $\sigma \in (0, 1)$, if $\mathrm{e}^{-m} < \sigma$, then we uniformly h-refine the cell and then p-refine the refined cells whereas if $\mathrm{e}^{-m} \geq \sigma$ then we only uniformly h-refine the cell K.

For (1) we found that the a posteriori estimate for the limit problem (3) provided the best error reduction.

[4]Practically these are easily found (without a need for quadrature) since there exists a banded matrix R with block bandwidths $(1, 1)$ such that $\mathbf{a} = R\mathbf{u}$ [60, Sec. 2.2.1].

Consider the local residuals for $K \in \mathcal{T}_h$:

$$\begin{aligned}\eta_1^2(K) &:= \frac{h_K^2}{p_K^2}\|f_{hp} + \Delta u_{hp} + \lambda_{hp}\|_{L^2(K)} + \frac{h_K^2}{p_K^2}\|f - f_{hp}\|_{L^2(K)}^2 + \|\varphi - \varphi_{hp}\|_{L^2(K)} \\ \eta^2(K) &:= \eta_1^2(K) + |(\lambda_{hp}, \varphi_{hp} - u_{hp})_{L^2(K)}| + \|\min\{\varphi_{hp} - u_{hp}, 0\}\|_{L^2(K)}^2,\end{aligned} \tag{22}$$

where φ_{hp} is the projection of φ in Ψ_{hp}, $\lambda_{hp} = (\psi_{k-1,hp} - \psi_{hp})/\alpha$ and $\psi_{k-1,hp}$ is the FEM latent variable solution at the previous hpG iterate $k-1$. By combining the local measurements for analyticity and error, we derive the hp-refinement algorithm in Algorithm 1.

Remark 5.1 (pG error indicator)**.** *We are the using a strong reformulation of the limit problem, the variational inequality* (3)*, to estimate the error on each cell. Instead one might argue that an error indicator derived from the pG subsystem might be more accurate, i.e.*

$$\eta^2(K) := \alpha^2\eta_1^2(K) + \|u_{hp} + \exp(-\psi_{hp}) - \varphi_{hp}\|_{L^2(K)}^2. \tag{23}$$

However, we found that for the examples considered in Section 7, the error indicator in (23) *performed worse than the one in* (22)*.*

Algorithm 1 hp-adaptivity

1: **Input.** $\mathcal{T}_h$, u_{hp}, λ_{hp}, local error estimator η, adaptivity parameters $\sigma \in (0,1)$, $\delta \in (0,1)$.
2: **Output.** A refined mesh $\mathcal{T}_{\tilde{h}}$ and associated degree p_K for each $K \in \mathcal{T}_{\tilde{h}}$.
3: For each $K \in \mathcal{T}_h$, compute the local error estimate $\eta(K)$.
4: Mark cells $\tilde{K} \subset \mathcal{N}$ where $\mathcal{N} = \{T \in \mathcal{T}_h : \eta(T) \geq \delta \max_{K\in\mathcal{T}_h} \eta(K)\}$.
5: **for** $K \in \mathcal{N}$ **do**
6: Compute the analyticity coefficient m_K.
7: **if** $\mathrm{e}^{-m_K} < \sigma$ **then** p-refine K **end**
8: h-refine K.
9: **end for**

6 Notes for fast implementation

Fast transforms. An essential aspect for fast solve times when handling high-degree polynomials is the ability to implement the analysis (expansion in the basis) and synthesis (evaluation of the basis on a grid) routines in quasi-optimal complexity. Such transforms exist for the L^2-conforming basis Ψ_{hp} consisting of scaled-and-shifted Legendre polynomials [3, 56, 92, 70]. In our implementation we opt for the approach described in [70, Sec. 4.6.4] and implemented in [87] where the coefficients of the Legendre expansion are converted to and from the coefficients of the equivalent Chebyshev expansion. The Chebyshev expansion then enjoys the use of the Discrete Cosine Transform (DCT) for quasi-optimal analysis and synthesis.

Alleviating ill-conditioning. As observed by Keith and Surowiec [57] the nonlinear convergence of the Newton solver may degrade as $h \to 0$ and $p \to \infty$. We hypothesize this is due to the ill-conditioning of the linear system in (15). We find that there are three remedies to such a situation:

- Utilization of the block preconditioning strategy in Section 4.4.

- Adding a small modification to the Jacobian, which we denoted by E_β in (15). The local modification also makes the GMRES solver more robust.
- Choosing small values for the α-sequence. Since the solver converges without requiring $\alpha \to \infty$, one may use small values of α, for instance terminating whilst $\alpha < 1$, and still observe numerical convergence.

Quadrature. Quadrature is required to assemble or apply the action of the matrix D_ψ. For instance, in the obstacle problem, given ψ_{hp} we must compute (I) $(\zeta_i, \mathrm{e}^{-\psi_{hp}})_{L^2(\Omega)}$ and (II) $(\zeta_i, \mathrm{e}^{-\psi_{hp}}\zeta_j)_{L^2(\Omega)}$ for each basis function $\zeta_i, \zeta_j \in \Psi_{hp}$. In our implementation we opted for a fast but non-standard quadrature. We consider the approximations $\mathrm{e}^{-\psi_{hp}} \approx \sum_k c_k \zeta_k$ for (I) and $\mathrm{e}^{-\psi_{hp}}\zeta_j \approx \sum_k \tilde{c}_k \zeta_k$ for (II). Once the expansions are performed, the integrals (I) and (II) are computed efficiently due to the orthogonality and support properties of the basis functions in Ψ_{hp}. The integrals are only exact if $\mathrm{e}^{-\psi_{hp}}$ is a piecewise constant. Nevertheless, experimentally the quadrature rule provided excellent results with the hpG solver exhibiting robustness against polynomial aliasing [54, Ch. 2.4.1.2]. This quadrature scheme does not preserve symmetry in D_ψ but when tested against a symmetry-conserving Clenshaw–Curtis quadrature scheme [33, 93] we found that the non-standard route allowed for quicker assembly of D_ψ and actually reduced the number of nonlinear and GMRES iterations required for the obstacle problem solve. We use the same technique for the action and assembly of D_ψ in the gradient-type case.

Inexact solves. Speedups are available if one does not solve the Newton linear systems exactly but rather terminates the linear system iterative solver once the error falls below a user-chosen tolerance.

7 Examples

Data availability. The implementation is contained in the package HierarchicalProximalGalerkin.jl [71] and is written in the open-source language Julia [22]. The version of HierarchicalProximalGalerkin.jl run in our experiments is archived on Zenodo [72]. We utilize a number of available Julia packages [81, 14, 39, 31, 55, 68, 87] and, in particular, the PiecewiseOrthogonalPolynomials.jl package [69] for its implementation of the hierarchical p-FEM basis. The experiments in Sections 7.1 to 7.3 were run on a deskstop with 16GB of RAM and 8 CPUs Intel(R) Core(TM) i7-10700 CPU @ 2.90GHz. The final example in Section 7.4 was run on a machine with 768GB of RAM and 72 CPUs HPE Synergy 660 Gen10 Xeon @ 3.1GHz.

