Accepted Manuscript

A hybridizable discontinuous Galerkin method for both thin and 3D nonlinear elastic structures

S. Terrana, N.C. Nguyen, J. Bonet, J. Peraire

PII: S0045-7825(19)30232-4
DOI: https://doi.org/10.1016/j.cma.2019.04.029
Reference: CMA 12417

To appear in: *Comput. Methods Appl. Mech. Engrg.*

Received date : 23 December 2018
Revised date : 19 April 2019
Accepted date : 24 April 2019

Please cite this article as: S. Terrana, N.C. Nguyen, J. Bonet et al., A hybridizable discontinuous Galerkin method for both thin and 3D nonlinear elastic structures, *Computer Methods in Applied Mechanics and Engineering* (2019), https://doi.org/10.1016/j.cma.2019.04.029

Highlights (for review)

- We present a hybridizable discontinuous Galerkin (HDG) method for thin and thick structures at finite deformations
- A technique of elimination of unknowns for thin structures significantly reduces the computational cost
- We present an empirical penalization that both alleviates the locking effects and stabilizes the HDG method
- The optimal convergence is achieved for the displacement and an extra half-order of convergence can be gained with an inexpensive postprocessing
- The method gives accurate results for various classical nonlinear shell problems

A hybridizable discontinuous Galerkin method for both thin and 3D nonlinear elastic structures

S. Terrana[a,*], N.C. Nguyen[a], J. Bonet[b], J. Peraire[a]

[a] *Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA 02139, USA*
[b] *University of Greenwich, London, SE10 9LS, United Kingdom*

Abstract

We present a 3D hybridizable discontinuous Galerkin (HDG) method for nonlinear elasticity which can be efficiently used for thin structures with large deformation. The HDG method is developed for a three-field formulation of nonlinear elasticity and is endowed with a number of attractive features that make it ideally suited for thin structures. Regarding robustness, the method avoids a variety of locking phenomena such as membrane locking, shear locking, and volumetric locking. Regarding accuracy, the method yields *optimal convergence* for the displacements, which can be further improved by an inexpensive postprocessing. And finally, regarding efficiency, the only globally coupled unknowns are the degrees of freedom of the numerical trace on the *interior faces*, resulting in substantial savings in computational time and memory storage. This last feature is particularly advantageous for thin structures because the number of interior faces is typically small. In addition, we discuss the implementation of the HDG method with arc-length algorithms for phenomena such as snap-through, where the standard load incrementation algorithm becomes unstable. Numerical results are presented to verify the convergence and demonstrate the performance of the HDG method through simple analytical and popular benchmark problems in the literature.

Keywords: Shell structures, Discontinuous Galerkin method, Nonlinear

*Corresponding author
Email address: terrana@mit.edu (S. Terrana)

elasticity, Superconvergence, Finite element, Hybridizable discontinuous Galerkin

1. Introduction

The discontinuous Galerkin (DG) method has been proven to be a valuable and versatile tool for numerical analysis in continuum mechanics, see e.g. [1, 2, 3]. In solid mechanics, DG methods have been proposed for linear elasticity (see [4] among many others), nonlinear hyperelasticity [5, 6, 7], as well as plasticity [8, 9].

However, DG methods have been often criticized for having to employ significantly more degrees of freedom than more standard continuous Galerkin (CG) methods. Hybridizable discontinuous Galerkin (HDG) methods have subsequently been developed to address this drawback. The advantage of HDG methods is twofold. First, HDG methods parametrize the finite element solution in terms of an approximation of the displacement on the element boundaries, the so-called *hybrid* field. Therefore, the only globally coupled unknowns are those corresponding to the hybrid field, which is unique for the two elements sharing a boundary. As a consequence, the global linear system to be solved is smaller than that obtained with the original DG degrees of freedom (DoF). Second, when polynomials of degree k are used to approximate both the displacement and its gradient, both approximations converge with the optimal order $k+1$. Then, provided $k \geq 1$, an elementwise post-processing step can be performed to obtain a superconvergent solution of order $k+2$ for the displacement. HDG approaches have been developed for both linear elasticity [10, 11, 12, 13, 14] and nonlinear elasticity [15, 11, 16, 17, 18]. Although the superconvergence have been often observed with these HDG approaches, it is not guaranteed in general for elasticity [13]. This paper proposes an extension of the HDG volumetric methods for nonlinear elasticity to thin structures.

Traditionally, thin structures have been modeled using special elements, as it is well known that classical low order finite elements fail to model such structures, due to several locking effects. Special plate and shell finite elements has been developed over the past fifty years, either based on a plate/shell theories, or by simplifying three-dimensional continuum theories (see reviews in [19, 20, 21], and in particular [22, 23, 24] for DG approaches). Both approaches have been shown to model accurately finite deformations of thin structures. However, they suffer from common disadvantages and difficulties. Among these difficulties are the coupling with solid finite elements (rotational degrees of freedom have to be connected with the solid element displacements using special transition elements), the application of particular boundary conditions, the complex updates of rotation vectors for large deformations, and the difficult degeneration of full 3D constitutive laws. Finally, the objectivity of the strain measures may be lost (see [25]). In order to overcome these drawbacks, alternative low-order solid-shell elements have been then developed (see [26, 27, 28, 29, 30] among many others), able to model both thick and thin structures. These elements are volumetric solid bricks modified with a variety of techniques in order to tame the locking pathologies. Among these techniques, the reduced integration [31, 32, 33] and the B-bar [34, 35] approaches address mainly the volumetric locking. The enhanced strain technique prevents volumetric [36] and membrane lockings [37] – see also [38, 29, 39, 40]. And coming from plate [41] and shell elements [42], the assumed natural strain technique can control the shear locking of solid shell elements [28, 27, 43, 39].

The present approach is different and, in many respects, simpler. We directly discretize the thin structures with high-order three dimensional elements and employ a nonlinear elasticity HDG volumetric formulation. This approach is motivated by the following observations. As a high-order finite element approach, all the thickness-related locking behaviors should vanish for high enough polynomial degrees [44, 45, 46]. Moreover, even for moderate polynomial degrees, the discontinuous nature of the approximations mitigates the locking effects, as previously observed for both beams [47] and shells [22]. In particular, as a discontinuous Galerkin approach, our method is free from volumetric locking for nearly-incompressible materials [4]. Finally, in our method, the only globally coupled unknowns are those representing the hybrid field which is defined on the *interior faces* only. This leads to substantial savings in computational time and memory storage for thin structures because the number of interior faces is small.

The article is organized as follows. In Section 2, we introduce the notations used throughout the paper. In Section 3, we introduce the HDG method based on a new variational principle. In Section 4, we discuss the implementation of the HDG method together with loading incrementation, Newton-Raphson, and Arc-Length algorithms. In particular, we explain how to take advantage of the discontinuity of the hybrid field to significantly reduce the size of the global linear system when thin structures are considered. In Section 5, we present numerical results to assess the convergence of the HDG method and its accuracy on several classical non-linear shells benchmark problems. Finally, in Section 6, we provide some concluding remarks.

2. Governing equations and notations

2.1. Nonlinear elasticity equations

We consider a deformable elastic body, occupying the volume $\Omega \in \mathbb{R}^d$ in the initial, undeformed configuration. The initial configuration Ω is assumed to be an open and bounded polygonal domain with a Lipschitz continuous boundary $\partial\Omega$. This boundary is divided into a Dirichlet boundary Γ_D and a Neumann boundary Γ_D such that $\partial\Omega = \overline{\Gamma_D \cup \Gamma_N}$ and $\Gamma_D \cap \Gamma_N = \emptyset$. The material position vector is denoted $\boldsymbol{\varphi}(\boldsymbol{X})$, with $\boldsymbol{X}$ denoting the reference material coordinates. Under given body forces $\boldsymbol{f}$, prescribed tractions $\boldsymbol{t}$ on Γ_N, and prescribed displacements $\boldsymbol{\varphi}_D$ on Γ_D, the elastic body undergoes a deformation satisfying the following static equilibrium equations

$$
\begin{aligned}
-\nabla \cdot \boldsymbol{P} &= \boldsymbol{f} && \text{in } \Omega && \text{(1a)}\\
\boldsymbol{F} - \nabla \boldsymbol{\varphi} &= 0 && \text{in } \Omega, && \text{(1b)}\\
\boldsymbol{P} - \frac{\partial \Psi}{\partial \boldsymbol{F}} &= 0 && \text{in } \Omega, && \text{(1c)}\\
\boldsymbol{\varphi} &= \boldsymbol{\varphi}_D, && \text{on } \Gamma_D, && \text{(1d)}\\
\boldsymbol{P}\boldsymbol{N} &= \boldsymbol{t} && \text{on } \Gamma_N. && \text{(1e)}
\end{aligned}
$$

Here, $\boldsymbol{F}$ is the deformation gradient and $\boldsymbol{P}$ is the first Piola-Kirchhoff stress tensor. The gradient ∇, and the divergence $\nabla\cdot$ operators are defined with respect to the initial (undeformed) material coordinate system. And $\boldsymbol{N}$ is the outward normal on the undeformed body surface. We assume that the material properties, applied loads and boundary conditions are sufficiently smooth.

We limit the scope of this article to hyperelastic materials. In particular, we assume that an elastic potential energy function $\Psi(\boldsymbol{F})$ exists as a function of the deformation gradient, and that it is related to the first Piola-Kirchhoff stress tensors through the relation (1c).

For the applications considered in this paper, only the Saint Venant-Kirchhoff and the Neo-Hookean hyperelastic models will be considered. Their respective elastic potential functions are given by

$$
\begin{aligned}
\Psi(\boldsymbol{F}) &= \frac{\lambda}{2}(\operatorname{tr} \boldsymbol{E})^2 + \mu \operatorname{tr}(\boldsymbol{E}^2) && \text{Saint Venant-Kirchhoff} && \text{(2a)}\\
\Psi(\boldsymbol{F}) &= \frac{\mu}{2}\big(\operatorname{tr}(\boldsymbol{F}^T\boldsymbol{F}) - 3 - 2\ln J\big) + \frac{\lambda}{2}(\ln J)^2 && \text{Neo-Hookean} && \text{(2b)}
\end{aligned}
$$

where (λ, μ) the Lamé parameters of the model, $J = \det \boldsymbol{F}$ is the the Jacobian, $\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I})$ is the Lagrangian strain tensor, and $\boldsymbol{I}$ the second order identity tensor.

2.2. Approximation spaces

We assume that Ω is divided into a partition $\mathcal{T}_h$ of disjoint elements K, and introduce the set $\partial\mathcal{T}_h = \{\partial K : K \in \mathcal{T}_h\}$, the set of internal faces $\mathcal{E}_h^o =$

$\{\partial K_i \cap \partial K_j : K_i, K_j \in \mathcal{T}_h\}$, the set of boundary faces $\mathcal{E}_h^\partial = \{\partial K_i \cap \partial\Omega : K_i \in \mathcal{T}_h\}$, and set of all faces $\mathcal{E}_h = \mathcal{E}_h^o \cup \mathcal{E}_h^\partial$.

We denote by $\mathcal{P}_k(K)$ the set of polynomials of degree at most k whose support is the element K., and introduce the following broken polynomial spaces

$$\boldsymbol{V}_h := \{\boldsymbol{G} \in [L^2(\Omega)]^{d\times d} : \boldsymbol{G}|_K \in [\mathcal{P}_k(K)]^{d\times d}, \forall K \in \mathcal{T}_h\}, \tag{3a}$$

$$\boldsymbol{W}_h := \{\boldsymbol{w} \in L^2(\Omega)^d : \boldsymbol{w}|_K \in \mathcal{P}_k(K)^d, \forall K \in \mathcal{T}_h\}, \tag{3b}$$

$$\boldsymbol{M}_h := \{\boldsymbol{\mu} \in L^2(\mathcal{E}_h)^d : \boldsymbol{\mu}|_F \in \mathcal{P}_k(F)^d, \forall F \in \mathcal{E}_h\}, \tag{3c}$$

where $L^2(D)$ is the space of square integrable functions on D. We have chosen equal polynomial degrees for vector, tensor and trace spaces. However, the HDG framework is quite general and, in principle, it allows for other approximation spaces such as the Raviart-Thomas and the Brezzi-Douglas-Marini spaces as noted in [48, 49].

Finally, we define various inner products for our finite element spaces

$$(\boldsymbol{u}, \boldsymbol{v})_K := \int_K \boldsymbol{u}\cdot\boldsymbol{v}\,\mathrm{d}\Omega, \qquad (\boldsymbol{u}, \boldsymbol{v})_{\mathcal{T}_h} := \sum_{K\in\mathcal{T}_h} (\boldsymbol{u}, \boldsymbol{v})_K, \qquad \forall \boldsymbol{u}, \boldsymbol{v} \in L^2(\Omega)^d,$$

$$(\boldsymbol{G}, \boldsymbol{H})_K := \int_K \boldsymbol{G} : \boldsymbol{H}\,\mathrm{d}\Omega, \qquad (\boldsymbol{G}, \boldsymbol{H})_{\mathcal{T}_h} := \sum_{K\in\mathcal{T}_h} (\boldsymbol{G}, \boldsymbol{H})_K, \quad \forall \boldsymbol{G}, \boldsymbol{H} \in L^2(\Omega)^{d\times d},$$

$$\langle \boldsymbol{\mu}, \boldsymbol{\eta}\rangle_{\partial K} := \int_{\partial K} \boldsymbol{\mu}\cdot\boldsymbol{\eta}\,\mathrm{d}\Gamma, \qquad \langle \boldsymbol{\mu}, \boldsymbol{\eta}\rangle_{\partial\mathcal{T}_h} := \sum_{K\in\mathcal{T}_h} \langle \boldsymbol{\mu}, \boldsymbol{\eta}\rangle_{\partial K}, \qquad \forall \boldsymbol{\mu}, \boldsymbol{\eta} \in L^2(\partial\mathcal{T}_h)^d.$$

In the next section, we will define the HDG method for solving the problem (1).

3. Hybridizable discontinuous Galerkin formulation

3.1. Variational principle

As explained in [16], the HDG method for nonlinear elasticity can be seen as a minimization problem of an energy functional. The functional proposed therein is a function of the deformation $\boldsymbol{\varphi}$ and the deformation traces $\hat{\boldsymbol{\varphi}} := \boldsymbol{\varphi}|_{\mathcal{E}_h}$, with the deformation gradient being retrieved via the use of the DG-derivative [5, 16].

We present here an alternative 4-variables variational principle that can be used to derive the same HDG equations, without making use of the HDG-derivative.

