UNIVERSIDAD DE CONCEPCIÓN

Centro de Investigación en Ingeniería Matemática (CI^{2}MA)

A Hybridizable Discontinuous Galerkin solver for the Grad-Shafranov equation

Tonatiuh Sanchez-Vizuet, Manuel Solano

PREPRINT 2017-30

SERIE DE PRE-PUBLICACIONES

A Hybridizable Discontinuous Galerkin solver for the Grad-Shafranov equation

Tonatiuh Sánchez-Vizuet[a], Manuel Solano[b]

[a]*New York University, Courant Institute of Mathematical Sciences*
[b]*Universidad de Concepción, Department of Mathematical Engineering and Center for Research in Mathematical Engineering (CI²MA)*

Abstract

In axisymmetric fusion reactors, the equilibrium magnetic configuration can be expressed in terms of the solution to a semi-linear elliptic equation known as the Grad-Shafranov equation, the solution of which determines the poloidal component of the magnetic field. When the geometry of the confinement region is known, the problem becomes an interior Dirichlet boundary value problem. We propose a high order solver based on the Hybridizable Discontinuous Galerkin method. The resulting algorithm (1) provides high order of convergence for the flux function and its gradient, (2) incorporates a novel method for handling piecewise smooth geometries by extension from polygonal meshes, (3) can handle geometries with non-smooth boundaries and x-points, (4) deals with the semi-linearity through an accelerated two-grid fixed-point iteration, and (5) is ideally suited for parallel implementations. The effectiveness of the algorithm is verified with computations for cases where analytic solutions are known on configurations similar to those of actual devices (ITER, NSTX, ASDEX upgrade, and Field Reversed Configurations).

Keywords: Hybridizable Discontinuous Galerkin (HDG), Curved Boundary, Grad-Shafranov, Anderson Acceleration, Plasma Equilibrium, Magnetohydrodynamics (MHD)
2010 MSC: 65N30, 65Z05

1. Introduction

In an axisymmetric geometry with the standard cylindrical coordinates denoted by (r, ϕ, z) along the unit direction vectors $\hat{\mathbf{r}}$, $\hat{\boldsymbol{\phi}}$, and $\hat{\mathbf{z}}$, the magnetic field $\mathbf{B}$ can be written as

$$\mathbf{B} = \left(\frac{\widehat{\boldsymbol{\phi}}}{r}\right) \times \nabla\psi(r, z) + g(\psi)\left(\frac{\widehat{\boldsymbol{\phi}}}{r}\right),$$

where the function ψ is known as the *poloidal flux function* and $g(\psi)/r$ is the *toroidal field function* [1, 2]. The latter is also related to the net current flowing in the plasma and the external coils in the poloidal direction, I_p, through the relation

$$I_p = \frac{2\pi}{\mu_0} g(\psi).$$

In every cross section $\phi = constant$, the plasma will be confined to the region where the level sets of ψ are closed curves. Since the magnetic field depends on ψ only through its gradient, with the introduction of an appropriate shift, the boundary between the confinement and free regions can be made to correspond to the level set $\psi = 0$ and we will assume so without loss of generality.

Under axisymmetry requirements, it can be shown that the equilibrium condition

$$\nabla p = \mathbf{J} \times \mathbf{B},$$

Email addresses: tonatiuh@cims.nyu.edu (Tonatiuh Sánchez-Vizuet), msolano@ing-mat.udec.cl (Manuel Solano)

between the force due to the kinetic pressure p in the plasma and that one produced by the effect of the magnetic field $\mathbf{B}$ on the current density $\mathbf{J}$ can be expressed entirely in terms of ψ, p, and g. The resulting equivalent expression

$$-\Delta^*\psi = \mu_0 r^2 \frac{dp}{d\psi} + g\frac{dg}{d\psi} \tag{1}$$

was derived independently by Grad and Rubin, [3], Shafranov [4], and Lust and Schlüter [5] and is known as the *Grad-Shafranov* equation. In the above expression, the magnetic permeability of vacuum μ_0 is constant and the toroidal operator Δ^* is defined by

$$\Delta^*\psi := r^2\nabla\cdot\left(\frac{1}{r^2}\nabla\psi\right) = r\partial_r\left(\frac{1}{r}\partial_r\psi\right) + \partial_z^2\psi.$$