7.1 Oscillatory data

The first example is a one-dimensional obstacle problem with an oscillatory right-hand side. Let $\omega = 10\pi$ and $c = 2\omega^2$ and consider the parameters:

$$\Omega = (0, 1), \quad f(x) = c\sin(\omega x), \quad \varphi \equiv 1. \tag{24}$$

The solution of the obstacle problem with the setup (24) is

$$u(x) = \begin{cases} 2\sin(\omega x) + a_0 x & \text{if } x \in [0, x_0), \\ 1 & \text{if } x \in [x_0, x_1), \\ 2\sin(\omega x) - 1 & \text{if } x \in [x_1, x_2), \\ 1 & \text{if } x \in [x_2, x_3), \\ 2\sin(\omega x) + a_1 x - a_1 & \text{if } x \in [x_3, 1], \end{cases} \tag{25}$$

where $x_0 \approx 0.038276$, $x_1 = 0.05$, $x_2 = 0.85$, $x_3 \approx 0.853423$, $a_0 \approx -22.621665$, and $a_1 \approx 6.743534$. The goal is to compare the hpG solver with the primal-dual active set strategy (PDAS) [46] and investigate the effectiveness of an adaptive hp-refinement. When using PDAS, u is discretized with continuous piecewise linear finite elements. The PDAS solver has no assembly cost at each nonlinear iteration and is effectively solving a Poisson equation over the inactive set of the domain at each iteration. However, each time the active set changes, the stiffness matrix must modified and re-factorized via a sparse Cholesky factorization. For the hpG solver, we use the α-update rule $\alpha_1 = 2^{-7}$, $\alpha_{k+1} = \min(\sqrt{2}\alpha_k, 2^{-3})$ and terminate once $\alpha_k = \alpha_{k-1} = 2^{-3}$. We fix the stabilization parameter $\beta = 10^{-8}$ in (15). The results are visualized in Figure 4.

We observe that the hp-refinement strategy of Algorithm 1, labelled "hp-adaptiveïn Figure 4, is very effective at error reduction. In fact, we discover a convergence rate of $O((h/p)^{10})$ for the regimes of h and p that we consider. At the final data point, we have an H^1-norm error of 5.61×10^{-5}, a minimum and maximum discretization degree of 13 and 18 and mesh sizes $h_{\text{min}} = 3.9 \times 10^{-4}$ and $h_{\text{max}} = 5 \times 10^{-2}$ across all the cells of the mesh, respectively. Moreover, the hpG iterations are bounded for all ten data points where the solves required 32, 33, 28, 30, 30, 32, 32, 32, 33, and 33 cumulative Newton iterations for each successive refinement, respectively. Recall that on each mesh, the initial guess is initialized as the zero function. Hence, these Newton iterations are *not* dependent on grid-sequencing from the discretized solution of the parent coarser mesh.

The strategy labelled "h-adaptive, p-uniformïn Figure 4 also provided a fast convergence rate of $O((h/p)^5)$. Here we use an adaptive h-refinement but coupled with an aggressive p-refinement where on the first mesh $p = 4$ and then on each successive mesh we increase the polynomial degree by one across all the cells on the mesh. The main advantage of such a strategy is that the transforms required for the assembly and action of nonlinear block D_ψ are parallelized more easily and lead to fast wall-clock solve times. In fact this strategy provided both the smallest H^1-norm error as measured by both the number of dofs and the time cost per linear solve, at least in the regime of h and p that we consider. The hpG solves were mesh independent requiring 22 Newton iterations on each refinement. The solve for the finest discretization took 0.087 seconds to achieve an H^1-norm error of 2.0×10^{-4}. Most notably, this strategy required half the average wall-clock time per Newton iteration, when compared to the PDAS strategy to achieve an H^1-norm error smaller than 10^{-2}.

The three strategies of uniform p-refinement (with a fixed h) and uniform h-refinement with either $p = 2$ or $p = 4$ all delivered a convergence rate of slightly faster than $O((h/p)^{3/2})$. This is expected behaviour due to the $H^s(\Omega)$, $s < 5/2$, regularity of the solution. In the p-refinement, the largest p considered is $p = 41$. As expected, both a uniform or adaptive h-refinement with $p = 1$ via the PDAS solver was capped at a convergence of $O(h)$.

Remark 7.1 (Comparing solver times)**.** *We solve the linear systems in the PDAS and hpG strategies via sparse direct solvers. The linear solves in the PDAS could be greatly accelerated, for example, by using algebraic multigrid techniques. However, we contend that a similar improvement is possible for the linear solves in the hpG solver via the preconditioning strategy of Section 4.4 and other*

similar strategies [13, 21, 61, 85]. To conclude, we are not claiming that a high-order discretization is always the most computationally efficient strategy in terms of error reduction. However, we have clear evidence to assert that they are certainly competitive.

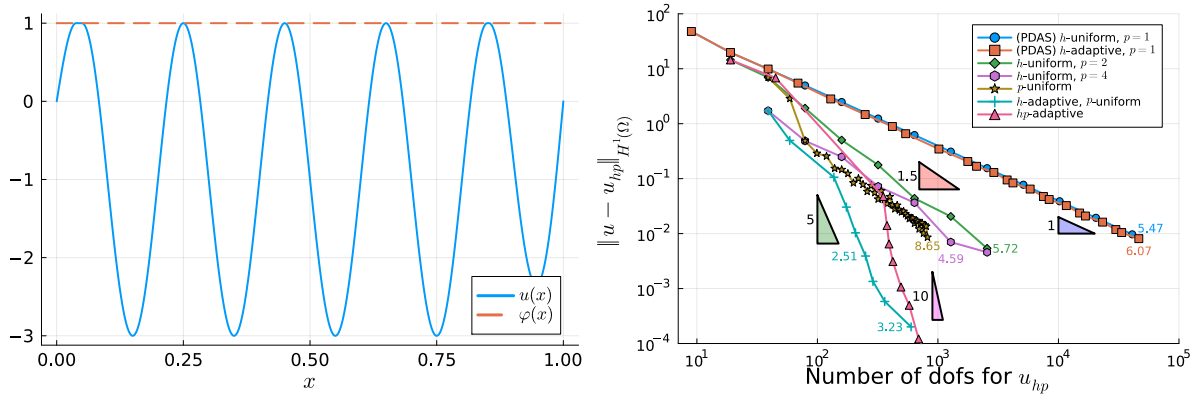


Figure 4: (Left) The solution to the 1D obstacle problem with the setup (24). (Right) Convergence of 7 refinement strategies. hp-adaptive utilizes Algorithm 1 ($\delta = 0.7$, $\sigma = 0.8$), h-adaptive is Algorithm 1 with $\sigma = 0$ ($\delta = 0.3$), h-uniform implies the mesh size is uniformly halved with each refinement and p-uniform is where the polynomial degree is incremented by one with each refinement. Any number associated with a data point is the average time taken per Newton iteration, measured in milliseconds, via a direct sparse factorization. In the order listed in the legend, the first two strategies converge at a rate of 1, the next 3 at a rate slightly faster than 3/2 and the final two at a rate of 5 and 10, respectively (in the regime of h and p considered).

7.2 Oscillatory obstacle

Consider the following setup of a two-dimensional obstacle problem:

$$\Omega = (0,1)^2, \quad f(x,y) = 100, \quad \text{and} \quad \varphi(x,y) = (1 + J_0(20x))(1 + J_0(20y)), \tag{26}$$

where J_0 denotes the zeroth order Bessel function of the first kind [67, Sec. 10.2(ii)]. We are not aware of the exact solution of this problem and, therefore, estimate the error against a heavily-refined discretization that is plotted in Figure 5. In this example we focus primarily on uniform h and p-refinements and discover that the best ratio of (approximate) error to computational expense is achieved by a hp-uniform refinement, i.e. with each refinement the mesh size is halved and the partial discretization degree is incremented by one. We believe that this subsection provides a clear counterexample to the notion that low-order discretizations should always be preferred.