This variational principle is associated to the following functional defined for fields $(\boldsymbol{\varphi}, \boldsymbol{P}, \boldsymbol{F}, \hat{\boldsymbol{\varphi}}) \in \boldsymbol{W}_h \times \boldsymbol{V}_h \times \boldsymbol{V}_h \times \boldsymbol{M}_h$,

$$\begin{aligned}\Pi(\boldsymbol{\varphi}, \boldsymbol{P}, \boldsymbol{F}, \hat{\boldsymbol{\varphi}}) := {} & (\Psi(\boldsymbol{F}), 1)_{\mathcal{T}_h} + (\boldsymbol{P}, (\nabla\boldsymbol{\varphi} - \boldsymbol{F}))_{\mathcal{T}_h} \\ & - \langle \boldsymbol{P}\boldsymbol{N}, (\boldsymbol{\varphi} - \hat{\boldsymbol{\varphi}})\rangle_{\partial\mathcal{T}_h} + \frac{1}{2}\langle(\boldsymbol{\varphi} - \hat{\boldsymbol{\varphi}}), \boldsymbol{\tau}(\boldsymbol{\varphi} - \hat{\boldsymbol{\varphi}})\rangle_{\partial\mathcal{T}_h} \\ & - (\boldsymbol{f}, \boldsymbol{\varphi})_{\mathcal{T}_h} - \langle \boldsymbol{t}, \hat{\boldsymbol{\varphi}}\rangle_{\Gamma_N} + \frac{1}{2}\langle(\hat{\boldsymbol{\varphi}} - \boldsymbol{\varphi}_D), \boldsymbol{\tau}(\hat{\boldsymbol{\varphi}} - \boldsymbol{\varphi}_D)\rangle_{\Gamma_D},\end{aligned} \tag{5}$$

where $\boldsymbol{\tau}$ is the stabilization matrix. The first term on the right hand side corresponds to the internal energy of the elastic body, the second measures an energy associated to the mismatch between $\nabla\boldsymbol{\varphi}$ and $\boldsymbol{F}$. The third and fourth terms measure an energy related to the jump of the solution at the elements boundaries. In particular, the fourth term is typical of the HDG formulation. As an energy quantity, it has to be positive i.e. the matrix $\boldsymbol{\tau}$ has to be symmetric definite positive. The choice of $\boldsymbol{\tau}$ crucially affects the performances of the method (see discussion in 3.3). The fifth term is the energy related with the external body forces. Finally, the last two terms are the energies associated with the imposed tractions and displacements. Although the Dirichlet boundary condition is applied weakly here, it could be applied alternatively in a strong manner through a suitable modification of the space $\boldsymbol{M}_h$.

Interestingly, the variational principle (5) becomes the Hu-Washizu principle when $\hat{\boldsymbol{\varphi}} \equiv \boldsymbol{\varphi}$ on $\mathcal{F}_h$. Moreover, if $\boldsymbol{F} \equiv \nabla\boldsymbol{\varphi}$, it becomes the standard total energy used for continuous Galerkin displacement formulations (see for instance [50]).

We now define the directional derivative of Π with respect to its first variable and in the direction $\boldsymbol{w}$ as

$$D_1\Pi(\boldsymbol{\varphi}, \boldsymbol{P}, \boldsymbol{F}, \hat{\boldsymbol{\varphi}})[\boldsymbol{w}] := \frac{\partial}{\partial\epsilon}\Pi(\boldsymbol{\varphi} + \epsilon\boldsymbol{w}, \boldsymbol{P}, \boldsymbol{F}, \hat{\boldsymbol{\varphi}})\bigg|_{\epsilon=0}, \tag{6}$$

for any $\boldsymbol{w} \in \boldsymbol{W}_h$. The directional derivatives $D_2\Pi, D_3\Pi$ and $D_4\Pi$ with respect to the other variables can be defined in a similar way.

We can now express the HDG equations as a variational principle. The HDG approximation $(\boldsymbol{\varphi}_h, \boldsymbol{P}_h, \boldsymbol{F}_h, \hat{\boldsymbol{\varphi}}_h)$ to the exact solution $(\boldsymbol{\varphi}, \boldsymbol{P}, \boldsymbol{F}, \hat{\boldsymbol{\varphi}})$ is the element of the approximation space $\boldsymbol{W}_h \times \boldsymbol{V}_h \times \boldsymbol{V}_h \times \boldsymbol{M}_h$ that locally minimizes the energy functional Π, that is making all the directional derivatives simultaneously equal to zero.

Thus, for the first directional derivative, $D_1\Pi = 0$ yields the following HDG approximation

$$(\boldsymbol{P}_h, (\nabla \boldsymbol{w}))_{\mathcal{T}_h} - (\boldsymbol{f}, \boldsymbol{w})_{\mathcal{T}_h} - \langle \boldsymbol{P}_h \boldsymbol{N}, \boldsymbol{w} \rangle_{\partial\mathcal{T}_h} + \langle \boldsymbol{\tau}(\boldsymbol{\varphi}_h - \hat{\boldsymbol{\varphi}}_h), \boldsymbol{w} \rangle_{\partial\mathcal{T}_h} = 0, \tag{7}$$

for all $\boldsymbol{w} \in \boldsymbol{W}_h$. By introducing the numerical traction traces

$$\widehat{\boldsymbol{P}_h \boldsymbol{N}} := \boldsymbol{P}_h \boldsymbol{N} - \boldsymbol{\tau}(\boldsymbol{\varphi}_h - \hat{\boldsymbol{\varphi}}_h) \quad \text{on } \partial\mathcal{T}_h, \tag{8}$$

we can rewrite (7) as

$$(\boldsymbol{P}_h, (\nabla \boldsymbol{w}))_{\mathcal{T}_h} - (\boldsymbol{f}, \boldsymbol{w})_{\mathcal{T}_h} - \langle \widehat{\boldsymbol{P}_h \boldsymbol{N}}, \boldsymbol{w} \rangle_{\partial\mathcal{T}_h} = 0, \quad \forall \boldsymbol{w} \in \boldsymbol{W}_h. \tag{9}$$

For the second directional derivative, setting $D_2\Pi = 0$ we get

$$((\nabla \boldsymbol{\varphi}_h - \boldsymbol{F}_h), \boldsymbol{G})_{\mathcal{T}_h} - \langle (\boldsymbol{\varphi}_h - \hat{\boldsymbol{\varphi}}_h, \boldsymbol{G}\boldsymbol{N} \rangle_{\partial\mathcal{T}_h} = 0, \quad \forall \boldsymbol{G} \in \boldsymbol{V}_h,$$

where the gradient term can be integrated by parts to obtain

$$-(\boldsymbol{\varphi}_h, \nabla \cdot \boldsymbol{G})_{\mathcal{T}_h} - (\boldsymbol{F}_h, \boldsymbol{G})_{\mathcal{T}_h} + \langle \hat{\boldsymbol{\varphi}}_h, \boldsymbol{G}\boldsymbol{N} \rangle_{\partial\mathcal{T}_h} = 0, \quad \forall \boldsymbol{G} \in \boldsymbol{V}_h. \tag{10}$$

The vanishing condition for third directional derivative, $D_3\Pi = 0$, yields

$$\left(\frac{\partial \Psi(\boldsymbol{F}_h)}{\partial \boldsymbol{F}_h} - \boldsymbol{P}_h, \boldsymbol{Q}\right)_{\mathcal{T}_h} = 0, \qquad \forall \boldsymbol{Q} \in \boldsymbol{V}_h. \tag{11}$$

And finally, enforcing $D_4\Pi = 0$, we obtain

$$\langle \widehat{\boldsymbol{P}_h \boldsymbol{N}}, \boldsymbol{\mu} \rangle_{\partial\mathcal{T}_h} - \langle \boldsymbol{t}, \boldsymbol{\mu} \rangle_{\Gamma_N} + \langle \boldsymbol{\tau}(\hat{\boldsymbol{\varphi}}_h - \boldsymbol{\varphi}_D), \boldsymbol{\mu} \rangle_{\Gamma_D} = 0, \qquad \forall \boldsymbol{\mu} \in \boldsymbol{M}_h. \tag{12}$$

3.2. Weak formulation

The HDG solution satisfies equations (9), (10), (11) and (12), which we gather now in a more customary fashion : the HDG method seeks an approximation $(\boldsymbol{\varphi}_h, \boldsymbol{P}_h, \boldsymbol{F}_h, \hat{\boldsymbol{\varphi}}_h) \in \boldsymbol{W}_h \times \boldsymbol{V}_h \times \boldsymbol{V}_h \times \boldsymbol{M}_h$ such that

$$(\boldsymbol{P}_h, \nabla \boldsymbol{w})_{\mathcal{T}_h} - \langle \widehat{\boldsymbol{P}_h} \boldsymbol{N}, \boldsymbol{w} \rangle_{\partial \mathcal{T}_h} = (\boldsymbol{f}, \boldsymbol{w})_{\mathcal{T}_h} \quad \forall \boldsymbol{w} \in \boldsymbol{W}_h, \tag{13a}$$

$$(\boldsymbol{F}_h, \boldsymbol{G})_{\mathcal{T}_h} + (\boldsymbol{\varphi}_h, \nabla \cdot \boldsymbol{G})_{\mathcal{T}_h} - \langle \hat{\boldsymbol{\varphi}}_h, \boldsymbol{G}\boldsymbol{N} \rangle_{\partial \mathcal{T}_h} = 0 \quad \forall \boldsymbol{G} \in \boldsymbol{V}_h, \tag{13b}$$

$$(\boldsymbol{P}_h, \boldsymbol{Q})_{\mathcal{T}_h} - (\frac{\partial \Psi}{\partial \boldsymbol{F}_h}, \boldsymbol{Q})_{\mathcal{T}_h} = 0 \quad \forall \boldsymbol{Q} \in \boldsymbol{V}_h, \tag{13c}$$

$$\langle \widehat{\boldsymbol{P}_h} \boldsymbol{N}, \boldsymbol{\mu} \rangle_{\partial \mathcal{T}_h \setminus \Gamma_D} + \langle \boldsymbol{\tau}(\hat{\boldsymbol{\varphi}}_h - \boldsymbol{\varphi}_D), \boldsymbol{\mu} \rangle_{\Gamma_D} = \langle \boldsymbol{g}, \boldsymbol{\mu} \rangle_{\Gamma_N} \quad \forall \boldsymbol{\mu} \in \boldsymbol{M}_h, \tag{13d}$$

where the numerical traction traces are

$$\widehat{\boldsymbol{P}_h} \boldsymbol{N} := \boldsymbol{P}_h \boldsymbol{N} - \boldsymbol{\tau}(\boldsymbol{\varphi}_h - \hat{\boldsymbol{\varphi}}_h) \quad \text{on } \partial \mathcal{T}_h. \tag{13e}$$

Note that the equation involving the traces (13d), enforce both the boundary conditions (Neumann and Dirichlet) and the jump of $\widehat{\boldsymbol{P}_h} \boldsymbol{N}$ to be zero on the internal faces. This last condition is commonly referred as the *conservativity condition*.

The HDG method presented in this article is therefore similar to [15, 16]. It differs from [11] since no approximation of the pressure field as such is made in our formulation.

Although $\boldsymbol{F}_h$ and $\boldsymbol{P}_h$ are considered here as separate variables, $\boldsymbol{P}_h(\boldsymbol{F}_h)$ can be computed elementwise with equation (13c). Therefore, in the remainder of this paper we will consider only $(\boldsymbol{\varphi}_h, \boldsymbol{F}_h, \hat{\boldsymbol{\varphi}}_h)$ as separate variables.

3.3. Choice of the stabilization Tensor

The choice of the stabilization tensor $\boldsymbol{\tau}$ plays a crucial role in both the accuracy and the stability of the method. A very large $\boldsymbol{\tau}$ means a strong penalization of the inter element discontinuities, in which case the HDG solution becomes very close to a conforming continuous solution. Therefore, for large $\boldsymbol{\tau}$, the HDG solution mimics the good and bad properties of conforming methods. Among the good ones, the coercivity is ensured, and hence the stability of the linearized problem. Among the bad ones are the sensitivity of the numerical solution to the chosen mesh and the various locking phenomena (volumetric and thickness-related lockings). In particular, the thickness-related lockings, i.e. shear locking, membrane locking and trapezoidal locking are often too severe to make legacy volumetric finite elements applicable to shell problems. These effects are briefly illustrated in subsection 5.3.2.

On the contrary, for smaller values of the stabilization parameter, the DG-based methods have shown a better accuracy than their CG counterparts, as allowing jumps of the solution at the elements boundaries provides a mechanism that significantly mitigates the various locking pathologies. This advantage has justified the over-cost of the DG-based approximations. However, it comes with the risk of loosing the coercivity of the discrete problem, which translates into either a non-convergence of the Newton algorithm, or into a converged non-physical state of deformation (see [16]). Therefore, an ideal $\boldsymbol{\tau}$ would be large enough to ensure the stability, while being small enough to retain an optimal accuracy. How to automatically choose the optimal stabilization is still a theoretical issue which has been partially addressed in [18] by providing lower bounds for $\boldsymbol{\tau}$ and for simplex meshes. Although a theoretical estimate of $\boldsymbol{\tau}$ is beyond the scope of this paper, we hope to provide some practical insights to choose an appropriate stabilization.

3.3.1. Review of some stabilization tensors for nonlinear elasticity

Several stabilization strategies have been proposed in the literature for both DG and HDG approaches. However, all of them were found to be of little use when nonlinear shell problems are solved with HDG. We now briefly review them and mention what we think are their shortcomings.

The simplest approach [11], based on a dimensional analysis is to choose

$$\boldsymbol{\tau} = \frac{1}{L_c}\,\mu\,\boldsymbol{I}, \tag{14}$$

where L_c is a length scale that only depends on the discretized geometry of the structure, μ the Lamé parameter and $\boldsymbol{I}$ the second order identity tensor. The first issue is the choice of L_c. For a shell structure problem, there are at least three different length scales candidates: the typical size of the whole structure L, the element size h and the thickness t. Numerical experiments strongly suggest that $h \lessapprox L_c < L$. Indeed $L_c \approx t$ usually gives an over-stiff discrete model and underestimated displacements, while the coercivity may be lost for $L_c \approx L$. Our experience has shown that, although L_c may be determined after a few trials, $L_c \approx h$ is a safe default choice and is always satisfactory for moderate strains. However, this relation has to remain loose since the accuracy of the approximate gradient will deteriorate if $L_c = h$ for very fine meshes (see [10]) and the postprocessing benefit will then be lost. Therefore we understand L_c as the typical mesh size of a coarse mesh able to capture the features of the solution. If the mesh is further refined to get a better accuracy, L_c is kept the same.

However the stabilization tensor (14) usually fails when large strains occur. For instance, the cylindrical test cases presented in section 5.3 need a greater stabilization near the applied point forces, which means that the stabilization should adaptive, i.e. depending on the local state of strains/stresses, as already noted in [51, 52].

The first attempt [15] to design an adaptive $\boldsymbol{\tau}$ was to make use of the material fourth order elasticity tensor $\mathbf{C}$, by defining

$$[\boldsymbol{\tau}]_{IK} = \frac{1}{L_c} C_{IJKL} N_J N_L \quad \text{with} \quad [\mathbf{C}]_{IJKL} = \frac{\partial^2 \Psi}{\partial E_{IJ} \partial E_{KL}}. \tag{15}$$

However, this stabilization is also insufficient for large strains, notably when a Saint Venant-Kirchhoff model is considered since $\mathbf{C}$ is then constant.