The function $g(\psi)$ and the pressure $p(\psi)$ depend purely on ψ and the specific functional form of the dependence can be considered to be user provided. In view of this, it is convenient to write (1) in the form

$$-\nabla\cdot\left(\frac{1}{r^2}\nabla\psi\right) = \frac{F(r,z,\psi)}{r^2}, \qquad F(r,z,\psi) := \mu_0 r^2\frac{dp}{d\psi} + g\frac{dg}{d\psi}, \tag{2}$$

which highlights the semi-linear nature of the equation and leads naturally to a weak formulation. If the right hand side is truly nonlinear as a function of ψ, the equation can be solved iteratively.

When the precise location of the plasma-vacuum region is known a priori, the problem of determining the flux function in the plasma region becomes an interior Dirichlet boundary-value problem. In this case, the $\psi = 0$ level set is of particular interest, for it becomes the boundary of the plasma domain, henceforth denoted Ω, where homogeneous Dirichlet boundary conditions are imposed. This problem is often referred to as a fixed boundary problem and its numerical solution will be the focus of the present work.

Depending on the location, number, and current intensity of the external coils, the level set $\psi = 0$ may become a separatrix, presenting what is known as an *x-point*, as depicted in Figure 1. This kind of non-smooth geometry often poses additional challenges for some numerical solvers but the presence of such a point is often a desirable engineering feature and fusion reactors are frequently designed to produce such a configuration. One of the reasons for this is that in the absence of the clear division between the vacuum and plasma regions provided by the separatrix, a physical *limiter* must be introduced to prevent the plasma from touching the walls of the reactor. Since the limiter is in direct contact with the plasma, this severely limits the range of temperatures attainable in experiments and may introduce additional impurities in the plasma. Moreover, as the particles drift along the magnetic flux lines, the region between the open ends of the separatrix provides the optimal placement location for the divertor, a device that removes plasma impurities, extracts excess heat and protects the walls of the reactor [6, 7]. Therefore, methods that can handle these kinds of geometries can prove to be advantageous.

Many different approaches for the solution of the Grad-Shafranov equation have been employed over the years a very detailed –if dated– description of the different approaches can be found in the review by Takeda and Tokuda [2], which is also a very complete reference on general aspects of plasma equilibrium. A concise discussion on plasma equilibrium and some related numerical techniques can also be found in [1]. In what follows we briefly discuss only some relevant recent attempts.

The weak form associated to (2) is well suited for numerical computations and has been exploited in many computational efforts, such as the widely used Finite Element codes CHEASE [8] and HELENA [9, 10] both of which use bi-cubic Hermite elements, and in more recent works involving the use of high-order spectral elements on rectangular geometries [11] or mimetic elements [12]. Nevertheless, weak/variational treatments are by no means the only way to approach the solution and many different alternatives have been employed.

The use of Multi-Grid methods can be traced back at least to the work of Braams [13] and have continued to attract attention over time both for the fixed boundary problem [14] and the free boundary problem [15]. In recent years, integral equation techniques combined with conformal mapping have been successfully employed to achieve fast and high order algorithms that can provide accurate approximations for both the flux function and its derivatives [16], this approach has been adopted in the flexible code ECOM [17].

Some other recent alternatives that have attracted attention include the hybrid approach EEC-ESC that couples Hermite elements near the plasma edge with Fourier decomposition methods in the plasma core [18], the use of

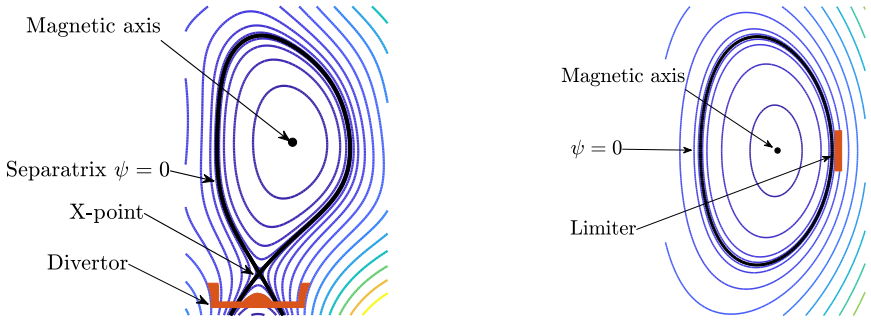

Figure 1: The plasma remains confined inside the region where the contour lines of the poloidal flux ψ are closed. For engineering purposes, reactors are often designed so that the last closed flux surface has an x-point, below which a divertor is placed (left). When the contour lines do not present a separatrix, a limiter is required in order to forcefully stop the plasma from getting in contact with the reactor walls (right). Fixed-boundary computations solve the interior boundary value problem in the confinement region delimited by the level set $\psi = 0$.