We initialize the discretization by meshing the domain into 10×10 uniform quadrilateral cells and use the same α-update rule as in Section 7.1. We do not use any stabilization, i.e. $\beta = 0$ in (15). We find that all hpG runs required 24 Newton iterations independently of h and p. We plot the (approximate) H^1-norm error of the three strategies in Figure 6 and include the average time taken per linear solve for each solver (including any matrix assembly costs).

The convergence of the PDAS solver is capped at $O(h)$ since the partial degree is fixed at $p = 1$ whereas the other strategies observe an expected convergence rate of $O((h/p)^{3/2})$. Per dof, the high-order discretizations achieve the smallest error. Moreover, the average time taken per linear solve is clearly competitive with the low-order discretizations coupled with a finer mesh. In fact the best

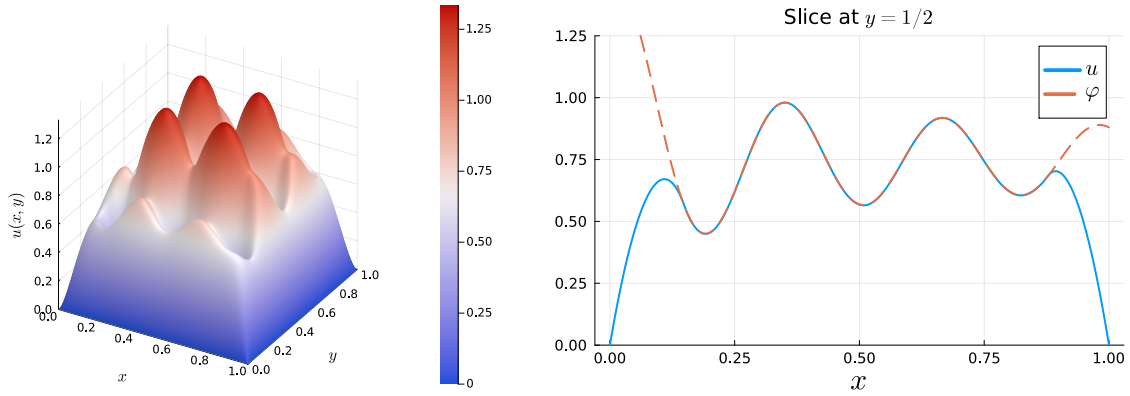


Figure 5: (Left) The solution to the obstacle problem with the setup (26). (Right) A 1D slice through the solution at $y = 1/2$.

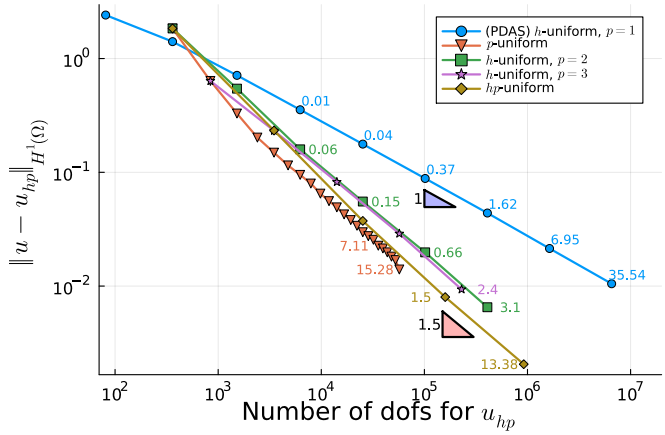


Figure 6: Approximate H^1-norm error of five strategies for the obstacle problem with the setup (26). All hpG solves required a total of 24 Newton iterations independently of p and h. The numbers attached to a data point are the average time taken per linear solve, as measured in seconds, via a sparse factorization. The higher order discretizations achieve a smaller error with fewer dofs and a faster solve time. The highest partial degree plotted is $p = 24$. The triangles indicate rates of convergence. When $p = 1$ we observe an $O(h)$ rate of convergence whereas for the other strategies the rate is roughly $O((h/p)^{3/2})$.

strategy, hp-uniform refinement, featured a discretization that resulted in wall-clock times that were roughly 24 times faster, per linear solve, than the PDAS strategy for an error of 10^{-2}.

To test the preconditioner constructed in Sections 4.4 and 4.5, in Table 1 we tabulate the average right-preconditioned GMRES iterations and wall-clock time per Newton step of the hpG solver for various choices of h and p. We pick $\beta = 10^{-4}$ in E_β in (15) and choose a relative stopping tolerance of 10^{-5} for the GMRES solver. For comparison, we include a row with the average wall-clock time per Newton step via a sparse LU factorization. The GMRES iterations are bounded with respect to p and only grow at a mild polylogarithmic rate as $h \to 0$. Moreover, the GMRES solver always resulted in faster average solve times per linear iteration than the LU factorization.

	$p = n, h = 1/10$				$p = 2, h = 2^{-n}/10$			$p = 3, h = 2^{-n}/10$			$p = n + 2, h = 2^{-n}/10$		
n	3	10	17	24	1	3	5	0	2	4	0	2	4
GMRES Its.	16.38	23.63	23.54	23.38	20.33	32.08	36.54	16.38	26.79	35.13	14.21	29.33	35.29
GMRES Time	0.00	0.19	1.67	7.99	0.00	0.12	2.64	0.00	0.06	1.58	0.00	0.23	12.07
LU Time	0.00	0.22	2.75	15.28	0.01	0.15	3.10	0.01	0.10	2.40	0.00	0.20	13.38

Table 1: A comparison of the average right-preconditioned GMRES iterations and wall-clock timings (in seconds) per Newton iteration for various choices of h and p over a run of the hpG algorithm to solve the obstacle problem with the setup (26). We utilize the preconditioner outlined in Sections 4.4 and 4.5 with a GMRES relative tolerance of 10^{-5}. We use the stabilization E_β with $\beta = 10^{-4}$ and include the average wall-clock timings of a sparse LU factorization for comparison. We observe a bounded iteration count with respect to p and a mild polylogarithmic growth with respect to h. The GMRES solver is almost always faster than the LU factorization.

7.3 A gradient-type constrained problem

Consider the gradient-type constrained Dirichlet minimization problem in (2b) and fix the parameters as

$$\Omega = (0,1)^2, \quad f(x,y) = 20, \quad \text{and} \quad \varphi(x,y) = \begin{cases} 1/2 & \text{if } x \text{ or } y \in [0,1/4] \cup [3/4,1], \\ \infty & \text{otherwise.} \end{cases} \tag{27}$$

In Figure 7 we plot the solution of the gradient-constrained problem with the setup (27). There is a kink in the solution at $x, y \in \{1/4, 3/4\}$ which coincides directly with the definition of φ. Once again, we are not aware of the exact solution of this problem and, therefore, we measure the convergence against a heavily-refined discretization. We initialize the discretization by meshing the domain into 4×4 uniform quadrilateral cells (or 8×8 in the p-uniform strategy) which aligns with the discontinuity of φ. We use the α-update rule $\alpha_1 = 2^{-7}$, $\alpha_{k+1} = \min(\sqrt{2}\alpha_k, 2^2)$ and terminate once $\alpha_k = \alpha_{k-1} = 2^2$. The hpG solver features hp-robustness with all hpG solves requiring between 74 and 78 Newton iterations irrespective of p and h. The convergence plot is also provided in Figure 7. Surprisingly, in the strategies that also refine the mesh, we observe regions with $O(h^p)$ convergence rates in contrast to the obstacle problems considered previously. We hypothesize that this superconvergence arises from the alignment of the mesh with the discontinuity in φ. In this example, this alignment coincides with the transition between the active and inactive regions of the solution which is where the loss of regularity occurs. Hence, the solution restricted to each cell K is more regular than $H^{5/2}(K)$ and, therefore, benefits from the high-order discretization. Once again, we observe significant speedups when utilizing discretizations $p \geq 2$. We observe that the induced linear systems are around 100 times faster to solve for the strategies where $p > 1$ in order to reach an error of 10^{-2}.