Alternatively, the viscous stabilization designed for the Navier-Stokes equations [53] could be used, with

$$[\boldsymbol{\tau}]_{ik} = \frac{1}{L_c} \frac{\partial P_{iJ}}{\partial F_{kL}} N_J N_L. \tag{16}$$

Although this stabilization should intuitively grow with the local deformation gradient, there is no control on the smallest eigenvalue of $\boldsymbol{\tau}$ and it may actually become very small, making the model unstable.

All three of the above stabilizations fail at some point for the problems presented in section 5.3. See the Appendix C for the detailed results.

Another DG stabilization strategy [51] is based on the observation that the regions where the numerical instabilities develop usually coincide with the regions were the elasticity tensor becomes indefinite. A small amount of initial stabilization $\boldsymbol{\tau}_0$ is then increased with an adaptive term proportional to the lowest negative eigenvalue of the local elasticity tensor

$$\boldsymbol{\tau} = \frac{\beta}{L_c}\left(\boldsymbol{\tau}_0 - \rho_{\min}\left(\frac{\partial \boldsymbol{P}_h}{\partial \boldsymbol{F}_h}\right)\boldsymbol{I}\right), \tag{17}$$

where β is some scaling factor, and $\rho_{\min}$ is the minimum negative eigenvalue of the tensor $\frac{\partial \boldsymbol{P}_h}{\partial \boldsymbol{F}_h}$ locally evaluated, with $\rho_{\min} = 0$ if the eigenvalues are all positive. This approach worked successfully for large strains experiments considered in [51]. However, our numerical experiments reported in Appendix C show only mitigated results for the shell problems considered in this paper, since the parameter β has to be tuned case by case.

Lately, a lower bound for the HHO stabilization has been derived in [18] for nonlinear elasticity

$$\boldsymbol{\tau} = \tau \boldsymbol{I} \quad \text{with} \quad \tau > \frac{C_o}{h_F} + \frac{C_\theta}{h_F}, \tag{18}$$

where h_F if the diameter of the face, C_o is a local constant depending only on the local mesh properties, and C_θ is a local constant depending on both local and global eigenvalues of the elasticity tensor. The authors propose an astute way of solving the global eigenvalue problem by using an embedded Discontinuous Galerkin approximation. Moreover, although the optimal convergence of the gradient may be lost since this stabilization is of order $1/h$, it can be retrieved by using locally a polynomial degree $k+1$ in the elements where the elasticity tensor is indefinite. However, this method being designed for simplexes, it cannot be included in our comparative study.

3.3.2. Proposed stabilization tensor

In this paper, we propose an empirical stabilization based on the maximum eigenvalue of the elasticity tensor, by choosing

$$\boldsymbol{\tau} = \frac{1}{2L_c}\rho_{\max}\left(\frac{\partial \boldsymbol{P}_h}{\partial \boldsymbol{F}_h}\right)\boldsymbol{I}, \tag{19}$$

with $\rho_{\max}$ the maximum of the largest eigenvalue of the elasticity tensor evaluated at all the Gauss points of a given face. Therefore $\boldsymbol{\tau}$ is constant face by face and depends on the deformations gradient. This stabilization is related to the local Lax-Friedrichs numerical flux for the hyperbolic problems. Here are a few comments regarding equation (19).

- In the linear elasticity limit, the stabilization tensor becomes $\boldsymbol{\tau} = \frac{1}{2L_c}(2\mu + 3\lambda)\,\boldsymbol{I}$. Moreover, when $\nu = 0$, it becomes equal to (14).
- For shell applications, we could notice that equation (19) adds a substantial stabilization near the point forces and the wrinkles, where the onset of instabilities usually occurs.
- In the nearly-incompressible limit of nonlinear hyperelastic models, our experience is that the stabilization has to be significantly increased for HDG to converge. Equation (19) provides such a mechanism since $\rho_{\max}$ will grow as $\nu \to 0.5$.

All the results presented in this paper have been obtained using this stabilization mechanism, and selecting L_c along the guidelines mentioned above. For a detailed comparison of the performances of the above stabilizations, see Appendix C.

4. Implementation

4.1. Loading Incrementation

Our load-control algorithm 1 is provided in Appendix A. It is based on the standard incrementation algorithm used in ABAQUS ([54]) with some minors modifications. Note that the external forces or prescribed tractions appearing in (13) are denoted with the generic term $\mathbf{P}_{\max}$. However, $\mathbf{P}_{\max}$ is not directly prescribed, but an incremented fraction of it $\mathbf{P} = \lambda \mathbf{P}_{\max}$ with the fraction coefficient $0 \leq \lambda \leq 1$.

The Newton-Raphson procedure is not allowed to do more than $n_{\max}$ iterations. If the convergence is not obtained after $n_{\max}$ iterations, the load increment is then decreased. On the contrary, if the convergence has been sufficiently fast for the last two increments, the increment magnitude is then increased. We take $n_{\max} = 20$ and $\lambda_{\text{init}} = 0.05$ as default values. This algorithm is found to work well for all the test cases we have studied in this article. Our algorithm also gives the final number of load increments n_{inc} and the total accumulated number of Newton-Raphson iterations n_{tot} in order to get an idea of the method's efficiency and stability.

4.2. Newton-Raphson algorithm

At each step of the loading algorithm 1, the Newton-Raphson procedure is called to solve the nonlinear system (13). The procedure evaluates successive approximations $(\boldsymbol{F}_h^l, \boldsymbol{\varphi}_h^l, \hat{\boldsymbol{\varphi}}_h^l)$ of the unknowns under the current load P starting from the converged state at the previous load $(\boldsymbol{F}_h^0, \boldsymbol{\varphi}_h^0, \hat{\boldsymbol{\varphi}}_h^0)$. For each Newton step l, the system of equations (13) is linearized with respect to the Newton increments $\left(\delta\boldsymbol{F}_h^l, \delta\boldsymbol{\varphi}_h^l, \delta\hat{\boldsymbol{\varphi}}_h^l\right) \in \boldsymbol{V}_h \times \boldsymbol{W}_h \times \boldsymbol{M}_h$. These increments then satisfy the system

$$
\begin{aligned}
&((\partial_{\boldsymbol{F}_h}\boldsymbol{P}_h)\delta\boldsymbol{F}_h^l, \nabla\boldsymbol{w})_{\mathcal{T}_h} \\
&-\langle(\partial_{\boldsymbol{F}_h}\widehat{\boldsymbol{P}}_h\boldsymbol{N})\delta\boldsymbol{F}_h^l + (\partial_{\boldsymbol{\varphi}_h}\widehat{\boldsymbol{P}}_h\boldsymbol{N})\delta\boldsymbol{\varphi}_h^l + (\partial_{\hat{\boldsymbol{\varphi}}_h}\widehat{\boldsymbol{P}}_h\boldsymbol{N})\delta\hat{\boldsymbol{\varphi}}_h^l, \boldsymbol{w}\rangle_{\partial\mathcal{T}_h} = r_1(\boldsymbol{w}), && \text{(20a)}\\
&(\delta\boldsymbol{F}_h^l, \boldsymbol{G})_{\mathcal{T}_h} + (\delta\boldsymbol{\varphi}_h^l, \nabla\cdot\boldsymbol{G})_{\mathcal{T}_h} - \langle\delta\hat{\boldsymbol{\varphi}}_h^l, \boldsymbol{G}\boldsymbol{N}\rangle_{\partial\mathcal{T}_h} = r_2(\boldsymbol{G}), && \text{(20b)}\\
&((\partial_{\boldsymbol{F}_h}\boldsymbol{P}_h)\delta\boldsymbol{F}_h^l, \boldsymbol{Q})_{\mathcal{T}_h} - (\partial_{\boldsymbol{F}_h}(\partial_{\boldsymbol{F}_h}\Psi(\boldsymbol{F}_h))\delta\boldsymbol{F}_h^l, \boldsymbol{Q})_{\mathcal{T}_h} = 0, && \text{(20c)}\\
&\langle(\partial_{\boldsymbol{F}_h}\widehat{\boldsymbol{P}}_h\boldsymbol{N})\delta\boldsymbol{F}_h^l + (\partial_{\boldsymbol{\varphi}_h}\widehat{\boldsymbol{P}}_h\boldsymbol{N})\delta\boldsymbol{\varphi}_h^l + (\partial_{\hat{\boldsymbol{\varphi}}_h}\widehat{\boldsymbol{P}}_h\boldsymbol{N})\delta\hat{\boldsymbol{\varphi}}_h^l, \boldsymbol{\mu}\rangle_{\partial\mathcal{T}_h\setminus\Gamma_D} \\
&\qquad + \langle\boldsymbol{\tau}\delta\hat{\boldsymbol{\varphi}}_h^l, \boldsymbol{\mu}\rangle_{\Gamma_D} = r_3(\boldsymbol{\mu}), && \text{(20d)}
\end{aligned}
$$

for all $(\boldsymbol{G}, \boldsymbol{Q}, \boldsymbol{w}, \mu) \in \boldsymbol{V}_h \times \boldsymbol{V}_h \times \boldsymbol{W}_h \times \boldsymbol{M}_h$, the right-hand side residuals are given by

$$
\begin{aligned}
r_1(\boldsymbol{w}) &= (\boldsymbol{f}, \boldsymbol{w})_{\mathcal{T}_h} - (\boldsymbol{P}_h^l, \nabla\boldsymbol{w})_{\mathcal{T}_h} + \langle\widehat{\boldsymbol{P}}_h(\boldsymbol{F}_h^l, \boldsymbol{\varphi}_h^l, \hat{\boldsymbol{\varphi}}_h^l)\boldsymbol{N}, \boldsymbol{w}\rangle_{\partial\mathcal{T}_h}, && \text{(21a)}\\
r_2(\boldsymbol{G}) &= -(\boldsymbol{F}_h, \boldsymbol{G})_{\mathcal{T}_h} - (\boldsymbol{\varphi}_h^l, \nabla\cdot\boldsymbol{G})_{\mathcal{T}_h} + \langle\hat{\boldsymbol{\varphi}}_h^l, \boldsymbol{G}\boldsymbol{N}\rangle_{\partial\mathcal{T}_h}, && \text{(21b)}\\
r_3(\boldsymbol{\mu}) &= \langle\boldsymbol{t}, \boldsymbol{\mu}\rangle_{\Gamma_N} - \langle\widehat{\boldsymbol{P}}_h(\boldsymbol{F}_h^l, \boldsymbol{\varphi}_h^l, \hat{\boldsymbol{\varphi}}_h^l)\boldsymbol{N}, \boldsymbol{\mu}\rangle_{\partial\mathcal{T}_h\setminus\Gamma_D} - \langle\boldsymbol{\tau}(\hat{\boldsymbol{\varphi}}_h^l - \boldsymbol{\varphi}_D), \boldsymbol{\mu}\rangle_{\Gamma_D}. && \text{(21c)}
\end{aligned}
$$

All the residuals (21) are evaluated at the current iterate $(\boldsymbol{F}_h^l, \boldsymbol{\varphi}_h^l, \hat{\boldsymbol{\varphi}}_h^l)$. The external forces $\boldsymbol{f}$ and tractions $\boldsymbol{t}$ are given by the current load P. Note that equations (20a), (20b) and (20d) are the linearization of equations (13a), (13b) and (13d) respectively, while equation (20c) is the differentiation of (13c) with respect to $\delta\boldsymbol{F}_h^l$ and yields the sensitivity of $\boldsymbol{P}_h$ with respect to $\boldsymbol{F}_h$. In (20), $(\partial_{\boldsymbol{F}_h}\cdot), (\partial_{\boldsymbol{\varphi}_h}\cdot)$ and $(\partial_{\hat{\boldsymbol{\varphi}}_h}\cdot)$ denote the partial derivatives with respect to $\boldsymbol{F}_h, \boldsymbol{\varphi}_h$ and $\hat{\boldsymbol{\varphi}}_h$ respectively.

After solving (20), the numerical approximations are then updated

$$\left(\boldsymbol{F}_h^{l+1}, \boldsymbol{\varphi}_h^{l+1}, \hat{\boldsymbol{\varphi}}_h^{l+1}\right) := \left(\boldsymbol{F}_h^{l}, \boldsymbol{\varphi}_h^{l}, \hat{\boldsymbol{\varphi}}_h^{l}\right) + \alpha\left(\delta\boldsymbol{F}_h^{l}, \delta\boldsymbol{\varphi}_h^{l}, \delta\hat{\boldsymbol{\varphi}}_h^{l}\right), \tag{22}$$

where the coefficient α is determined by a line-search algorithm in order to optimally decrease the residual. This process is repeated and l is incremented until the residual norm is smaller than a given tolerance, typically 10^{-7}.

4.3. Linear system resolution

At each step of the Newton-Raphson algorithm, the linearization (20) gives the following matrix system to be solved

$$\begin{pmatrix} \mathbb{A}^l & \mathbb{B}^l \\ \mathbb{C}^l & \mathbb{D}^l \end{pmatrix} \begin{pmatrix} \delta U^l \\ \delta \widehat{U}^l \end{pmatrix} = \begin{pmatrix} R_{12}^l \\ R_3^l \end{pmatrix}, \tag{23}$$

where δU^l and $\delta\widehat{U}^l$ are the vectors of degrees of freedom of $(\delta\boldsymbol{F}_h^l, \delta\boldsymbol{\varphi}_h^l)$ and $\delta\hat{\boldsymbol{\varphi}}_h^l$ respectively. Following the HDG resolution strategy, the system (23) is first solved for the traces only $\delta\widehat{U}^l$

$$\mathbb{K}^l \delta\widehat{U}^l = R^l, \tag{24}$$

where $\mathbb{K}^l$ is the Schur complement of the block $\mathbb{A}$ and R^l is the reduced residual

$$\mathbb{K}^l = \mathbb{D}^l - \mathbb{C}^l\left(\mathbb{A}^l\right)^{-1}\mathbb{B}^l, \qquad R^l = R_3^l - \mathbb{C}^l\left(\mathbb{A}^l\right)^{-1}R_{12}^l. \tag{25}$$

The reduced system (24) involves fewer degrees of freedom than the full system (23). Moreover due to the discontinuous nature of the approximate solution $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h)$, the matrix $\mathbb{A}^l$ and its inverse are block diagonal, and can be computed

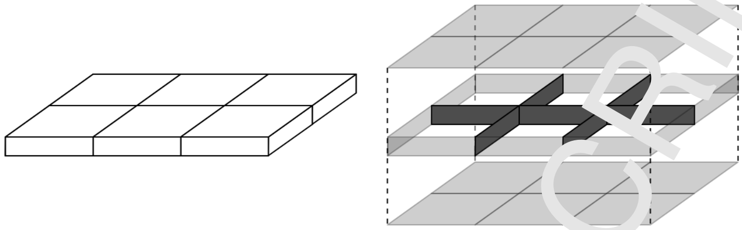

Figure 1: On the left is represented a simple thin rectangular structure. The support of the hybrid unknowns $\delta\widehat{\boldsymbol{\varphi}}_h^l$ is on all the faces. On the right is represented, for the same structure, the support of the internal DoF $\delta\widehat{U}_I^l$ in dark gray and the support of the boundary DoF $\delta\widehat{U}_B^l$ in light gray. Only the internal DoF are actually globally coupled.

elementwise. Once $\delta\widehat{U}^l$ is known, the other unknowns U^l are then retrieved element-wise. Therefore, the full system (23) is never explicitly built, and the reduced matrix $\mathbb{K}^l$ is build directly in an elementwise fashion, thus reducing the memory storage.