meshless methods [19, 20], the use of approximate particular solutions [21], and the method of fundamental solutions [22].

The references above do not focus on high order approximation of the derivatives of the flux function. However, the quantity of physical relevance in the problem is the magnetic field $\mathbf{B}$, therefore an important requirement for a Grad-Shafranov solver is to be able to provide accurate approximations to the partial derivatives of the flux as well, and some advanced post-processing techniques towards that goal have been developed lately [23].

The present work represents an intermediate stage in a wider effort that aims to build a robust and flexible solver, in the spirit of ECOM, capable of dealing with direct equilibria as well as q and $J_\parallel$ solves. The novelty, and strength, of our approach resides on the use of a high order Hybridizable Discontinuous Galerkin (HDG) method combined with a technique for handling geometries with curved boundaries that preserves the order of accuracy of the method. The result is a robust algorithm that is able to provide high order of accuracy for the approximation of both the flux function and the magnetic field, is naturally suited for parallel computations [24, 25], and provides the flexibility of handling curved geometries (with or without x-points) relying only on polygonal meshes [26].

This work will focus only on the algorithmic aspects of the method and the analytical parts will be dealt with in a separate communication. The rest of the paper is structured as follows: In Section 2 we present the mixed formulation of the Grad-Shafranov equation at the continuous level; Section 3 describes the solution algorithm starting with the handling of curved boundaries by extension from a polygonal mesh and the HDG formulation is described afterwards. The treatment of the non-linearity through an accelerated fixed-point iteration closes the description of the algorithm. We then move on to the validation of the method in Section 4 where the analytic solutions used as benchmarks are described and we present the results of the numerical experiments; concluding remarks and directions for future and ongoing work are given in the final Section 5.

2. The continuous mixed formulation

In order to apply an HDG discretization, the interior problem for the Grad-Shafranov equation must be first recast as a first order system. This is required since HDG methods include the gradient (or flux) as an additional unknown, but the reformulation is also physically motivated, since the quantity of interest is the magnetic field as opposed to the scalar poloidal flux.

Consider the fixed-boundary problem for the Grad-Shafranov equation

$$-\widetilde{\nabla}\cdot\left(\frac{1}{r}\widetilde{\nabla}\psi\right) = \frac{F}{r} \qquad \text{in } \Omega \subset \mathbb{R}^2, \tag{3a}$$

$$\psi = 0 \qquad \text{on } \partial\Omega, \tag{3b}$$

where $\widetilde{\nabla} := (\partial_r, \partial_z)$ and Ω is a cross section of the reactor at constant toroidal angle. The choice of an HDG discretization requires us to formulate the problem in mixed form through the introduction of the *flux* $\boldsymbol{q} := \frac{1}{r}\widetilde{\nabla}\psi$ as an additional

unknown. This addition transforms (3) into the equivalent system

$$\boldsymbol{q} - \frac{1}{r}\widetilde{\nabla}\psi = \boldsymbol{0} \qquad \text{in } \Omega \subset \mathbb{R}^2, \tag{4a}$$

$$-\widetilde{\nabla}\cdot\boldsymbol{q} = \frac{F}{r} \qquad \text{in } \Omega \subset \mathbb{R}^2, \tag{4b}$$

$$\psi = 0 \qquad \text{on } \Gamma := \partial\Omega. \tag{4c}$$

If we had a standard continuous Galerkin discretization in mind, at this point the above system would be transformed into a weak formulation by the usual process of testing with arbitrary functions from appropriately chosen spaces. However, the HDG method does not rely on a continuous weak formulation. Instead, the domain is tessellated first and then a piecewise polynomial approximation is built by solving local weak problems defined over a polygonal approximation of Ω. The strategy will be described in detail in Section 3.