In Table 2 we report the effectiveness of the preconditioning strategy of Sections 4.4 and 4.6. The average GMRES iterations per linear solve are p-robust and grow with up to a linear rate with $1/h$. The growth is exacerbated by the choice of an adaptive stabilization parameter where $\beta = 10^{-p+\log_2 h} \to 0$ as $p \to \infty$ and $h \to 0$. If β is fixed to be constant, then the growth is at most polylogarithmic but causes the error to plateau. For all reported choices of h and p, we observe that the average wall-clock timing per linear solve of the preconditioned GMRES strategy is faster than a direct sparse LU factorization.

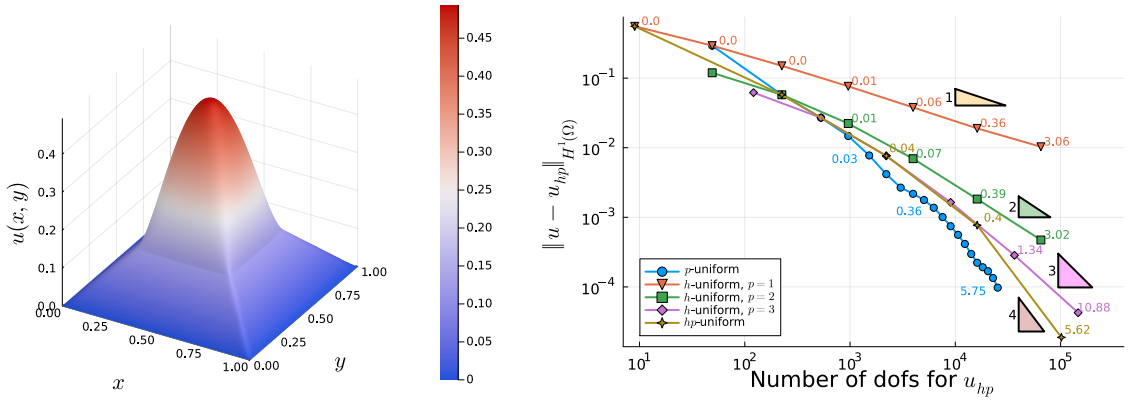


Figure 7: (Left) The solution to the gradient-constraint problem (2b) with the setup (27). (Right) Approximate H^1-norm error of five strategies. All hpG solves required between 74 and 78 Newton iterations. The numbers attached to data points are the average time taken per linear solve, as measured in seconds, with a sparse LU factorization. The highest partial degree plotted is $p = 20$. The triangles indicate rates of convergence. We observe $O(h^p)$ convergence in the range of h and p considered.

	$p = n, h = 2^{-3}$				$p = 1, h = 2^{-n}$			$p = 3, h = 2^{-n}$			$p = n - 1, h = 2^{-n}$	
n	5	10	15	20	4	6	8	3	5	7	4	6
GMRES Its.	61.76	71.76	72.03	72.00	10.31	20.73	33.73	21.26	50.71	72.76	34.73	77.48
GMRES Time	0.03	0.33	1.70	5.45	0.00	0.02	0.83	0.00	0.16	4.49	0.03	4.66
LU Time	0.03	0.36	2.60	5.75	0.00	0.06	3.06	0.01	0.19	10.88	0.04	5.62

Table 2: A comparison of the average right-preconditioned GMRES iterations and wall-clock timings (in seconds) per Newton iteration for various choices of h and p over a run of the hpG algorithm to solve (2b) with the setup (27). We utilize the preconditioning strategy outlined in Sections 4.4 and 4.6 with a GMRES relative tolerance of 10^{-3} and terminate GMRES if it reaches 150 iterations. We use the stabilization E_β with $\beta = 10^{-p+\log_2 h}$ and include the average wall-clock timings obtained with a sparse LU factorization for comparison. We observe that the iterations are p-robust and grow with up to a mild linear rate with $1/h$. The preconditioning strategy always offers a speedup when compared to a direct sparse LU factorization.

7.4 An obstacle-type quasi-variational inequality

In this example we consider the thermoforming *quasi-variational* inequality (QVI). In a QVI, the obstacle φ is dependent on the solution itself. Solvers for QVIs posed in an infinite-dimensional setting are scarce and most examples in the literature are solved by means of a fixed point iteration, penalty or augmented Lagrangian technique [53, 4]. Very recently a semismooth Newton method was introduced for a class of obstacle-type QVIs in [5]. The solver in [5] requires the realization of an active set and, therefore, is restricted to low-order FEM discretizations. The QVI can also be directly tackled by an extension of the pG algorithm [34, Sec. 3.5] but handling the nonlinear terms such that the discretizations remain sparse as $p \to \infty$ is nontrivial. Hence in this example we opt for a fixed point approach where the obstacle subproblems are solved via the hpG solver. We believe this is the highest order discretization of an elliptic obstacle-type QVI that is reported in the literature.

Given a $\Phi_0 \in H^1(\Omega)$, $\xi \in C^2(\bar{\Omega}) \cap H_0^1(\Omega)$, $f \in L^2(\Omega)$, $\gamma > 0$, and a globally Lipschitz and

nonincreasing function $g : H^1(\Omega) \to L^2(\Omega)$, the thermoforming problem is given by

$$\begin{aligned} &\text{Find } u \in H^1_0(\Omega) \text{ satisfying, for all } v \in \{w \in H^1_0(\Omega) : w \le \Phi(u) \coloneqq \Phi_0 + \xi T\}, \\ &\qquad u \le \Phi(u), \quad (\nabla u, \nabla(v-u))_{L^2(\Omega)} \ge (f, v-u)_{L^2(\Omega)}, \\ &\text{with } T \in H^1(\Omega) \text{ satisfying, for all } q \in H^1(\Omega), \\ &\qquad (\nabla T, \nabla q)_{L^2(\Omega)} + \gamma (T, q)_{L^2(\Omega)} = (g(\Phi_0 + \xi T - u), q)_{L^2(\Omega)}. \end{aligned} \tag{28}$$

Remark 7.2 (Thermoforming QVI)**.** *In two dimensions, $\Omega \subset \mathbb{R}^2$,* (28) *provides a simple model for the problem of determining the displacement $u \in H^1_0(\Omega)$ of an elastic membrane, clamped at the boundary $\partial\Omega$, that has been heated, and is pushed by means of an external force $f \in L^2(\Omega)$ into a metallic mould with original shape $\Phi_0 \in H^1(\Omega)$ and final shape $\Phi(u) \in H^1(\Omega)$. The deformation is due to the mould's temperature field $T \in H^1(\Omega)$ which varies according to the membrane's temperature. The heat transfer is modelled by a conduction coefficient $\gamma > 0$, a given globally Lipschitz and nonincreasing function $g : H^1(\Omega) \to L^2(\Omega)$, and a smoothing function $\xi \in H^1_0(\Omega) \cap C^2(\bar{\Omega})$ that incorporates the distance between the membrane and the mould. For more details on the thermoforming problem, including its derivation and its background, we refer to [4, Section 6] and the references therein.*

The fixed point approach proceeds as follows, for $T_0 \equiv 0$ and $i \in \mathbb{N}_0$, repeat the following two steps until convergence:

(I) Given a $T_i \in H^1(\Omega)$, solve the obstacle problem

$$\begin{aligned} &\text{find } u_{i+1} \in H^1_0(\Omega) \text{ satisfying for all } v \in \{w \in H^1_0(\Omega) : w \le \Phi_0 + \xi T_i \text{ a.e.}\} \\ &\qquad u_{i+1} \le \Phi_0 + \xi T_i, \quad (\nabla u_{i+1}, \nabla(v - u_{i+1}))_{L^2(\Omega)} \ge (f, v - u_{i+1})_{L^2(\Omega)}. \end{aligned} \tag{29}$$

(II) For all $q \in H^1(\Omega)$, solve the (nonlinear) PDE for $T_{i+1} \in H^1(\Omega)$:

$$(\nabla T_{i+1}, \nabla q)_{L^2(\Omega)} + \gamma (T_{i+1}, q)_{L^2(\Omega)} = (g(\Phi_0 + \xi T_{i+1} - u_{i+1}), q)_{L^2(\Omega)}. \tag{30}$$

We now fix the example parameters as:

$$\begin{aligned} &\Omega = (0,1)^2, \qquad f(x,y) = 100, \qquad \xi(x,y) = \sin(\pi x)\sin(\pi y), \qquad \gamma = 1, \\ &\Phi_0(x,y) = 11/10 - 2\max(|x-1/2|, |y-1/2|) + \cos(8\pi x)\cos(8\pi y)/10, \\ &\qquad g(s) = \begin{cases} 1/5 & \text{if } s \le 0, \\ (1-s)/5 & \text{if } 0 < s < 1, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{31}$$

The data (31) is such that we are guaranteed the existence and uniqueness of a solution to (28) [4]. Moreover, one can show there is a global contraction in the fixed point iteration induced by steps (I) and (II), cf. [5, Sec. 4.4] and, therefore, the scheme is guaranteed to converge to the (unique) solution. We plot the resulting membrane u and mould $\Phi_0 + \xi T$ in Figure 8 as well as a slice at $y = 1/2$.

We mesh the domain into a 4×4 uniform mesh (i.e. 16 cells total) and discretize (u, ψ, T) with $(u_{hp}, \psi_{hp}, T_{hp}) \in U_{h,\boldsymbol{p}_x,\boldsymbol{p}_y} \times \Psi_{h,\boldsymbol{p}_x-2,\boldsymbol{p}_y-2} \times U_{h,\boldsymbol{p}_x,\boldsymbol{p}_y}$. We terminate the fixed point algorithm once $\|u_i - u_{i-1}\|_{H^1(\Omega)} \le 3 \times 10^{-3}$.

To solve the obstacle problem in step (I), we use the hpG solver with the α-update rule $\alpha_1 = 2^{-6}$, $\alpha_{k+1} = 4\alpha_k$, and terminate once $\alpha_k = 1$. We use the stabilization E_β in (15) with $\beta = 10^{-6}$ and a GMRES relative and absolute stopping tolerance of 10^{-7}.

We found that an efficient nonlinear solver for step (II) was Newton's method where the linear systems are solved by GMRES left-preconditioned with $A + \gamma M_u$ where A and M_u are the stiffness and mass matrices for U_{hp}. Hence, the assembly of the (increasingly dense as $p \to \infty$) Jacobian is never required and the Jacobian-vector product can be computed in quasi-optimal complexity (similar to the discussion in Section 6). Since $A + \gamma M_u$ is very sparse, it can be factorized efficiently with a sparse Cholesky factorization (the factorization took 0.2 seconds with $p = 82$). We chose a relative stopping tolerance of 1.5×10^{-8} for the GMRES solver.

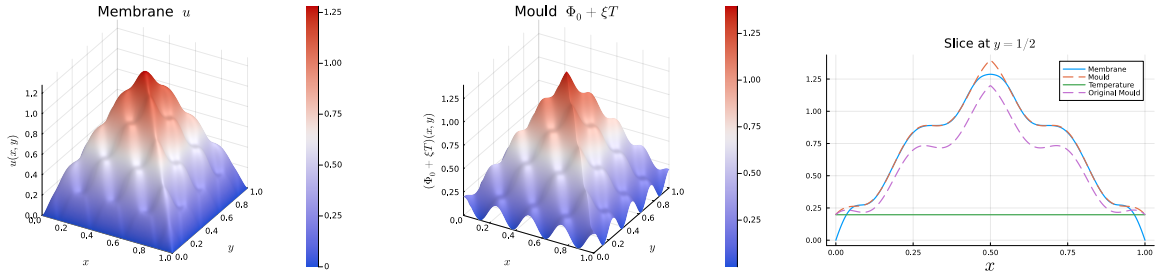


Figure 8: Surface plots of the membrane u (left) and the final mould (middle) of the thermoforming problem (28) with setup (31). On the right is a slice plot at $y = 1/2$ of the membrane and final mould as well as the original mould Φ_0 and final temperature T.

In Table 3 we provide the iteration counts of the outer fixed point method as well as the average number of Newton iterations per fixed point iteration and the average number of preconditioned GMRES iterations per Newton iteration for steps (I) and (II) with increasing partial degree p. We observe p-independent Newton iteration counts for both steps (I) and (II). The average number of GMRES iterations per Newton iteration appears to be bounded above by 22.4 and 3.11 for steps (I) and (II), respectively.

		Step (I)		Step (II)	
p	Fixed point	Avg. Newton	Avg. GMRES	Avg. Newton	Avg. GMRES
6	4	15.00	11.00	1.50	2.83
12	4	15.25	15.85	2.00	3.13
22	4	16.00	19.36	2.00	3.00
32	4	16.00	21.09	2.00	3.00
42	4	15.75	21.75	2.25	3.11
52	4	15.00	22.40	2.00	3.00
62	4	15.00	21.90	2.00	3.00
72	4	15.00	21.90	2.00	3.00
82	4	15.25	21.61	2.00	3.00

Table 3: The partial degree p, the number of outer iterations of the fixed point scheme as well as the average number of Newton iterations per fixed point iteration and average number of preconditioned GMRES iterations per Newton iteration in steps (I) and (II) to approximate the solution of the QVI (28) with setup (31). The algorithm terminates once $\|u_i - u_{i-1}\|_{H^1(\Omega)} \leq 3 \times 10^{-3}$. The outer fixed point loop is p-independent and we observe bounded iteration counts in the Newton and GMRES solvers for both steps (I) and (II).

8 Conclusions

In this paper we discretized the latent variable proximal point method [57, 34] with the hierarchical p-FEM basis [12, 85, 60] to construct the hierarchical proximal Galerkin algorithm, a high-order solver for variational problems with pointwise obstacle- and gradient-type inequality constraints. As $p \to \infty$, the choice of FEM basis retains sparsity in the discretized Newton systems and also admits a block preconditioner that, when coupled with a GMRES iterative solver, bounds the inner Krylov iterations with polylogarithmic growth in the worst case. Moreover, we observed that the outer Newton iterations are experimentally *mesh* and *degree independent*. The method is amenable to hp-adaptivity techniques which were fully explored in a one-dimensional obstacle problem example. We also successively apply the solver to a two-dimensional obstacle problem and gradient-constrained problem as well as a thermoforming problem, an example of an obstacle-type quasi-variational inequality. We consider discretizations with up to partial degree $p = 82$ on each element.

We also compare wall-clock timings of our solver with low-degree discretizations and note that, for the regimes of h and p considered, we found high-order discretizations deliver up to 100 times faster solves to achieve the same errors. This discovery is, perhaps, in contrast to what the community has come to expect.