The global system (24) can be further reduced by eliminating the unknowns located on the boundary faces. If we denote the unknowns on the boundary faces by $\delta\widehat{U}_B^l$ and the unknowns on the the interior faces $\mathcal{E}_h^o$ by $\delta\widehat{U}_I^l$ such that $\delta\widehat{U}^l = (\delta\widehat{U}_B^l, \delta\widehat{U}_I^l)^T$, the system (24) becomes

$$\begin{pmatrix} \mathbb{K}_{BB}^l & \mathbb{K}_{BI}^l \\ \mathbb{K}_{IB}^l & \mathbb{K}_{II}^l \end{pmatrix} \begin{pmatrix} \delta\widehat{U}_B^l \\ \delta\widehat{U}_I^l \end{pmatrix} = \begin{pmatrix} R_B^l \\ R_I^l \end{pmatrix}. \tag{26}$$

The geometrical supports of $\delta\widehat{U}_B^l$ and $\delta\widehat{U}_I^l$ are illustrated in Fig 1. Thanks to the discontinuous nature of the approximation space $\boldsymbol{M}_h$, the matrix $\mathbb{K}_{BB}^l$ is block diagonal and each block can be inverted independently. Therefore, we can efficiently reduce the system (26) to

$$\mathbb{K}_I^l \delta\widehat{U}_I^l = R_R^l, \tag{27}$$

where

$$\mathbb{K}_I = \mathbb{K}_{II}^l - \mathbb{K}_{IB}^l \left(\mathbb{K}_{BB}^l\right)^{-1} \mathbb{K}_{BI}^l \quad \text{and} \quad R_R^l = R_I^l - \mathbb{K}_{IB}^l \left(\mathbb{K}_{BB}^l\right)^{-1} R_B^l. \tag{28}$$

For typical thin structures, the size of the global system (27) is about half of the original system (26). As a result, the global linear system (27) resulting from the HDG method is cheaper to solve than those from other volumetric finite element methods. This makes the HDG method ideally suited for thin structures.

4.4. Coupled degrees of freedom

Let us consider a simple thin structure such as a rectangular plate, as shown in in Fig. 1. This plate is divided into a regular 2D grid of $\mathtt{N_x}$ times $\mathtt{N_y}$ hexahedra elements (with only one element in the thickness direction). Such a structure contains a total of $\mathtt{N_x}(\mathtt{N_y}+1)+\mathtt{N_y}(\mathtt{N_x}+1)+2\mathtt{N_x}\mathtt{N_y}$ faces. Assuming that $\mathtt{NDOF}_{\mathtt{face}}$ degrees of freedom are associated to each face, there is a total of

$$\mathtt{NDOF}_{\mathtt{face}}\left[\mathtt{N_x}(\mathtt{N_y}+1)+\mathtt{N_y}(\mathtt{N_x}+1)+2\mathtt{N_x}\mathtt{N_y}\right] \tag{29}$$

degrees of freedom in the $\delta\widehat{U}^l$ vector of the reduced system (24). Removing the degrees of freedom located on the boundary faces implies that $2(\mathtt{N_x}\mathtt{N_y}+\mathtt{N_x}+\mathtt{N_y})$ faces are excluded. That reduces the total number of degrees of freedom on the interior faces to be

$$\mathtt{NDOF}_{\mathtt{face}}\left[\mathtt{N_x}(\mathtt{N_y}-1)+\mathtt{N_y}(\mathtt{N_x}-1)\right] \tag{30}$$

which is the size of the vector $\delta\widehat{U}^l_I$ in the twice-reduced linear system (27). We see that, in this case, eliminating the boundary unknowns results in a reduction by half in the total number of coupled degrees of freedom.

4.5. Arc-Length Algorithm

Our loading algorithm 1 is known to be unstable when snap-through behaviors appear. The Arc-Length algorithms ([55, 56]) address this shortcoming and are much more robust in the presence of complex snapping behaviors. We propose here an adaptation of the Arc-Length method to the HDG method.

The description of the Arc-Length method 2 is given in Appendix B. It makes uses of two user-defined parameters $(\psi, \Delta l)$. While the parameter ψ does not have much influence on the results, the characteristic length Δl controls the increment size, and has to be small enough to capture the snapping behavior. The classical Arc-Length method makes use of the global vector of nodal displacement increments $\delta\boldsymbol{\varphi}_h$ in the process of determining both the load and displacement increments. Interestingly, in the context of the HDG method, the smaller vector of hybrid increments $\delta\hat{\boldsymbol{\varphi}}_h$ is used instead. The computation of $\delta\lambda$ by solving a quadratic equations gives two possible increments and the choice of the best increment is then based upon a comparison with the previously converged increments (see e.g. [50]).

5. Numerical examples

In this section, we discuss the behavior of the present HDG solid element formulation. We compare our numerical results with shell elements or analytical solutions, when available. When a comparisons with ABAQUS shell elements is shown, it implies that a Saint Venant-Kirchhoff constitutive law has been used, in order to be consistent with the ABAQUS-S4R element formulation [57]. We use the loading algorithm 1 by default, and all results presented in this section make use of the proposed stabilization (19).

Whenever a point force is applied, it is implemented as a nodal force. The node is located on the external (resp. internal) surface for an outward (resp. inward) force, in order to avoid the unpleasant $\det(\boldsymbol{F}_h) \leq 0$ locally.

5.1. Numerical convergence test

In order to illustrate the convergence of the HDG method for thin structures, we propose the following numerical test. A thin square plate of length $L = 1$ and thickness $t = 0.005$ is clamped on its four sides. A Saint Venant-Kirchhoff model is considered with $\mu = 1$ and $\lambda = 2$, i.e. $\nu = 1/3$. Body forces and tractions are prescribed to the plate such that the exact solution for the deformed configuration is $u_x = X, u_y = Y$, and $u_z = Z + 0.4\sin(\pi X)\sin(\pi Y)$, where $\boldsymbol{X} = (X, Y, Z)^T$ are the coordinates of the undeformed plate. Fig. 2 shows the undeformed plate, and the plate at maximum deformation.

We use the postprocessing presented in [11] to get a more accurate approximation of the deformation by making use of the approximate gradient. For each

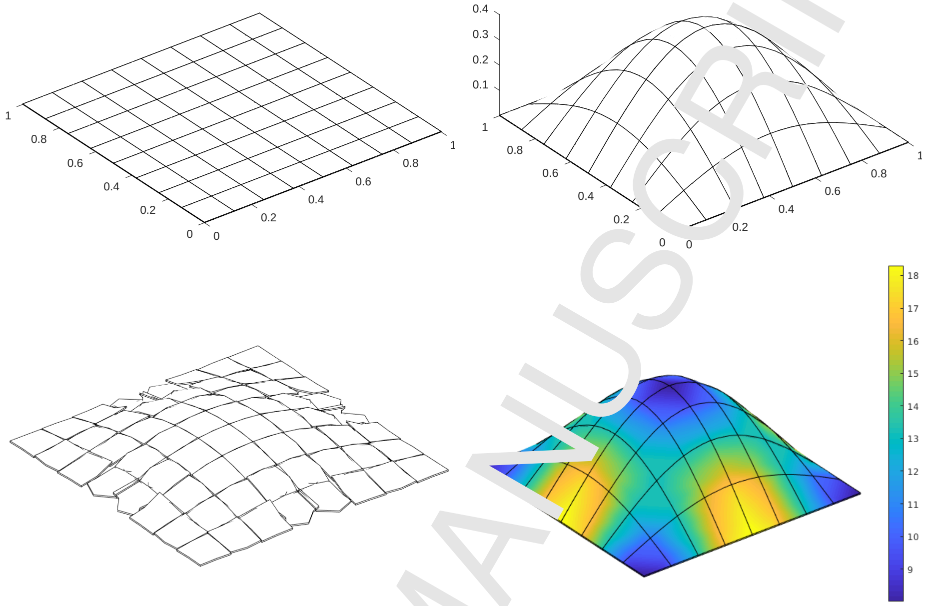

Figure 2: Top left: undeformed square plate. Top right : deformed square plate at maximum load. Bottom left : incipient instability for an insufficient penalization (14) at 0.27 $\mathsf{P}_{\max}$. Bottom right : distribution of $\rho_{\max}$ at maximum load. Results are represented here for $k = 2$ and a 8×8 mesh.

$K \in \mathcal{T}_h$, we build the postprocessed variable $\boldsymbol{\varphi}_h^* \in \mathcal{P}_{k+1}(K)^d$ such as

$$(\nabla \boldsymbol{\varphi}_h^*, \nabla \boldsymbol{w})_{\mathcal{T}_h} = (\boldsymbol{F}_h, \nabla \boldsymbol{w})_{\mathcal{T}_h}, \qquad \forall \boldsymbol{w} \in \mathcal{P}_{k+1}(K)^d \tag{31a}$$

$$(\boldsymbol{\varphi}_h^*, 1)_{\mathcal{T}_h} = (\boldsymbol{\varphi}_h, 1)_{\mathcal{T}_h}. \tag{31b}$$

The table 1 shows the errors of the HDG results and the estimated orders of convergence (e.o.c) when the mesh is refined uniformly in the $\boldsymbol{e}_x$ and $\boldsymbol{e}_y$ directions. All simulations make use of only one element in the thickness direction. Polynomial orders $k \in \{1, 2, 3\}$ are considered, and the adaptive stabilization is given by (19) with $L_c = 0.5$ as characteristic length.

The optimal order of convergence $k + 1$ is observed for the displacement at all polynomial degrees. The observed order of convergence of the gradient varies between $k + \frac{1}{2}$ and $k + 1$. Accordingly, the postprocessed displacement converges with orders between $k+\frac{3}{2}$ and $k+2$. Note that, by varying the values of t and ν, slightly different orders of convergence may be observed, but the previous observations remain valid. See, for instance, the e.o.c in the nearly-incompressible limit (with $\mu = 1$ and $\nu = 0.49999$) reported in table 2. Also, in order to get an accurate postprocessing, L_c has to be large enough, typically $L > L_c \gg t$.

Interestingly, for a linear elastic body and for the mesh sizes considered, the optimal orders of convergence are achieved for a small uniform $\tau_{ii} \leq 0.5$. However, that level of stabilization would be clearly insufficient in the nonlinear case and the Newton algorithm would quickly diverge. Even the higher amount of uniform stabilization given by (14) would fail at some point before $\mathsf{P}_{\max}$ is reached. On the contrary, our adaptive stabilization is successful by using higher values $8 \leq \tau_{ii} \leq 19$, and the areas where the stabilization is large seems to match the areas of incipient instabilities (see Fig. 2, bottom). Therefore, we believe that the discrepancy between the amount of stabilization expected to converge optimally, and the one needed to stabilize the nonlinear model at finite strains is the cause of the slightly suboptimal orders of convergence observed for the gradient.

Although the postprocessing may not always achieve an extra full order of convergence, it always computes a significantly more accurate displacement, at a negligible cost. It therefore remains a attractive feature of the HDG approach. Consequently, all the results presented in this paper are the postprocessed displacements.

5.2. Cantilever problems

5.2.1. Cantilever subjected to a lifting force

Let us consider a cantilever of length $L = 10\,\mathrm{m}$, width $l = 1\,\mathrm{m}$ and thickness $t = 0.1\,\mathrm{m}$, with mechanical properties $E = 1.2 \times 10^6\,\mathrm{kPa}$ and $\nu = 0$. The cantilever is clamped at one end, and is subjected to a lifting force $\mathbf{P} = 4\,\mathrm{kPa}$ at its other end (see Fig. 3). The lifting force is usually a distributed line force when shell elements are considered (for instance [58, 59, 60]). Here the corresponding force is applied through a Neumann boundary condition prescribing the traction

k	mesh size h	$\|\boldsymbol{\varphi}-\boldsymbol{\varphi}_h\|$	e.o.c	$\|\boldsymbol{F}-\boldsymbol{F}_h\|$	e.o.c	$\|\boldsymbol{\varphi}-\boldsymbol{\varphi}_h^*\|$	e.o.c
1	0.5000	2.08e-03	-	2.15e-02	-	2.38e-03	-
	0.2500	8.54e-04	1.28	7.06e-03	1.61	2.88e-04	3.05
	0.1667	3.99e-04	1.88	3.80e-03	1.53	1.01e-04	2.58
	0.1250	2.34e-04	1.86	2.44e-03	1.54	4.80e-05	2.60
	0.0833	1.08e-04	1.92	1.[illegible]e-03	1.54	1.66e-05	2.62
	0.0625	6.15e-05	1.94	8.[illegible]e-04	1.53	7.77e-06	2.63
2	0.5000	3.04e-04	-	[illegible].14e-03	-	1.16e-04	-
	0.2500	3.19e-05	3.25	2.66e-04	3.01	8.32e-06	3.80
	0.1667	1.01e-05	2.[illegible]5	8.79e-05	2.73	1.77e-06	3.81
	0.1250	4.34e-06	2.92	3.80e-05	2.91	5.69e-07	3.95
	0.0833	1.3[illegible]e-06	[illegible].97	1.17e-05	2.91	1.16e-07	3.93
	0.0625	5.52e-07	2.98	5.13e-06	2.87	3.77e-08	3.90
3	0.5000	2.6[illegible]e-05	-	4.69e-04	-	2.15e-05	-
	0.2500	2.70e-06	3.30	2.19e-05	4.42	4.26e-07	5.66
	0.1667	5.32e-07	4.01	5.44e-06	3.43	6.34e-08	4.70
	0.1250	1.74e-07	3.89	1.90e-06	3.66	1.76e-08	4.45
	0.0833	3.53e-08	3.94	4.24e-07	3.69	2.95e-09	4.41
	0.0625	1.13e-08	3.96	1.48e-07	3.64	7.86e-10	4.60

Table 1: History of convergence of the HDG method for the sinusoidally loaded plate, and for a compressible material ($\nu = 1/3$).

k	mesh size h	$\|\boldsymbol{\varphi} - \boldsymbol{\varphi}_h\|$	e.o.c	$\|\boldsymbol{F} - \boldsymbol{F}_h\|$	e.o.c	$\|\boldsymbol{\varphi} - \boldsymbol{\varphi}_h^*\|$	e.o.c
1	0.5000	2.22e-03	-	2.37e-02	-	2.43e-03	-
	0.2500	6.75e-04	1.72	7.86e-03	1.59	3.06e-04	2.99
	0.1667	3.18e-04	1.85	4.25e-03	1.51	1.20e-04	2.30
	0.1250	1.87e-04	1.84	2.72e-03	1.56	6.07e-05	2.38
	0.0833	8.66e-05	1.90	1.44e-03	1.56	2.26e-05	2.43
	0.0625	4.96e-05	1.94	9.[illegible]e-04	1.56	1.10e-05	2.48
2	0.5000	2.72e-04	-	[illegible].40e-03	-	1.14e-04	-
	0.2500	2.61e-05	3.38	3.64e-04	2.72	1.35e-05	3.42
	0.1667	7.81e-06	2.98	1.19e-04	2.75	2.84e-06	3.84
	0.1250	3.26e-06	3.04	5.20e-05	2.89	9.00e-07	3.99
	0.0833	9.51e-07	3.04	1.59e-05	2.91	1.78e-07	3.99
	0.0625	3.97e-07	3.03	6.88e-06	2.92	5.70e-08	3.97
3	0.5000	1.7[illegible]e-05	-	7.08e-04	-	3.64e-05	-
	0.2500	4.47e-06	3.42	3.08e-05	4.52	6.40e-07	5.83
	0.1667	8.87e-07	3.99	5.71e-06	4.15	6.24e-08	5.74
	0.1250	2.88e-07	3.90	2.07e-06	3.53	1.59e-08	4.74
	0.0833	5.81e-08	3.95	5.06e-07	3.47	2.67e-09	4.40
	0.0625	1.85e-08	3.97	1.84e-07	3.50	7.63e-10	4.36

Table 2: History of convergence of the HDG method for the sinusoidally loaded plate, and for a nearly-incompressible material ($\nu = 0.49999$).