3. The numerical method

The solution strategy that we propose consists of three main components: (a) An HDG solver for a linear elliptic operator on polygonal domains, (b) a "transfer" algorithm that allows for the handling of curved geometries through computation on polygonal sub-domains, and (c) an accelerated fixed point strategy that takes care of the semi-linearity of the problem (or eigenvalue problem).

Since the HDG discretization is tied to the actual tessellation of the domain used for the computations, we start with the construction of the simplified polygonal domain used to approximate the plasma region, we then describe the method used to transfer Dirichlet data between the physical (curved) boundary and the computational (polygonal) boundary, next we introduce the HDG method applied to (4) and we conclude the section describing the iterative strategy used to treat the non-linearity and the eigenvalue problem.

3.1. The treatment of curved boundaries

In general, standard numerical methods are defined over polygonal or polyhedral domains that can be triangulated. When dealing with domains having a curved boundary, *fitted* [27, 28] or *unfitted* [29, 30] methods can be considered. In the former case, the boundary Γ is matched or "fitted" by the computational boundary Γ_h. For instance, Γ_h can be constructed by interpolating Γ. On the other hand, unfitted methods approximate the domain by a polygonal domain whose boundary Γ_h does not necessarily "fit" Γ. For example, this can be achieved by immersing Ω in a background mesh and setting the computational domain to be the union of all the elements of the mesh that lie inside Ω. In both approaches the boundary condition on Γ_h must be properly defined.

The boundary data can be easily imposed in the computational domain using fitted methods, which is one of their main advantages. For a high order fitted method, Γ_h must approximate Γ with enough accuracy to be able to recover a high order approximation of the solution. For instance, isoparametric finite elements can be used [27], where the elements near the boundary have a curved side that locally interpolates Γ. This construction might be unpractical in complicated geometries or evolving domains and that is why unfitted methods are preferred in these cases since the computational mesh is not adjusted to Ω. However, the main drawback of standard unfitted methods is that only low order approximations can be obtained due the fact that the boundary data on the computational domain is imposed "away" from the true boundary.

Recently, an approach that combines the flexibility in the generation of the mesh characteristic of unfitted methods with a technique to transfer the boundary data from Γ to Γ_h has been developed [26, 31]. This method proposes an approximation of the boundary data by performing line integration along segments, called *transferring paths*, connecting Γ_h to Γ. This strategy preserves the order of accuracy used for the approximation and has been our choice for handling the curved boundary. We describe it in detail in what follows.

Computational domain. Let $\mathcal{B}$ be a background polygonal domain such that $\Omega \subset \mathcal{B}$ and $\mathcal{T}_h$ a triangulation of $\mathcal{B}$ consisting of triangles K that are uniformly shape-regular as Figure 2 (left) shows. For a triangle K, we denote its diameter by h_K and its outward unit normal by $\boldsymbol{n}_K$, writing $\boldsymbol{n}$ instead of $\boldsymbol{n}_K$ when there is no confusion. The mesh size h is defined as $\max_{K\in\mathsf{T}_h} h_K$ and we assume the triangulation does not have hanging nodes. Let T_h be the set of

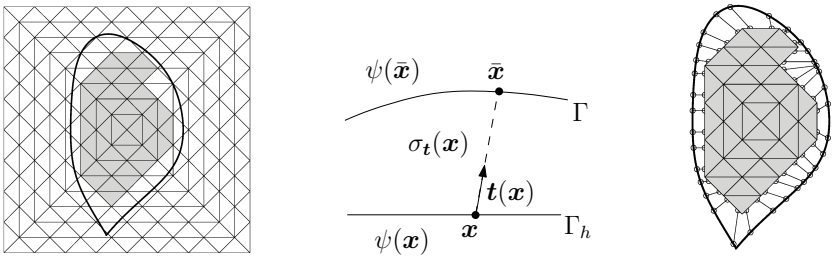

Figure 2: Left: Domain Ω, background domain (square) and polygonal subdomain (shaded). Middle: Transferring path $\sigma_{\boldsymbol{t}}$ connecting $\boldsymbol{x} \in \Gamma_h$ to $\bar{\boldsymbol{x}} \in \Gamma$. Right: Transferring paths (segments with starting and ending points marked with $\circ$) associated to two points on each boundary edge.