We now outline some extensions:

- **Non-Cartesian cells.** The discretization relies on a tensor-product structure of the mesh whenever $d \geq 2$. For applications where non-Cartesian cells are required, we advocate that our solver be used as a preconditioner for the more general discretization via equivalent operator preconditioning [9]. We refer the interested reader to an excellent introduction by Brubeck and Farrell in [28, Sec. 2.7] who also prove their preconditioner is spectrally equivalent to the original problem.
- **hp-adaptivity when $d \geq 2$.** Similar to the case of non-Cartesian cells, the required tensor-product structure places a limit on a full implementation of hp-adaptivity whenever $d \geq 2$. In such a case, we once again recommend to use our solver as a preconditioner via equivalent operator preconditioning.
- **p-multigrid.** We believe an investigation into utilizing p-multigrid as a preconditioner (cf. [28]) is worthwhile and may allow one to consider significantly finer meshes and achieve inner Krylov iteration counts that are even more robust to p and h.
- **Alternative problems.** Minimizing the Dirichlet energy subject to obstacle- and gradient-type pointwise constraints constitutes the simplest PDE structure one might consider. The proximal Galerkin method has been shown to be a flexible framework, as evidenced in [34], and can be extended to tackle problems in contact mechanics, brittle fracture, multiphase species, quasi-variational inequalities, and the Monge–Ampère equation among others. The goal here would be to adapt the chosen high-order FEM discretization of the pG subproblems to also provide fast solves for these more general problems.

Acknowledgements

I would like to thank Michael Hintermüller, Brendan Keith, Patrick Farrell, Thomas Surowiec, Jørgen Dokken, Sheehan Olver, Richard M. Slevinsky, Jishnu Bhattacharya, Lothar Banz, Andreas Schröder,

and Pablo Brubeck for discussions and comments that have significantly aided with the writing of this manuscript.

References

[1] R. A. Adams and J. J. Fournier. *Sobolev spaces*. Second. Elsevier, 2003. ISBN: 978-0-12-044143-3.

[2] M. Ainsworth and C. Parker. “Unlocking the secrets of locking: Finite element analysis in planar linear elasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 395 (2022), p. 115034. DOI: 10.1016/j.cma.2022.115034.

[3] B. K. Alpert and V. Rokhlin. “A fast algorithm for the evaluation of Legendre expansions”. In: *SIAM Journal on Scientific and Statistical Computing* 12.1 (1991), pp. 158–179. DOI: 10.1137/0912009.

[4] A. Alphonse, M. Hintermüller, and C. N. Rautenberg. “Directional differentiability for elliptic quasi-variational inequalities of obstacle type”. In: *Calculus of Variations and Partial Differential Equations* 58.1 (2019), p. 39. DOI: 10.1007/s00526-018-1473-0.

[5] A. Alphonse et al. *A Globalized Inexact Semismooth Newton Method for Nonsmooth Fixed-point Equations involving Variational Inequalities*. 2024. arXiv: 2409.19637 [math.NA].

[6] J. Andrej et al. “High-performance finite elements with MFEM”. In: *The International Journal of High Performance Computing Applications* (), p. 10943420241261981. DOI: 10.1177/10943420241261981.

[7] H. Antil et al. “Nondiffusive variational problems with distributional and weak gradient constraints”. In: *Advances in Nonlinear Analysis* 11.1 (2022), pp. 1466–1495. DOI: 10.1515/anona-2022-0227.

[8] T. F. R. Arndt. *Variational and quasi-variational problems with gradient constraints and applications to sandpile growth*. George Mason University, 2021.

[9] O. Axelsson and J. Karátson. “Equivalent operator preconditioning for elliptic problems”. In: *Numerical Algorithms* 50 (2009), pp. 297–380. DOI: 10.1007/s11075-008-9233-4.

[10] I. Babuška and M. R. Dorr. “Error estimates for the combined h and p versions of the finite element method”. In: *Numerische Mathematik* 37 (1981), pp. 257–277. DOI: 10.1007/BF01398256.

[11] I. Babuška and B. A. Szabó. *Lecture notes on finite element analysis*. 1983–1985.

[12] I. Babuska, B. A. Szabo, and I. N. Katz. “The p-version of the finite element method”. In: *SIAM Journal on Numerical Analysis* 18.3 (1981), pp. 515–545. DOI: 10.1137/0718033.

[13] I. Babuška et al. “Efficient preconditioning for the p-version finite element method in two dimensions”. In: *SIAM Journal on Numerical Analysis* 28.3 (1991), pp. 624–661. DOI: 10.1137/0728034.

[14] S. Badia and F. Verdugo. “Gridap: An extensible Finite Element toolbox in Julia”. In: *Journal of Open Source Software* 5.52 (2020), p. 2520. DOI: 10.21105/joss.02520.

[15] P. Bammer, L. Banz, and A. Schröder. *A Posteriori Error Estimates for hp-FE Discretizations in Elastoplasticity*. 2024. arXiv: 2401.09105 [math.NA].

[16] L. Banz, M. Hintermüller, and A. Schröder. “A posteriori error control for distributed elliptic optimal control problems with control constraints discretized by hp-finite elements”. In: *Computers & Mathematics with Applications* 80.11 (2020), pp. 2433–2450. DOI: 10.1016/j.camwa.2020.08.007.

[17] L. Banz and A. Schröder. “A posteriori error control for variational inequalities with linear constraints in an abstract framework”. In: *Journal of Applied & Numerical Optimization* 3.2 (2021). DOI: 10.23952/jano.3.2021.2.07.

[18] L. Banz and A. Schröder. “Biorthogonal basis functions in hp-adaptive FEM for elliptic obstacle problems”. In: *Computers & Mathematics with Applications* 70.8 (2015), pp. 1721–1742. DOI: 10.1016/j.camwa.2015.07.010.

[19] L. Banz and E. P. Stephan. “A posteriori error estimates of hp-adaptive IPDG-FEM for elliptic obstacle problems”. In: *Applied Numerical Mathematics* 76 (2014), pp. 76–92. DOI: 10.1016/j.apnum.2013.10.004.

[20] M. P. Bendsøe and O. Sigmund. *Topology Optimization*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. ISBN: 978-3-642-07698-5. DOI: 10.1007/978-3-662-05086-6.

[21] S. Beuchler and D. Braess. “Improvements for some condition number estimates for preconditioned system in p-FEM”. In: *Numerical Linear Algebra with Applications* 13.7 (2006), pp. 573–588. DOI: 10.1002/nla.489.

[22] J. Bezanson et al. “Julia: A fresh approach to numerical computing”. In: *SIAM Review* 59.1 (2017), pp. 65–98. DOI: 10.1137/141000671.

[23] T. Borrvall and J. Petersson. “Topology optimization of fluids in Stokes flow”. In: *International Journal for Numerical Methods in Fluids* 41.1 (2003), pp. 77–107. DOI: 10.1002/fld.426.

[24] A. Brandt and C. W. Cryer. “Multigrid algorithms for the solution of linear complementarity problems arising from free boundary problems”. In: *SIAM Journal on Scientific and Statistical Computing* 4.4 (1983), pp. 655–684. DOI: 10.1137/0904046.

[25] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. 3rd ed. Vol. 15. Texts in Applied Mathematics. New York, NY: Springer New York, 2008. ISBN: 978-0-387-75933-3. DOI: 10.1007/978-0-387-75934-0.

[26] H. Brezis and M. Sibony. “Equivalence de deux inéquations variationnelles et applications”. In: *Archive for Rational Mechanics and Analysis* 41.4 (Jan. 1971), pp. 254–265. DOI: 10.1007/bf00250529.

[27] H. Brézis. “Nouveaux théorèmes de régularité pour les problèmes unilatéraux”. In: *Les rencontres physiciens-mathématiciens de Strasbourg-RCP25* 12 (1971), pp. 1–14.