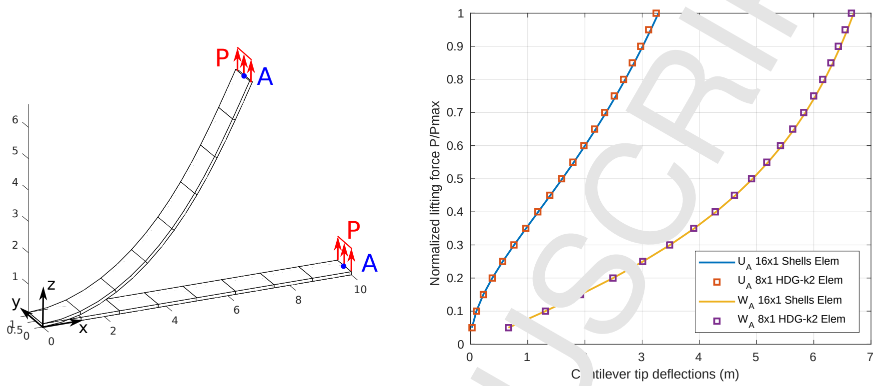

Figure 3: Cantilever subjected to a lifting force. Left: cantilever undeformed, and under the maximum deformation, for a 8×1 mesh and with $k = 2$. Right : corresponding deflections of the cantilever's tip, recorded at point A. Results from 8×1 S4R shell elements are reported as a reference.

$\boldsymbol{t} = \mathcal{A}^{-1}\mathbf{P}$ with $\mathcal{A}$ the cantilever end area. The adaptive stabilization (19) is used with $L_c = 1\,\mathrm{m}$. The displacements of the lifted tip are reported on Fig. 3 and show a good agreement when compared to S4R shell elements when quadratic HDG elements ($k = 2$) are used. For linear HDG elements, at least 50 elements would have been necessary to get reasonably accurate results.

5.2.2. Cantilever subjected to a bending moment

The following example is a very popular benchmark considered by [59, 60] and others. The purpose of this benchmark is to test the modeling of large bending deformations for thin beams. We consider the same cantilever as before, but slightly longer ($L = 12\,\mathrm{m}$). Instead of a lifting force, the cantilever is now subject to a maximum bending moment $M_{\max} = \frac{50\pi}{3} \times 1000\,\mathrm{kN\,m^{-1}}$ at its other end (see Fig. 4, left). The bending moment is numerically applied as a Neumann boundary condition, prescribing on the tip surface an equivalent normal traction $\boldsymbol{t}$ varying linearly in the vertical direction. For an applied moment $0 \leq M \leq M_{\max}$, analytical solutions give the horizontal and vertical displacements of a tip point A located on the mean surface

$$U_A = L\frac{M_{\max}}{2\pi M}\sin\left(\frac{2\pi M}{M_{\max}}\right) + L \quad \text{and} \quad W_A = L\frac{M_{\max}}{2\pi M}\left(1 - \cos\left(\frac{2\pi M}{M_{\max}}\right)\right).$$

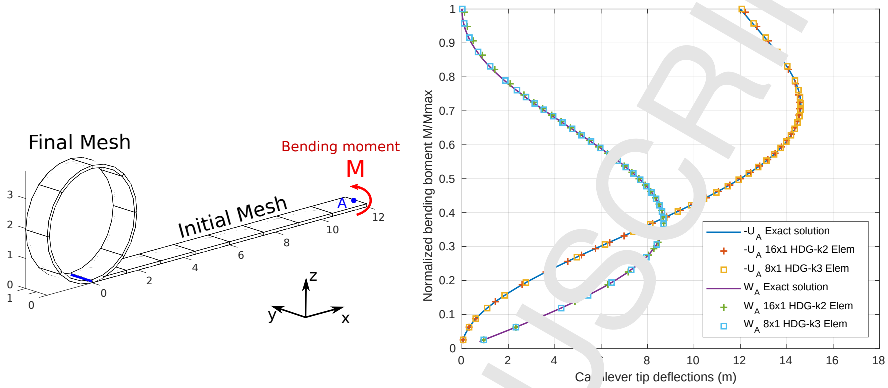

Figure 4: Cantilever subjected to a bending moment. Left : cantilever undeformed, and under the maximum deformation, for a 8×1 mesh and with $k = 3$. Right : corresponding deflections of the cantilever's tip, recorded at point A. The exact solution is also shown, as well as a solution performed with a 16×1 mesh and $k = 2$.

The displacements of the tip are reported on Fig. 4, right, for both quadratic and cubic HDG elements, and the adaptive stabilization is the same as before. Converged results are obtained with a 16×1 mesh for quadratic elements and a 8×1 mesh for cubic elements.

5.3. Shell problems

5.3.1. Slit annular plate

The slit annular plate benchmark checks the accuracy of the combined bending and torsional deformations. Let us consider a slit annular plate of internal radius $r = 6\,\mathrm{m}$, external radius $R = 10\,\mathrm{m}$ and thickness $t = 0.03\,\mathrm{m}$ clamped at one end of the slit and subjected to a lifting force $\mathbf{P}$ at the other end (see Fig. 5, left). The $\mathbf{P}$ force with a maximum magnitude $0.8\,\mathrm{kN}$ is applied as a traction distributed over the slit end. The material parameters are $E = 21 \times 10^6\,\mathrm{kPa}$ and $\nu = 0$. We use $L_c = 1\,\mathrm{m}$, for the adaptive stabilization. For $k = 2$, the converged HDG results on a 6×30 mesh are in excellent agreement with the reference results computed with S4R shell elements (see Fig. 5, right).

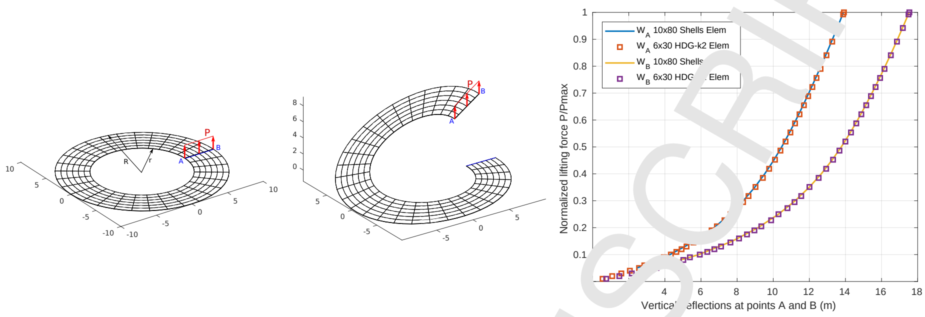

Figure 5: Slit annular plate subjected to a lifting force. Left : plate undeformed, and under the maximum deformation, for a 6×30 mesh and with $k = 2$. Right : corresponding deflections of the lifted end, recorded at points A and B. A reference solution computed with a mesh of 10×80 S4R shell elements is also displayed.

5.3.2. Hemispherical shell with a 18^o hole

We present here the hemispherical shell problem considered by [61, 59, 27, 60, 26, 30] and others. This benchmark tests the ability to model combined large membrane and bending deformations in double-curved shell geometries.

The structure studied is a hemispherical shell with a 18^o centered circular hole. The material properties considered are $E = 6.825 \times 10^7$ kPa, $\nu = 0.3$. The radius of the hemisphere is $R = 10$ m and its thickness is $t = 0.04$ m. The shell is subjected to four alternating radial point forces, whose magnitude are $P = 400$ kN each (see Fig. 6, left). Due to the symmetries, the computational domain is only one quarter of the full problem. Symmetry boundary conditions are then applied (see Fig. 6, right).

The HDG solution for the deflections at the nodes A and B is computed with an 8×8 mesh, using polynomial order $k = 3$ and the adaptive $\boldsymbol{\tau}$ is computed with $L_c = 3$ m. For $k = 2$, converged results are obtained for a finer mesh of 20×20 elements. The results, given in Fig. 7 show an excellent agreement with the reference solution presented in [60] which is computed with 16×16 S4R shell elements.

In order to illustrate how the right amount of stabilization mitigates the locking pathologies, we also display the displacements obtained by using a very large $\boldsymbol{\tau}_m = 1000\mu$ instead of the adaptive $\boldsymbol{\tau}$, for the same quadratic mesh.

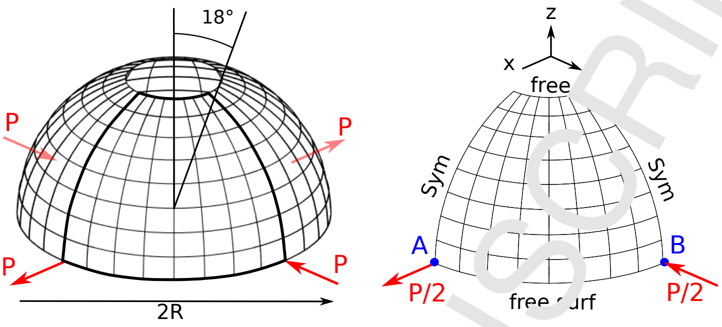

Figure 6: Left : cylindrical shell dimensions and applied point forces. Right : reduced computational domain and boundary conditions. Here a 8 × 8 mesh is used, with $k = 3$.

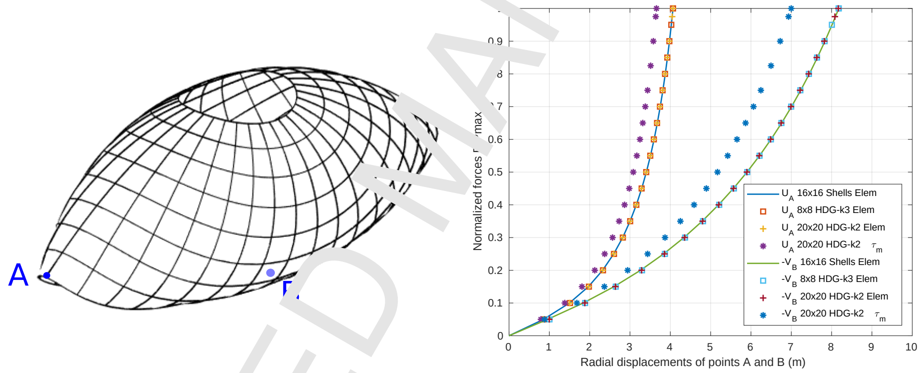

Figure 7: Left : deformed hemispherical shell under maximum load. Right : radial deflections of points A and B are compared against a shell elements reference result.

Clearly the deflections become severely underestimated and the HDG model *locks*. As continuous Galerkin can be regarded as a limit of HDG [48] when $\boldsymbol{\tau} \to \infty$ we expect that similar deflections would be obtained with a standard continuous Galerkin method.

5.3.3. Pullout of an open-end cylinder

The pullout of a cylindrical shell with free edges is a benchmark used to check the accuracy in modeling large bending and membrane deformations. We consider a cylinder of radius $R = 4.953$ m, length $L = 10.35$ m and thickness $t = 0.094$ m, subjected to a pair of symmetrical radial pulling forces $\mathbf{P}$ whose

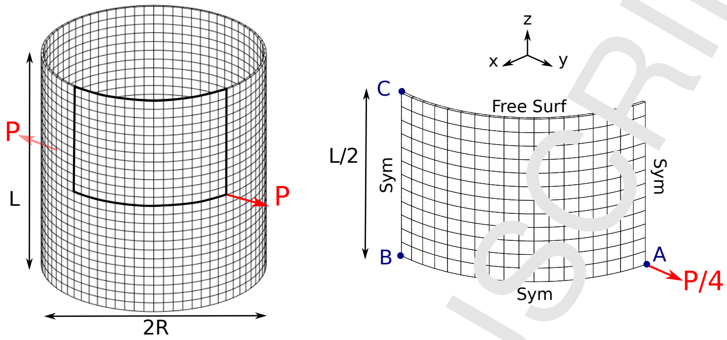

Figure 8: Left : cylinder dimensions and applied point forces. Right : reduced computational domain and boundary conditions. Here a 12×18 mesh is represented, and $k = 2$.

maximal magnitudes are $\mathsf{P}_{\max} = 4 \times 10^4$ kN. Material properties are $E = 10.5 \times 10^6$ kPa and $\nu = 0.3125$. Owing to symmetries, only one eighth of the structure is modeled, using the suitable symmetric boundary conditions (see Fig. 8). Based on the mesh size, the characteristic length is $L_c = 0.4$ m.

Accurate results are obtained with a 12×18 mesh for quadratic ($k = 2$) elements (see Fig. 8 and 9). For $k = 3$, a similar accuracy is obtained with a 8×12 mesh. Note that, although refining the mesh does not lead to a significant modification ($< 1\%$) of the deflection of points B and C, it will slightly increase the deflection of point A, where the force is applied. This is due to a local 3D effect, which is amplified when the support of the point force shrinks.

The displacements of all three points A, B and C, match very well the reference solution computed with 24×36 S4R shell elements (see Fig. 9).

The Table C.6 gives the solver metrics as well as a comparison between different stabilizations. Interestingly, by using the same characteristic length L_c, most stabilization functions would fail to reach $\mathsf{P}_{\max}$. By using (19), a sufficient amount of stabilization is provided near the point force, and the number of load increments is $n_{\text{inc}} = 26$. However, Fig. 9, right, displays more data points for the sake of comparison by using artificially lower load increments.