[28] P. D. Brubeck and P. E. Farrell. “A Scalable and Robust Vertex-Star Relaxation for High-Order FEM”. In: *SIAM Journal on Scientific Computing* 44.5 (2022), A2991–A3017. DOI: 10.1137/21M1444187.

[29] E. Bueler and P. E. Farrell. “A full approximation scheme multilevel method for nonlinear variational inequalities”. In: *SIAM Journal on Scientific Computing* 46.4 (2024), A2421–A2444. DOI: 10.1137/23M1594200.

[30] L. A. Caffarelli and A. Friedman. “The free boundary for elastic-plastic torsion problems”. In: *Transactions of the American Mathematical Society* 252 (1979), pp. 65–97. DOI: 10.1090/S0002-9947-1979-0534111-0.

[31] J. Chen, H. Stoppels, and others. *IterativeSolvers.jl*. Version 0.9.4. 2024. URL: https://github.com/JuliaLinearAlgebra/IterativeSolvers.jl.

[32] C. Christof and G. Wachsmuth. “On second-order optimality conditions for optimal control problems governed by the obstacle problem”. In: *Optimization* 70.10 (2021), pp. 2247–2287. DOI: 10.1080/02331934.2020.1778686.

[33] C. W. Clenshaw and A. R. Curtis. “A method for numerical integration on an automatic computer”. In: *Numerische Mathematik* 2 (1960), pp. 197–205. DOI: 10.1007/BF01386223.

[34] J. S. Dokken et al. *The latent variable proximal point algorithm for variational problems with inequality constraints*. 2025. arXiv: 2503.05672 [math.OC].

[35] L. C. Evans. “A second order elliptic equation with gradient constraint”. In: *Communications in Partial Differential Equations* 4.5 (1979), pp. 555–572. DOI: 10.1080/03605307908820103.

[36] L. C. Evans. *Partial Differential Equations*. 2nd ed. American Mathematical Society, 2010. ISBN: 978-0821849743.

[37] G. Fichera. *Problemi elastostatici con vincoli unilaterali: il problema di Signorini con ambigue condizioni al contorno*. Accademia nazionale dei Lincei, 1964.

[38] D. Fortunato and A. Townsend. “Fast Poisson solvers for spectral methods”. In: *IMA Journal of Numerical Analysis* 40.3 (2020), pp. 1994–2018. DOI: 10.1093/imanum/drz034.

[39] J. Fuhrmann, J. J. Taraz, and others. *ExtendableSparse.jl*. Version 1.5.1. 2024. URL: https://github.com/j-fu/ExtendableSparse.jl.

[40] E. Gagliardo. “Caratterizzazioni delle tracce sulla frontiera relative ad alcune classi di funzioni in n variabili”. In: *Rendiconti del seminario matematico della universita di Padova* 27 (1957), pp. 284–305.

[41] C. Gräser and R. Kornhuber. “Multigrid methods for obstacle problems”. In: *Journal of Computational Mathematics* (2009), pp. 1–44.

[42] T. Gustafsson, R. Stenberg, and J. Videman. “On finite element formulations for the obstacle problem–mixed and stabilised methods”. In: *Computational Methods in Applied Mathematics* 17.3 (2017), pp. 413–429. DOI: 10.1515/cmam-2017-0011.

[43] J. Gwinner. “hp-FEM convergence for unilateral contact problems with Tresca friction in plane linear elastostatics”. In: *Journal of computational and applied mathematics* 254 (2013), pp. 175–184. DOI: 10.1016/j.cam.2013.03.013.

[44] W. Hackbusch and H. D. Mittelmann. “On multi-grid methods for variational inequalities”. In: *Numerische Mathematik* 42 (1983), pp. 65–76. DOI: 10.1007/BF01400918.

[45] M. Hintermüller and M. Hinze. “Moreau–Yosida regularization in state constrained elliptic control problems: error estimates and parameter adjustment”. In: *SIAM Journal on Numerical Analysis* 47.3 (2009), pp. 1666–1683. DOI: 10.1137/080718735.

[46] M. Hintermüller, K. Ito, and K. Kunisch. “The primal-dual active set strategy as a semismooth Newton method”. In: *SIAM Journal on Optimization* 13.3 (2002), pp. 865–888. DOI: 10.1137/S1052623401383558.

[47] M. Hintermüller and K. Kunisch. “Feasible and noninterior path-following in constrained minimization with low multiplier regularity”. In: *SIAM Journal on Control and Optimization* 45.4 (2006), pp. 1198–1221. DOI: 10.1137/050637480.

[48] M. Hintermüller and M. Ulbrich. “A mesh-independence result for semismooth Newton methods”. In: *Mathematical Programming* 101 (2004), pp. 151–184. DOI: 10.1007/s10107-004-0540-9.

[49] I. Hlavácek et al. *Solution of variational inequalities in mechanics*. Vol. 66. Springer Science & Business Media, 1988. ISBN: 978-0-387-96597-0.

[50] R. H. Hoppe. “Multigrid algorithms for variational inequalities”. In: *SIAM Journal on Numerical Analysis* 24.5 (1987), pp. 1046–1065. DOI: 10.1137/0724069.

[51] R. H. Hoppe and R. Kornhuber. “Adaptive multilevel methods for obstacle problems”. In: *SIAM Journal on Numerical Analysis* 31.2 (1994), pp. 301–323. DOI: 10.1137/0731016.

[52] P. Houston and E. Süli. “A note on the design of hp-adaptive finite element methods for elliptic partial differential equations”. In: *Computer Methods in Applied Mechanics and Engineering* 194.2-5 (2005), pp. 229–243. DOI: 10.1016/j.cma.2004.04.009.

[53] C. Kanzow and D. Steck. “Quasi-variational inequalities in Banach spaces: theory and augmented Lagrangian methods”. In: *SIAM Journal on Optimization* 29.4 (2019), pp. 3174–3200. DOI: 10.1137/18M1230475.

[54] G. Karniadakis and S. Sherwin. *Spectral/hp element methods for computational fluid dynamics*. Oxford University Press on Demand, 2005. ISBN: 9780198528692. DOI: 10.1093/acprof:oso/9780198528692.001.0001.

[55] D. Karrasch et al. *LinearMaps.jl*. Version 3.11.3. 2024. URL: https://github.com/JuliaLinearAlgebra/LinearMaps.jl.

[56] J. Keiner. *Fast Polynomial Transforms*. Logos Verlag Berlin GmbH, 2011. ISBN: 978-3-8325-2850-8.

[57] B. Keith and T. M. Surowiec. “Proximal Galerkin: A structure-preserving finite element method for pointwise bound constraints”. In: *Foundations of Computational Mathematics* (2024), pp. 1–97. DOI: 10.1007/s10208-024-09681-8.

[58] D. Kinderlehrer and G. Stampacchia. *An introduction to variational inequalities and their applications*. SIAM, 2000. ISBN: 978-0-89871-466-1. DOI: 10.1137/1.9780898719451.

[59] R. C. Kirby and D. Shapero. “High-order bounds-satisfying approximation of partial differential equations via finite element variational inequalities”. In: *Numerische Mathematik* (2024), pp. 1–21. DOI: 10.1007/s00211-024-01405-y.

[60] K. Knook, S. Olver, and I. P. A. Papadopoulos. *Quasi-optimal complexity hp-FEM for Poisson on a rectangle*. 2024. arXiv: 2402.11299 [math.NA].

[61] V. G. Korneev and S. Jensen. “Domain decomposition preconditioning in the hierarchical p-version of the finite element method”. In: *Applied Numerical Mathematics* 29.4 (1999), pp. 479–518. DOI: 10.1016/S0168-9274(98)00077-4.