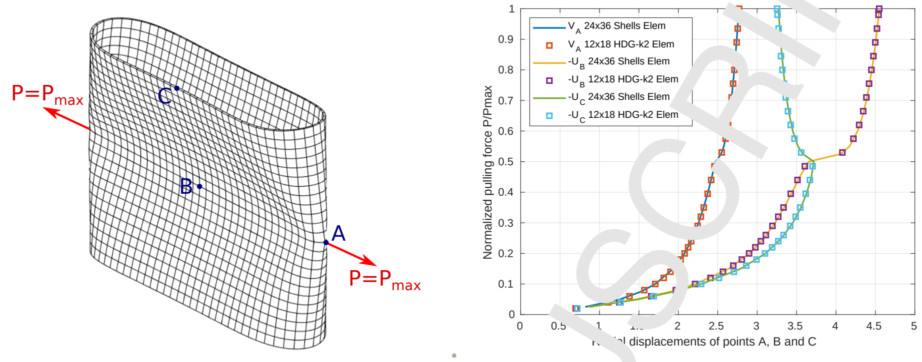

Figure 9: Left : deformed cylindrical shell under maximum load, for a 12×18 quadratic elements mesh. Right : radial deflections at points A, B and C are compared against a shell elements reference result.

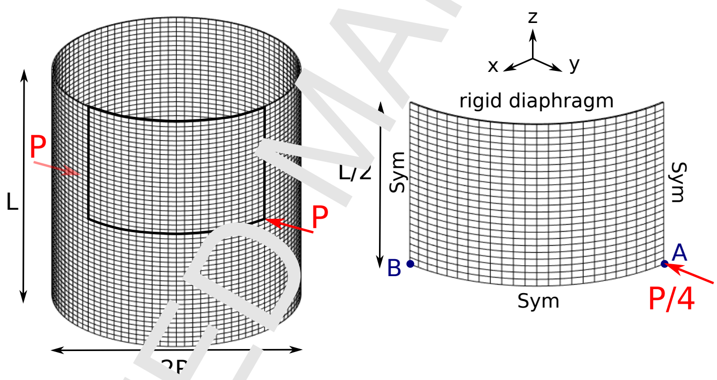

Figure 10: Left : cylindrical shell dimensions and applied point forces. Right : reduced computational domain and boundary conditions. Here a 48×48 mesh is used, and $k = 2$.

5.3.4. Pinched cylinder with end diaphragms

The pinched cylindrical shell is one of the most demanding classical benchmark that can be found in the literature. Simo et al [62] explained that the difficulty comes from the inextensional bending and the complex membrane states of stress. The deformations involve the development of wrinkles, which are quite hard to model with low order elements or with coarse meshes, and most of the finite elements formulations have a hard time converging for this particular example (see for instance [28]).

We consider a cylinder represented on Fig. 10, whose radius is $R = 10\,\text{m}$,

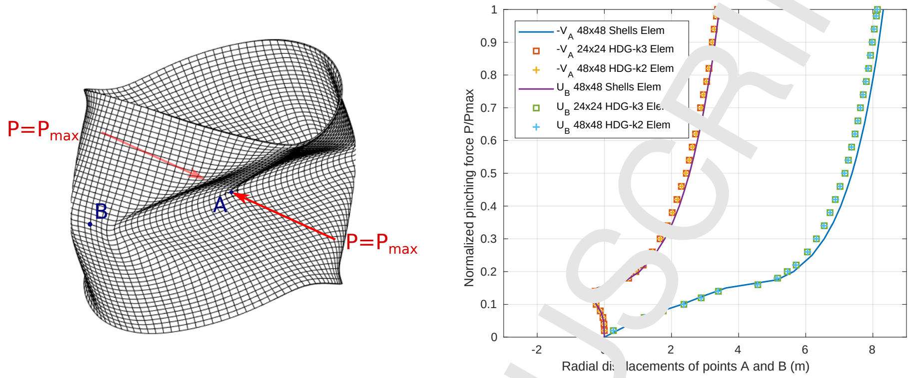

Figure 11: Left : deformed cylindrical shell under maximum load, for a 24×24 cubic elements mesh. Right : corresponding radial deflections at points A and B being compared against a 48×48 S4R shell elements reference result (only selected data points are displayed).

length $L = 20\,\mathrm{m}$ and thickness $t = 0.$[illegible] m. The cylinder is subjected to a pair of symmetrical radial pinching forces P whose maximal magnitudes are $\mathsf{P}_{\max} = 120\,\mathrm{kN}$. The cylinder is closed with rigid diaphragms on its ends such that the ends points can only move in the z-direction. Thanks to the different symmetries in the problem, only one octant of the geometry needs to be modeled (see Fig. 10). Converged results are obtained when the octant is meshed with a 48×48 mesh for polynomial order $k = 2$. Alternatively, for $k = 3$, the results converge for a coarser 24×24 mesh. We picked $L_c = 0.25\,\mathrm{m}$ for the stabilization. Without the proper adaptive mechanism, most stabilizations fail for that case or induce a non-physical oscillatory behavior (see table C.7).

The computed radial deflections at points A and B show a globally good agreement with the S4R solution, although HDG predicts slightly smaller deflections for the point A at large deformations. Such level of discrepancies between numerical methods are however common for the pinched cylinder case (see [63, 64]). For this specific case, the Newton-Raphson procedure converges rather slowly and we increased the maximum number of Newton iterations to 50. For $k = 2$, the total number of load increments and Newton iterations are $n_{\text{inc}} = 50$ and $n_{\text{tot}} = 468$ respectively, which is comparable to the the S4R results (respectively 70 and 406, according to [60]).

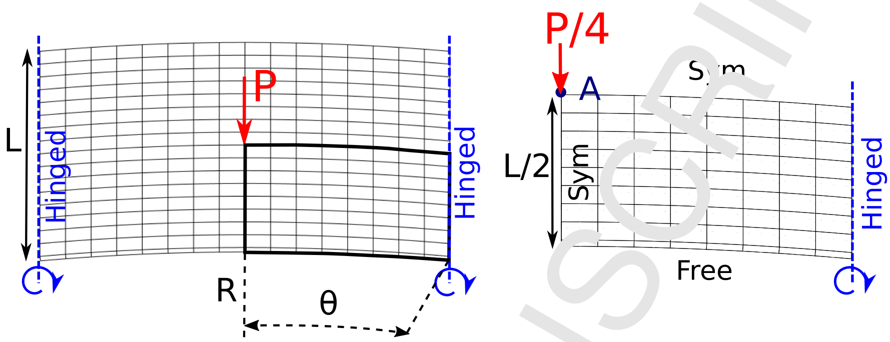

Figure 12: Left : full roof structure with boundary conditions. Right : quarter of the full structure actually modeled.

5.3.5. *Hinged roof*

The following numerical experiment was introduced first in [65] and since then has been extensively studied as an example of snapping instabilities.

The roof structure is a section of cylindrical shell hinged on two sides, with radius $R = 25.4\,\mathrm{m}$, length $L = 2.54\,\mathrm{m}$, and angle $\theta = 0.1\,\mathrm{rad}$ (see Fig. 12). A vertical point load $\mathsf{P}_{\max} = 300\,\mathrm{N}$ is applied at the center of the structure. The material properties are $E = 3102.75\,\mathrm{kPa}$ and $\nu = 0.3$. Only one quarter of the full structure is modeled and converged results are obtained using an 8×8 quadratic element mesh. The characteristic length is therefore $L_c = 0.3\,\mathrm{m}$.

As a side note, in order to implement the *hinged* boundary conditions we found it more practical to strongly enforce all the Dirichlet-like boundary conditions (including the hinged ones). Therefore, for this specific numerical experiment, the variational principle (5), the HDG trace equation (13d) and the space of traces (3) should all be modified accordingly.

The behavior of the structure changes dramatically with the thickness of the roof. For a thick roof, i.e $t = 127\,\mathrm{mm}$, the structure exhibits a snap-through instability, whereas for a thinner roof, i.e. $t = 63.5\,\mathrm{mm}$, a snap-back instability is observed (see Fig. 13).

The Newton-Raphson algorithm typically fails on either configuration, because the Jacobian matrix in (23) becomes singular for loads smaller then $\mathbf{P}_{\max}$.

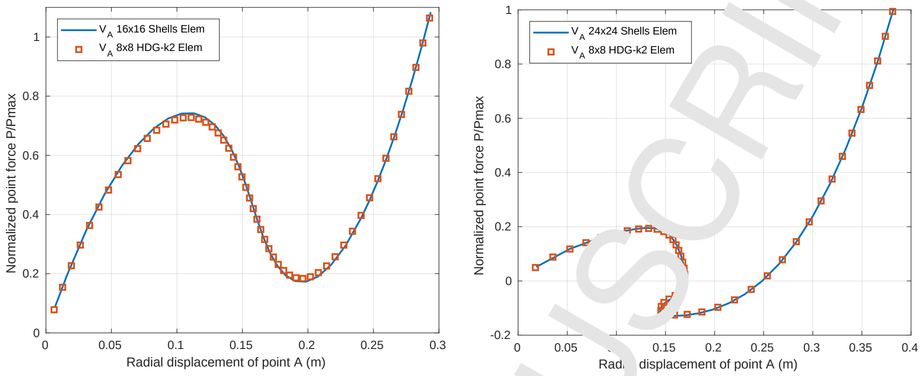

Figure 13: Results obtained for a 8×8 mesh with $k = 2$. Left : radial deflection of point A for $t = 127\,\mathrm{mm}$, and comparison against a 16×16 S4R shell element result. The snap-through instability arises at 72.5% of the maximum load. Right : radial deflection of point A for $t = 63.5\,\mathrm{mm}$ compared against a 24×24 shell elements result. The first snap-through instability arises around 20% of the total load while the first snap-back instability appears around 1%.

We therefore use the Arc-Length algorithm 2 with parameters

$$\psi = \frac{1}{0.1\,\|\mathbf{P}_{\max}\|} \quad \text{and} \quad \Delta l = \begin{cases} 0.2\,\mathrm{m} & \text{if } t = 127\,\mathrm{mm} \\ 0.3\,\mathrm{m} & \text{if } t = 63.5\,\mathrm{mm} \end{cases} \tag{32}$$

The converged deflections at the center of the roof, shown on Fig. 13, essentially agree with the results obtained using ABAQUS standard S4R shell elements. The full snapping behaviors are properly modeled, and the instabilities are handled correctly by the Arc-Length method. Although the agreement is excellent for the thin roof, the snap-through occurs slightly earlier with HDG for the thick roof (72.5% of the total load instead of 74% for the shell elements).

5.4. *Thick-thin structure*

We now present an simple example of a thick solid-thin shell structure. The structure is composed of a thick pillar supporting an arch with a variable thickness (see Fig. 14). The thickness of the arch is 0.5 m at the root, and 0.025 m at the tip. The 2D geometry presented on Fig. 14 is extruded 0.5 m in the normal direction. The base of the pillar is clamped and an uniform pressure $\mathbf{P}$

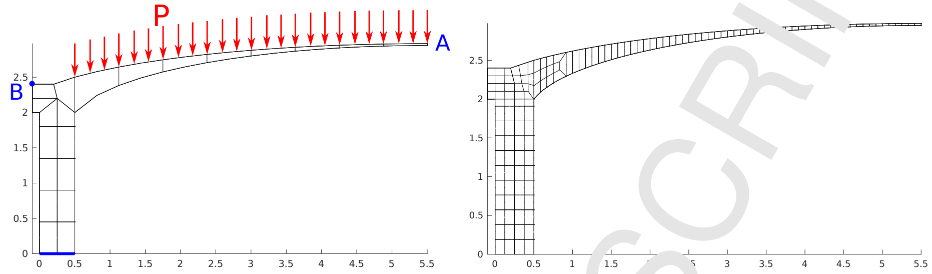

Figure 14: Meshes used for the thick-thin arch structure. Left: coarse mesh modeling the arch 8 elements, with $k = 2$. The pressure is applied on the arch only. Right: fine mesh modeling the arch with 68 elements, and with $k = 3$. Reference results are computed using the fine mesh.

	U_A	V_A
Coarse Mesh	-0.1411	-0.9796
Reference Mesh	-0.1405	-0.9453

Table 3: Horizontal and vertical deflections of the point A obtained with a coarse mesh and a fine reference mesh.

is applied on the upper side of the arch such that the total integrated pressure is equivalent to a 100 kN force. A neo-Hookean model with $E = 200\,\text{GPa}$ and $\nu = 0.28$ is used.

Two discretizations are considered. The first one uses a coarse mesh with $k = 2$, modeling the arch with only 8 elements. Therefore the element aspect ratios vary from almost 1 (near the pillar) to 20 (near the tip). A second discretization, used to generate the reference solution, makes use of a finer mesh with $k = 3$. By using 68 elements for modeling the arch, the aspect ratio is kept small for all the elements. For both meshes, the final deflections of the arch tip (point A) are recorded. For both numerical simulations, we will take $L_c = 1\,\text{m}$.

The results under maximum load are reported on table 3. The coarse mesh exhibit a good accuracy for the deflection of point A although the vertical displacement is slightly overestimated. This example confirms that our approach, making use of volumetric elements, is indeed suitable for thin-thick structures.

5.5. Realistic structure: wing mesh

While all the applications presented so far can be easily run on a single CPU core, many complex realistic structures will need more computational resources. We have therefore implemented a parallel version of our method that make use of the HDG solver previously developed for CFD applications (see [66]). The global system (27) is solved by a parallel Generalized Minimal Residual (GMRES) method, using a block incomplete LU (BILU) factorization as a left preconditioner.

We have used this parallel solver to compute the deformations of a complex aircraft wing structure, comprising an upper and a lower skin, spars and multiple ribs (see [67] for a precise description of the geometry). The mesh consists of 13382 hexahedra whose aspect ratio varies between 1.1 and 75. Although the parallel solver can solve nonlinear problems, we consider here a linear application in order to provide some indication of the the code efficiency in solving a single linearized step. The linear elastic moduli are $E = 70\,\mathrm{GPa}$ and $\nu = 0.35$ for the whole structure. The wing root is clamped, and a traction equivalent to a $200\,\mathrm{kN}$ lifting force is applied on the the wing tip. A polynomial degree $k = 2$ is considered, and the characteristic length is set to $L_c = 0.1\,\mathrm{m}$. Fig. 15 shows the deformed wing as well as the distribution of the Von Mises stresses.

The parallel code ran on 2 Haswell nodes of the NASA Pleiades supercomputer, each node being a 12-cores Intel Xeon E5-2680v3 at 2.50 GHz. The resolution of the linear system lasted approximately 10 min, with GMRES needing around 1500 iterations to converge. The BILU preconditioner, which is more suited for hyperbolic problems, is probably the cause of the relatively high number of iterations. We believe that a specific preconditioner for elastostatic HDG applications should therefore be developed.

6. Conclusion

We have presented a HDG method for solving nonlinear elastic structures including thin components. Our approach models the full 3D structure and does

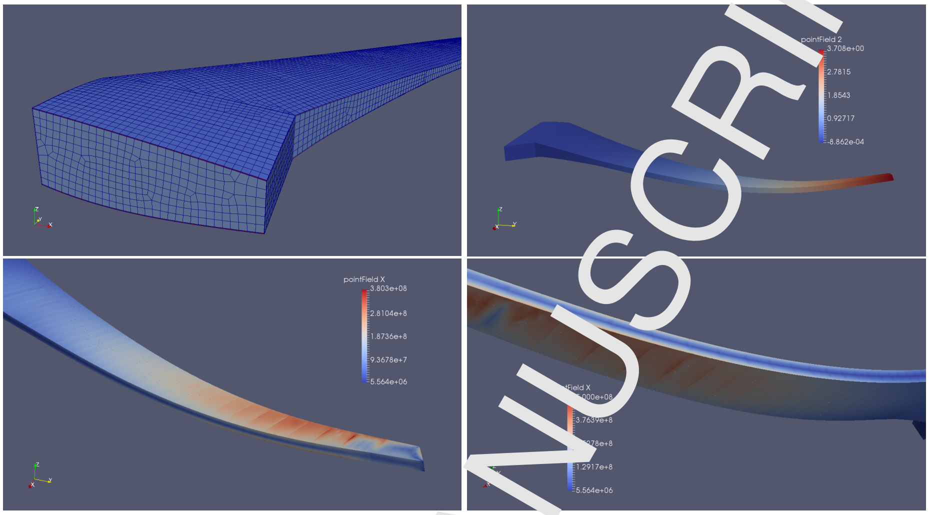

Figure 15: Static analysis of an airliner wing structure. Top left: mesh of the wing – zoom on the wing root. Top right: vertical displacements at maximum load. Bottom: Von Mises stress distribution on the upper skin (left) and on the lower skin (right). The concentration of stresses at the connection between the ribs and the skin is clearly visible, showing the strengthening effect of the ribs.

not require typical approximations used in shell theories. The size of the global systems of equations can be significantly reduced when thin structures are modeled, which is an appealing feature compared to other volumetric approaches. Moreover, optimal rate of convergence for the deformation is observed, and the postprocessing provides between one half and one full extra order of convergence at a negligible cost. We have validated our method studying classical benchmarks for both cantilever and shell structures. Our numerical results show that when quadratic or cubic polynomial approximations are used, the method is free from locking and gives accurate converged results. The HDG approach is therefore worth considering for modeling finite deformations of shell structures.

Acknowledgments

The authors gratefully acknowledge the NASA for supporting this work under grant number NNX16AP15A. The authors particularly thanks Steven J.

Massey for providing the wing mesh and an access to the NAS computational resources.

The first author warmly thanks Pablo Fernandez del Campo for his help regarding the development and the use of the parallel version of the code.

Appendix A. Loading algorithm

Algorithm 1 Load incrementation algorithm

Require: Initialize $\boldsymbol{F}_h^0 = \mathbf{Id}$; $(\boldsymbol{\varphi}_h^0, \hat{\boldsymbol{\varphi}}_h^0) = \boldsymbol{X}$ (initial geometry)

Require: Initialize $\lambda = \lambda_{\text{init}}$; $\mathsf{P} = \lambda \mathsf{P}_{\max}$; $n_{\text{tot}} = 0$; $n_{\text{inc}} = 0$; $n_{\text{it1}} = 0$

while $\lambda < 1$ **do**

  Assign $\lambda := \lambda + \Delta\lambda$

  Assign $\mathsf{P} := \lambda \mathsf{P}_{\max}$

  Call of the Newton-Raphson procedure, converging in n_{it} iterations

  Compute $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h, n_{\text{it}}) = \text{Newton-Raphson}(\boldsymbol{F}_h^0, \boldsymbol{\varphi}_h^0, \hat{\boldsymbol{\varphi}}_h^0, \mathsf{P})$

  if $n_{\text{it}} \leq n_{\max}$ **then**

    Convergence of Newton-Raphson

    Assign $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h) := (\boldsymbol{F}_h^0, \boldsymbol{\varphi}_h^0, \hat{\boldsymbol{\varphi}}_h^0)$

    if $n_{\text{it}} < 5$ **and** $n_{\text{it1}} < 5$ **then**

      Assign $\Delta\lambda := 1.5\ \Delta\lambda$

    end if

    Assign $n_{\text{tot}} := n_{\text{tot}} + n_{\text{it}}$

    Assign $n_{\text{inc}} := n_{\text{inc}} + 1$

    Assign $n_{\text{it1}} := n_{\text{it}}$

  else

    No or poor convergence of Newton-Raphson

    Assign $\lambda := \lambda - \Delta\lambda$

    Assign $\Delta\lambda := 0.5\ \Delta\lambda$

  end if

end while

return $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h, n_{\text{it}}, n_{\text{inc}})$

Appendix B. Arc-Length Algorithm

Algorithm 2 Arc-Length($\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h, \lambda, \Delta l, \psi$)

Require: Initialize $\Delta\hat{\boldsymbol{\varphi}}_h = 0$; $\Delta\lambda = 0$

  Compute $(\mathbb{K}, R)$ from $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h, \lambda\mathsf{P}_{\max})$ using (23)-(25)

  while residual>tol **do**

    Compute $\delta\hat{\boldsymbol{\varphi}}_h^R = \mathbb{K}^{-1}R$

    Compute $\delta\hat{\boldsymbol{\varphi}}_h^P = -\mathbb{K}^{-1}\mathsf{P}_{\max}$

    $\delta\lambda = \text{solve}\left[(\Delta\hat{\boldsymbol{\varphi}}_h + \delta\hat{\boldsymbol{\varphi}}_h^R + \delta\lambda\delta\hat{\boldsymbol{\varphi}}_h^P)^2 + \psi^2(\Delta\lambda + \delta\lambda)^2\mathsf{P}_{\max}^2 = \Delta l^2\right]$

    Compute $\delta\hat{\boldsymbol{\varphi}}_h = \delta\hat{\boldsymbol{\varphi}}_h^R + \delta\lambda\delta\hat{\boldsymbol{\varphi}}_h^P$

    Assign $\Delta\hat{\boldsymbol{\varphi}}_h := \Delta\hat{\boldsymbol{\varphi}}_h + \delta\hat{\boldsymbol{\varphi}}_h$

    Assign $\Delta\lambda := \Delta\lambda + \delta\lambda$

    Compute $(\delta\boldsymbol{F}_h, \delta\boldsymbol{\varphi}_h)$ from $\delta\hat{\boldsymbol{\varphi}}_h$ using (23)

    Assign $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h) := (\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h) + (\delta\boldsymbol{F}_h, \delta\boldsymbol{\varphi}_h, \delta\hat{\boldsymbol{\varphi}}_h)$

    Compute $(\mathbb{K}, R)$ from $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h, \lambda\mathsf{P}_{\max})$ using (23)-(25)

    Compute residual $= \|R\|$

  end while

  return $(\boldsymbol{F}_h, \boldsymbol{\varphi}_h, \hat{\boldsymbol{\varphi}}_h, \Delta\hat{\boldsymbol{\varphi}}_h, \Delta\lambda)$

Appendix C. Comparisons of several stabilization functions and solver metrics

We present here some comparative data to assess the relative performance of the five stabilization functions mentioned in subsection 3.3.

Note that we tested a slightly different version of stabilization (17). Indeed, we instead implemented

$$\boldsymbol{\tau} = \boldsymbol{\tau}_0 - \frac{\beta}{L_c}\rho_{\min}\left(\frac{\partial \boldsymbol{P}_h}{\partial \boldsymbol{F}_h}\right)\boldsymbol{I} \tag{C.1}$$

for the following reason. The original stabilization (17), presented for the DG method [51], makes use of a very small $\boldsymbol{\tau}_0$ (with even $\boldsymbol{\tau}_0 = 0$ for most numerical examples) such that the scaling factor β essentially amplifies $\rho_{\min}$. This is however impossible in a HDG context since a minimum amount of stabilization is always required for the method to work, even when $\rho_{\min} = 0$. For our HDG method, a good estimation of $\boldsymbol{\tau}_0$ for moderate strains is given by (14), which is noticeably larger than $\rho_{\min}$ for the applications considered in this paper. Therefore, it is more relevant to consider (C.1), where only $\rho_{\min}$ is amplified by β, such that the contribution of $\rho_{\min}$ to the stabilization can be isolated and assessed. When not specified, we use the default value $\beta = 1$.

In the following tables, we arbitrarily define the *slight locking* pathology as an underestimation of the displacements by less than 10% at maximum load. And we simply call *locking* the larger underestimations. When the loading algorithm 1 fails, the arc-length 2 is not expected to provide a more stable solution, except when the displacements are non-monotonic functions of the load, which happens only for the hinged roof case.

For the cantilever cases 5.2 and the slit plate 5.3.1, all the stabilization methods provide roughly the same amount of penalization since $\nu = 0$ and the strains remain moderate. For the first cantilever problem 5.2.1, all stabilizations ensure the convergence of the algorithm 1 with $n_{\text{inc}} = 10$ and $n_{\text{tot}} = 65$. For the second cantilever problem, some of the stabilizations lead to a slight locking pathology, as reported on table C.4. For the slit plate case, all the stabilizations work well with $42 \leq n_{\text{inc}} \leq 53$, and $250 \leq n_{\text{tot}} \leq 270$ without any locking.

stabilization	k	mesh	L_c	n_{inc}	n_{tot}	reach $\mathsf{P}_{\max}$	notes
(14)	2	16×1	1	23	678	✓	-
(14)	3	8×1	1	141	1652	✓	-
(15)	2	16×1	1	21	617	✓	slight locking
(15)	3	8×1	1	123	1283	✓	slight locking
(16)	2	16×1	1	22	652	✓	slight locking
(16)	3	8×1	1	139	1635	✓	slight locking
(C.1)	2	16×1	1	23	695	✓	-
(C.1)	3	8×1	1	140	1633	✓	-
(19)	2	16×1	1	23	6[illegible]	✓	-
(19)	3	8×1	1	14[illegible]	1629	✓	-

Table C.4: Cantilever bent into a ring 5.2.2: solver metrics for several stabilization functions. Here $n_{\max} = 50$ has been used for all runs.

The differences are more noticeable with the hemispheric shell, the pullout cylinder and the pinched cylinder cases, whose results are reported in tables C.5, C.6 and C.7 respectively. All these benchmarks have in common the concentration of large strains in localized areas (near the applied forces and wrinkles). Most stabilizations fail at some point, while (19) appears to work well. Interestingly, by using the minimum eigenvalue, the stabilization (C.1) may also work provided that the coefficient β is tuned. However, β appears to be case-dependent. If L_c is chosen too large for the pinched cylinder, the adaptive stabilization may still work, but the solution shows some strong spurious oscillations that require then the use of the arc-length algorithm 2.

The hinged roof 5.3.5, although having a very nonlinear response with respect to the load, involves only small strains. Therefore all the penalization functions perform equally well with $n_{\text{inc}} = 54$ and $n_{\text{tot}} = 108$ for the thick roof, and $n_{\text{inc}} = 48$ and $n_{\text{tot}} = 128$ for the thin roof. Note that these numbers largely depend on the choice of the user-defined characteristic length Δl.

stab.	β	k	mesh	L_c	n_{inc}	n_{tot}	reach $\mathsf{P}_{\max}$	notes
(14)	-	2	20×20	3	15	116		fails at 0.48 $\mathsf{P}_{\max}$
(14)	-	3	8×8	3	26	181		fails at 0.88 $\mathsf{P}_{\max}$
(15)	-	2	20×20	3	14	125		fails at 0.41 $\mathsf{P}_{\max}$
(15)	-	3	8×8	3	29	209		fails at 0.89 $\mathsf{P}_{\max}$
(16)	-	2	20×20	3	25	205		fails at 0.40 $\mathsf{P}_{\max}$
(16)	-	3	8×8	3	31	289		fails at 0.35 $\mathsf{P}_{\max}$
(C.1)	1	2	20×20	3	23	157		fails at 0.48 $\mathsf{P}_{\max}$
(C.1)	10	2	20×20	3	28	177		fails at 0.56 $\mathsf{P}_{\max}$
(C.1)	100	2	20×20	3	21	128	✓	-
(C.1)	1	3	8×8	3	31	206		fails at 0.89 $\mathsf{P}_{\max}$
(C.1)	10	3	8×8	3	31	189		fails at 0.89 $\mathsf{P}_{\max}$
(C.1)	100	3	8×8	3	21	131	✓	-
(19)	-	2	20×20	3	20	128	✓	-
(19)	-	3	8×8	3	21	132	✓	-

Table C.5: Hemispherical shell case 5.3.2: solver metrics for several stabilization functions.

stab.	β	k	mesh	L_c	n_{inc}	n_{tot}	reach $\mathsf{P}_{\max}$	notes
(14)	-	2	12×18	0.4	24	278		fails at $0.41\,\mathsf{P}_{\max}$
(14)	-	3	8×12	0.4	14	190		fails at $0.18\,\mathsf{P}_{\max}$
(15)	-	2	12×18	0.4	25	165		fails at $0.51\,\mathsf{P}_{\max}$
(15)	-	3	8×12	0.4	34	342		fails at $0.54\,\mathsf{P}_{\max}$
(16)	-	2	12×18	0.4	20	201		fails at $0.17\,\mathsf{P}_{\max}$
(16)	-	3	8×12	0.4	10	110		fails at $0.08\,\mathsf{P}_{\max}$
(C.1)	1	2	12×18	0.4	27	286		fails at $0.45\,\mathsf{P}_{\max}$
(C.1)	10	2	12×18	0.4	36	370		fails at $0.81\,\mathsf{P}_{\max}$
(C.1)	100	2	12×18	0.4	32	171	✓	-
(C.1)	1	3	8×12	0.4	19	264		fails at $0.20\,\mathsf{P}_{\max}$
(C.1)	10	3	8×12	0.4	42	529		fails at $0.67\,\mathsf{P}_{\max}$
(C.1)	100	3	8×12	0.4	35	212	✓	-
(19)	-	2	12×18	0.4	26	227	✓	-
(19)	-	3	8×12	0.4	38	255	✓	-

Table C.6: Pull-out cylinder case 5.3.3: solver metrics for several stabilization functions. Here, $\lambda_{\mathrm{init}} = 0.02$ have been used for all runs.

stab.	β	k	mesh	L_c	n_{inc}	n_{tot}	algo	P_{max}	notes
(14)	-	2	48×48	0.25	34	802	1		fails at 0.35 P_{max}
(14)	-	3	24×24	0.25	31	770	1		fails at 0.16 P_{max}
(15)	-	2	48×48	0.25	53	855	1	✓	oscillatory, slight locking
(15)	-	3	24×24	0.25	50	[illegible]	1	✓	oscillatory, locking
(16)	-	2	48×48	0.25	[illegible]	1138	1		fails at 0.18 P_{max}
(16)	-	3	24×24	0.25	33	761	1		fails at 0.15 P_{max}
(C.1)	1	2	48×48	0.25	[illegible]	540	1		fails at 0.76 P_{max}
(C.1)	10	2	48×48	0.25	68	601	1	✓	-
(C.1)	100	2	48×48	0.25	64	557	1	✓	-
(C.1)	1	3	24×24	0.25	23	644	1		fails at 0.20 P_{max}
(C.1)	10	3	24×24	0.25	156	2474	1		fails at 0.74 P_{max}
(C.1)	100	3	24×24	0.25	178	1895	1	✓	oscillatory
(19)	-	2	48×48	0.25	50	468	1	✓	-
(19)	-	3	24×24	0.25	50	1056	1	✓	-
(19)	-	2	48×48	0.50	48	573	2	✓	oscillatory

Table C.7: Pinched cylinder case 5.3.4: solver metrics for several stabilization functions. Here, $n_{\max} = 50$ and [illegible] $= 0.02$ have been used for all runs. In the notes column, *oscillatory* means that the model develops some mesh-dependent non-physical oscillatory pattern.