[62] R. Kornhuber. “Monotone multigrid methods for elliptic variational inequalities I”. In: *Numerische Mathematik* 69 (1994), pp. 167–184. DOI: 10.1007/BF03325426.

[63] R. Kornhuber. “Monotone multigrid methods for elliptic variational inequalities II”. In: *Numerische Mathematik* 72.4 (1996), pp. 481–499. DOI: 10.1007/s002110050178.

[64] A. Krebs and E. P. Stephan. “A p-version finite element method for nonlinear elliptic variational inequalities in 2D”. In: *Numerische Mathematik* 105 (2007), pp. 457–480. DOI: 10.1007/s00211-006-0035-0.

[65] M. Kunze and J.-F. Rodrigues. “An elliptic quasi-variational inequality with gradient constraints and some of its applications”. In: *Mathematical Methods in the Applied Sciences* 23.10 (2000), pp. 897–908. DOI: 10.1002/1099-1476(20000710)23:10<3C897::AID-MMA141>3E3.0.CO;2-H.

[66] J.-L. Lions and G. Stampacchia. “Variational inequalities”. In: *Communications on Pure and Applied Mathematics* 20.3 (1967), pp. 493–519. DOI: 10.1002/cpa.3160200302.

[67] F. W. J. Olver et al. https://dlmf.nist.gov/, Release 1.2.1 of 2024-06-15. URL: https://dlmf.nist.gov/.

[68] S. Olver, J. Bhattacharya, and others. *MatrixFactorizations.jl*. Version 3.0.1. 2024. URL: https://github.com/JuliaLinearAlgebra/MatrixFactorizations.jl.

[69] S. Olver and I. P. A. Papadopoulos. *PiecewiseOrthogonalPolynomials.jl*. Version 0.5.0. 2024. URL: https://github.com/JuliaApproximation/PiecewiseOrthogonalPolynomia
jl.

[70] S. Olver, R. M. Slevinsky, and A. Townsend. “Fast algorithms using orthogonal polynomials”. In: *Acta Numerica* 29 (2020), pp. 573–699. DOI: 10.1017/S0962492920000045.

[71] I. P. A. Papadopoulos. *HierarchicalProximalGalerkin.jl*. Version 0.0.1. 2024. URL: https://github.com/ioannisPApapadopoulos/HierarchicalProximalGalerkin.jl.

[72] I. P. A. Papadopoulos. *HierarchicalProximalGalerkin.jl, v0.0.1 (Zenodo)*. Version v0.0.1. 2024. DOI: 10.5281/zenodo.14499625.

[73] I. P. A. Papadopoulos. “Numerical analysis of a discontinuous Galerkin method for the Borrvall–Petersson topology optimization problem”. In: *SIAM Journal on Numerical Analysis* 60.5 (2022), pp. 2538–2564. DOI: 10.1137/21M1438943.

[74] I. P. A. Papadopoulos. “Numerical analysis of the SIMP model for the topology optimization problem of minimizing compliance in linear elasticity”. In: *Numerische Mathematik* (2024), pp. 1–36. DOI: 10.1007/s00211-024-01438-3.

[75] I. P. A. Papadopoulos, P. E. Farrell, and T. M. Surowiec. “Computing multiple solutions of topology optimization problems”. In: *SIAM Journal on Scientific Computing* 43.3 (2021), A1555–A1582. DOI: 10.1137/20M1326209.

[76] I. P. A. Papadopoulos and S. Olver. *A sparse hierarchical hp-finite element method on disks and annuli*. 2024. arXiv: 2402.12831 [math.NA].

[77] I. P. A. Papadopoulos and E. Süli. “Numerical analysis of a topology optimization problem for Stokes flow”. In: *Journal of Computational and Applied Mathematics* 412 (2022), p. 114295. DOI: 10.1016/j.cam.2022.114295.

[78] K. Porwal and R. Singla. “A Posteriori Error Analysis of Hybrid High-Order Methods for the Elliptic Obstacle Problem”. In: *Journal of Scientific Computing* 102.1 (2025), pp. 1–20. DOI: 10.1007/s10915-024-02744-6.

[79] L. Prigozhin. “On the Bean critical-state model in superconductivity”. In: *European Journal of Applied Mathematics* 7.3 (1996), 237–247. DOI: 10.1017/S0956792500002333.

[80] L. Prigozhin. “Variational model of sandpile growth”. In: *European Journal of Applied Mathematics* 7.3 (1996), 225–235. DOI: 10.1017/S0956792500002321.

[81] A. N. Riseth, P. K. Mogensen, et al. *LineSearches.jl*. Version 7.3.0. 2024. URL: https://github.com/JuliaNLSolvers/LineSearches.jl.

[82] J.-F. Rodrigues. *Obstacle problems in mathematical physics*. Elsevier, 1987. ISBN: 978-0444701879.

[83] J. F. Rodrigues and L. Santos. “Variational and quasi-variational inequalities with gradient type constraints”. In: *Topics in Applied Analysis and Optimisation: Partial Differential Equations, Stochastic and Numerical Analysis*. Springer. 2019, pp. 319–361.

[84] L. Santos. “Variational problems with non-constant gradient constraints”. In: *Portugaliae Mathematica, Nova Séri* 59.2 (2002), pp. 205–248. URL: http://dml.mathdoc.fr/item/01782991.

[85] C. Schwab. *p-and hp-finite element methods: Theory and applications in solid and fluid mechanics*. Clarendon Press, 1998. ISBN: 9780198503903.

[86] J. Shen. “Efficient spectral-Galerkin method I. Direct solvers of second-and fourth-order equations using Legendre polynomials”. In: *SIAM Journal on Scientific Computing* 15.6 (1994), pp. 1489–1505. DOI: 10.1137/0915089.

[87] R. M. Slevinsky, S. Olver, and others. *FastTransforms.jl*. Version 0.16.4. 2024. URL: https://github.com/JuliaApproximation/FastTransforms.jl.

[88] G. Stampacchia. “Formes bilineaires coercitives sur les ensembles convexes”. In: *Comptes Rendus Hebdomadaires Des Seances De L Academie Des Sciences* 258.18 (1964), p. 4413.

[89] B. Szabó and I. Babuška. *Introduction to finite element analysis: formulation, verification and validation*. Vol. 35. John Wiley & Sons, 2011. ISBN: 9780470977286.

[90] T. W. Ting. “Elastic-plastic Torsion of Convex Cylindrical Bars”. In: *Journal of Mathematics and Mechanics* 19.6 (1969), pp. 531–551.

[91] T. W. Ting. “Elastic-plastic torsion problem over multiply connected domains”. In: *Annali della Scuola Normale Superiore di Pisa-Classe di Scienze* 4.2 (1977), pp. 291–312.

[92] A. Townsend, M. Webb, and S. Olver. “Fast polynomial transforms based on Toeplitz and Hankel matrices”. In: *Mathematics of Computation* 87.312 (2018), pp. 1913–1934. DOI: 10.1090/mcom/3277.

[93] L. N. Trefethen. “Is Gauss quadrature better than Clenshaw–Curtis?” In: *SIAM Review* 50.1 (2008), pp. 67–87. DOI: 10.1137/060659831.

[94] L. W. White. “Control of a hyperbolic problem with pointwise stress constraints”. In: *Journal of Optimization Theory and Applications* 41.2 (Oct. 1983), 359–369. DOI: 10.1007/bf00935231.

[95] B. I. Wohlmuth and R. H. Krause. “Monotone multigrid methods on nonmatching grids for nonlinear multibody contact problems”. In: *SIAM Journal on Scientific Computing* 25.1 (2003), pp. 324–347. DOI: 10.1137/S1064827502405318.