References

[1] B. Cockburn, G. E. Karniadakis, C.-W. Shu, The development of discontinuous galerkin methods, in: Discontinuous Galerkin Methods, Springer, 2000, pp. 3–50.

[2] G. E. Karniadakis, C.-W. Shu, B. Cockburn, Discontinuous Galerkin Methods: Theory, Computation and Applications, Springer, 2000.

[3] J. S. Hesthaven, T. Warburton, Nodal discontinuous Galerkin methods: algorithms, analysis, and applications, Springer Science & Business Media, 2007.

[4] B. Cockburn, D. Schötzau, J. Wang, Discontinuous galerkin methods for incompressible elastic materials, Computer methods in applied mechanics and engineering 195 (25) (2006) 3184–3204.

[5] A. Ten Eyck, A. Lew, Discontinuous galerkin methods for non-linear elasticity, International Journal for Numerical Methods in Engineering 67 (9) (2006) 1204–1243.

[6] L. Noels, R. Radovitzky, A general discontinuous galerkin method for finite hyperelasticity. formulation and numerical applications, International Journal for Numerical Methods in Engineering 68 (1) (2006) 64–97.

[7] L. Noels, R. Radovitzky, Alternative approaches for the derivation of discontinuous galerkin methods for nonlinear mechanics, Journal of Applied Mechanics 74 (5) (2007) 1031–1036.

[8] I. Djoko, F. Ebobisse, A. McBride, B. Reddy, A discontinuous galerkin formulation for classical and gradient plasticity–part 1: Formulation and analysis, Computer methods in applied mechanics and engineering 196 (37) (2007) 3881–3897.

[9] L. Noels, R. Radovitzky, An explicit discontinuous galerkin method for non-linear solid dynamics: Formulation, parallel implementation and scalability

properties, International Journal for Numerical Methods in Engineering 74 (9) (2008) 1393–1420.

[10] S.-C. Soon, B. Cockburn, H. K. Stolarski, A hybridizable discontinuous Galerkin method for linear elasticity, International journal for numerical methods in engineering 80 (8) (2009) 1058–1092.

[11] N. C. Nguyen, J. Peraire, Hybridizable discontinuous galerkin methods for partial differential equations in continuum mechanics, Journal of Computational Physics 231 (18) (2012) 5955–5988.

[12] B. Cockburn, K. Shi, Superconvergent HDG methods for linear elasticity with weakly symmetric stresses, IMA Journal of Numerical Analysis 33 (3) (2013) 747–770.

[13] G. Fu, B. Cockburn, H. Stolarski, Analysis of an hdg method for linear elasticity, International Journal for Numerical Methods in Engineering 102 (3-4) (2015) 551–575.

[14] W. Qiu, J. Shen, K. Shi, An hdg method for linear elasticity with strong symmetric stresses, Mathematics of Computation 87 (309) (2018) 69–93.

[15] S. Soon, Hybridizable discontinuous galerkin method for solid mechanics, Ph.D. thesis, University of Minnesota (2008).

[16] H. Kabaria, A. J. Lew, B. Cockburn, A hybridizable discontinuous galerkin formulation for non-linear elasticity, Computer Methods in Applied Mechanics and Engineering 283 (2015) 303–329.

[17] J. Shen, Hybridizable discontinuous galerkin methods for nonlinear elasticity, Ph.D. thesis, University of Minnesota (2017).

[18] B. Cockburn, J. Shen, An algorithm for stabilizing hybridizable discontinuous galerkin methods for nonlinear elasticity, Results in Applied Mathematics (2019) 100001.

[19] T. Belytschko, A review of recent developments in plate and shell elements, Computational mechanics- Advances and trends (1986) 217–231.

[20] H. Stolarski, T. Belytschko, S.-H. Lee, A review of shell finite elements and corotational theories, Computational mechanics advances 2 (2) (1995) 125–212.

[21] H. T. Yang, S. Saigal, A. Masud, R. Kapania, A survey of recent shell finite elements, International Journal for numerical methods in engineering 47 (1-3) (2000) 101–127.

[22] S. Güzey, H. Stolarski, B. Cockburn, K. Tamma, Design and development of a discontinuous galerkin method for shells, Computer Methods in Applied Mechanics and Engineering 195 (25) (2006) 3528–3548.

[23] S. Güzey, B. Cockburn, H. Stolarski, The embedded discontinuous galerkin method: application to linear shell problems, International journal for numerical methods in engineering 70 (7) (2007) 757–790.

[24] L. Noels, A discontinuous galerkin formulation of non-linear kirchhoff–love shells, International journal for numerical methods in engineering 78 (3) (2009) 296–323.

[25] A. Ibrahimbegovic, R. L. Taylor, On the role of frame-invariance in structural mechanics models at finite rotations, Computer Methods in Applied Mechanics and Engineering 191 (45) (2002) 5159–5176.

[26] S. Klinkel, F. Gruttmann, W. Wagner, A robust non-linear solid shell element based on a mixed variational formulation, Computer methods in applied mechanics and engineering 195 (1) (2006) 179–201.

[27] K. Sze, L. Yao, A hybrid stress ans solid-shell element and its generalization for smart structure modelling. part isolid-shell element formulation, International Journal for Numerical Methods in Engineering 48 (4) (2000) 545–564.

[28] R. Hauptmann, K. Schweizerhof, A systematic development of solid-shellelement formulations for linear and non-linear analyses employing only displacement degrees of freedom, International Journal for Numerical Methods in Engineering 42 (1) (1998) 49–69.

[29] R. J. Alves de Sousa, R. P. Cardoso, R. A. Fontes Valente, J.-W. Yoon, J. J. Grácio, R. M. Natal Jorge, A new one-point quadrature enhanced assumed strain (eas) solid-shell element with multiple integration points along thicknesspart ii: nonlinear applications, International Journal for Numerical Methods in Engineering 67 (2) (2006) 160–188.

[30] M. Jabareen, E. Mtanes, A solid-shell cosserat point element (sscpe) for elastic thin structures at finite deformation, Computational Mechanics 58 (1) (2016) 59–89.

[31] O. Zienkiewicz, R. Taylor, J. Too, Reduced integration technique in general analysis of plates and shells, International Journal for Numerical Methods in Engineering 3 (2) (1971) 275–290.

[32] W. K. Liu, Y. Guo, S. Tang, T. Belytschko, A multiple-quadrature eight-node hexahedral finite element for large deformation elastoplastic analysis, Computer Methods in Applied Mechanics and Engineering 154 (1-2) (1998) 69–132.

[33] T. Belytschko, W. K. Liu, B. Moran, K. Elkhodary, Nonlinear finite elements for continua and structures, John wiley & sons, 2013.

[34] T. J. Hughes, Generalization of selective integration procedures to anisotropic and nonlinear media, International Journal for Numerical Methods in Engineering 15 (9) (1980) 1413–1418.

[35] T. J. Hughes, The finite element method: linear static and dynamic finite element analysis, Courier Corporation, 2012.

[36] J. C. Simo, M. Rifai, A class of mixed assumed strain methods and the method of incompatible modes, International journal for numerical methods in engineering 29 (8) (1990) 1595–1638.

[37] R. Hauptmann, S. Doll, M. Harnau, K. Schweizerhof, 'solid-shell' elements with linear and quadratic shape functions at large deformations with nearly incompressible materials, Computers & Structures 79 (18) (2001) 1671 – 1685. doi:https://doi.org/10.1016/S0045-7949(01)00103-1. URL http://www.sciencedirect.com/science/article/pii/S0045794901001031

[38] R. J. Alves de Sousa, R. P. Cardoso, R. A. Fontes Valente, J.-W. Yoon, J. J. Grácio, R. M. Natal Jorge, A new one-point quadrature enhanced assumed strain (eas) solid-shell element with multiple integration points along thickness: Part igeometrically linear applications, International journal for numerical methods in engineering 62 (7) (2005) 952–977.

[39] R. P. Cardoso, J. W. Yoon, M. Mahardika, S. Choudhry, R. Alves de Sousa, R. Fontes Valente, Enhanced assumed strain (eas) and assumed natural strain (ans) methods for one-point quadrature solid-shell elements, International Journal for Numerical Methods in Engineering 75 (2) (2008) 156–187.

[40] K. Rah, W. Van Paepegem, A.-M. Habraken, J. Degrieck, R. A. de Sousa, R. A. Valente, Optimal low-order fully integrated solid-shell elements, Computational Mechanics 51 (3) (2013) 309–326.

[41] T. J. Hughes, T. Tezduyar, Finite elements based upon mindlin plate theory with particular reference to the four-node bilinear isoparametric element, Journal of applied mechanics 48 (3) (1981) 587–596.

[42] E. N. Dvorkin, K.-J. Bathe, A continuum mechanics based four-node shell element for general non-linear analysis, Engineering computations 1 (1) (1984) 77–88.

[43] L. Vu-Quoc, X. Tan, Optimal solid shells for non-linear analyses of multi-layer composites. i. statics, Computer methods in applied mechanics and engineering 192 (9-10) (2003) 975–1016.

[44] M. Suri, I. Babuška, C. Schwab, Locking effects in the finite element approximation of plate models, Mathematics of Computation 64 (210) (1995) 461–482.

[45] H. Hakula, Y. Leino, J. Pitkäranta, Scale resolution, locking, and high-order finite element modelling of shells, Computer methods in applied mechanics and engineering 133 (3-4) (1996) 157–182.

[46] Y. Başar, U. Hanskötter, C. Schwab, A general high-order finite element formulation for shells at large strains and finite rotations, International Journal for Numerical Methods in Engineering 57 (15) (2003) 2147–2175.

[47] F. Celiker, B. Cockburn, H. K. Stolarski, Locking-free optimal discontinuous galerkin methods for timoshenko beams, SIAM Journal on Numerical Analysis 44 (6) (2006) 2297–2325.

[48] B. Cockburn, J. Gopalakrishnan, R. Lazarov, Unified hybridization of discontinuous galerkin, mixed, and continuous galerkin methods for second order elliptic problems, SIAM Journal on Numerical Analysis 47 (2) (2009) 1319–1365.

[49] N. C. Nguyen, J. Peraire, B. Cockburn, High-order implicit hybridizable discontinuous Galerkin methods for acoustics and elastodynamics, Journal of Computational Physics 230 (10) (2011) 3695–3718.

[50] J. Bonet, A. J. Gil, R. D. Wood, Nonlinear solid mechanics for finite element analysis: statics, Cambridge University Press, 2016.

[51] A. Ten Eyck, F. Celiker, A. Lew, Adaptive stabilization of discontinuous galerkin methods for nonlinear elasticity: Motivation, formulation, and numerical examples, Computer Methods in Applied Mechanics and Engineering 197 (45-48) (2008) 3605–3622.

[52] A. Ten Eyck, F. Celiker, A. Lew, Adaptive stabilization of discontinuous galerkin methods for nonlinear elasticity: analytical estimates, Computer Methods in Applied Mechanics and Engineering 197 (33-40) (2008) 2989–3000.

[53] P. Fernandez, A. Christophe, S. Terrana, N.-C. Nguyen, J. Peraire, Hybridized discontinuous galerkin methods for wave propagation, arXiv preprint arXiv:1807.00086.

[54] D. Systèmes, Abaqus analysis users manual, Simulia Corp. Providence, RI, USA.

[55] E. Riks, An incremental approach to the solution of snapping and buckling problems, International Journal of Solids and Structures 15 (7) (1979) 529–551.

[56] M. Crisfield, A fast incremental/iterative solution procedure that handles snap-through, Computers & Structures 13 (1-3) (1981) 55–62.

[57] D. Systèmes, Abaqus theory manual, Simulia Corp. Providence, RI, USA.

[58] G. Horrigmoe, P. G. Bergan, Nonlinear analysis of free-form shells by flat finite elements, Computer Methods in Applied Mechanics and Engineering 16 (1) (1978) 11–35.

[59] A. Saleeb, T. Chang, W. Graf, S. Yingyeunyong, A hybrid/mixed model for non-linear shell analysis and its applications to large-rotation problems, International Journal for Numerical Methods in Engineering 29 (2) (1990) 407–446.

[60] K. Sze, X. Liu, S. Lo, Popular benchmark problems for geometric nonlinear analysis of shells, Finite elements in analysis and design 40 (11) (2004) 1551–1569.

[61] R. H. Macneal, R. L. Harder, A proposed standard set of problems to test finite element accuracy, Finite elements in analysis and design 1 (1) (1985) 3–20.

[62] J. Simo, D. Fox, M. Rifai, On a stress resultant geometrically exact shell model. part ii: The linear theory; computational aspects, Computer Methods in Applied Mechanics and Engineering 73 (1) (1989) 53–92.

[63] R. Hauptmann, K. Schweizerhof, S. Doll, Extension of the solid-shell concept for application to large elastic and large elastoplastic deformations, International Journal for Numerical Methods in Engineering 49 (9) 1121–1141. doi:10.1002/1097-0207(20001130)49:9<1121::AID-NME130>3.0.CO;2-F.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/1097-0207%2820001130%2949%3A9%3C1121%3A%3AAID-NME130%3E3.0.CO%3B2-F

[64] M. W. Heinstein, J. D. Hales, J. L. Breivik, S. W. Key, Summary compilation of shell element performance versus formulation., Tech. rep., Sandia National Laboratories (2011).

[65] A. Sabir, A. Lock, The applications of finite elements to large deflection geometrically nonlinear behaviour of cylindrical shells.

[66] P. Fernandez, N. C. Nguyen, J. Peraire, The hybridized discontinuous galerkin method for implicit large-eddy simulation of transitional turbulent flows, Journal of Computational Physics 336 (2017) 308–329.

[67] G. Kenway, G. Kennedy, J. Martins, Aerostructural optimization of the common research model configuration, in: 15th AIAA/ISSMO multidisciplinary analysis and optimization conference, 2014, p. 3274.