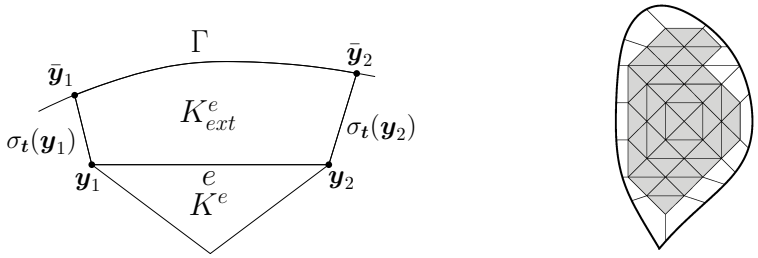

Figure 3: Left: A boundary edge e with corresponding element K^e and region K^e_{ext}. Right: $\Omega^h_{ext} := \cup_{e \in \mathcal{E}^\partial_h} K^e_{ext}$ (white region).

triangles of $\mathcal{T}_h$ that lie completely inside of Ω. Then, we define the computational domain $\Omega^h := (\bigcup_{K \in \mathsf{T}_h} \overline{K})^\circ$ (shaded region in Figure 2). The set of edges of T_h is denoted by $\mathcal{E}_h$ and $\mathcal{E}_h = \mathcal{E}^\circ_h \cup \mathcal{E}^\partial_h$, where $\mathcal{E}^\circ_h$ and $\mathcal{E}^\partial_h$ are the sets of interior and boundary edges, respectively. Finally, we denote by $\Gamma_h := \cup \mathcal{E}^\partial_h$ the boundary of the computational domain.

Transferring paths. Let $\boldsymbol{x} = (r, z)$ be a point of Γ_h to which we associate a point $\bar{\boldsymbol{x}} = (\bar{r}, \bar{z})$ in Γ. We denote by $\sigma_{\boldsymbol{t}}(\boldsymbol{x})$ the segment joining $\bar{\boldsymbol{x}}$ and $\boldsymbol{x}$ with unit tangent vector $\boldsymbol{t}(\boldsymbol{x})$ and length $l(\boldsymbol{x})$ as depicted in Figure 2 (middle). We will refer to $\sigma_{\boldsymbol{t}}(\boldsymbol{x})$ as the *transferring path* associated to $\boldsymbol{x}$.

In principle, $\bar{\boldsymbol{x}}$ can be *any* point in Γ as long its distance to $\boldsymbol{x}$ is of order h. Given $\boldsymbol{x} \in \Gamma_h$, we will use the algorithm proposed in [26] to find $\bar{\boldsymbol{x}} \in \Gamma$ such that the following three conditions are satisfied: (1) $\sigma_{\boldsymbol{t}}(\boldsymbol{x})$ does not intersect another transferring path before terminating at Γ, (2) it does not intersect the interior of the computational domain Ω^h and (3) the distance form $\bar{\boldsymbol{x}}$ to $\boldsymbol{x}$ is of order h. More details can be found in Section 2.4.1 of [26]. Figure 2 (right) shows the transferring paths associated to two quadrature points on each boundary edge. For the computations, only transferring paths associated to the quadrature points and vertices of a boundary edge are needed.

Extension from subdomains. Given a boundary edge e with vertices at $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$, we define K^e_{ext} as the interior of the region determined by e, the segments $\sigma_{\boldsymbol{t}}(\boldsymbol{y}_1)$ and $\sigma_{\boldsymbol{t}}(\boldsymbol{y}_2)$, and the arc of Γ connecting $\bar{\boldsymbol{y}}_2$ and $\bar{\boldsymbol{y}}_1$ as shown in Figure 3. By construction, K^e_{ext} does not intersect the corresponding region associated to a neighboring boundary edge. In addition, we observe that the union of all the elements K^e_{ext} coincides with the complementary region $\Omega \setminus \Omega^h$ as shown in Figure 3. Hence, we define $\Omega^h_{ext} := \cup_{e \in \mathcal{E}^\partial_h} K^e_{ext}$ and note that $\Omega = \Omega^h \cup \Omega^h_{ext}$.

Finally, it is convenient at this point to define a local polynomial extrapolation that will be used in order to build the approximation of the boundary data at Γ_h. More precisely, let e be the boundary edge that belongs to the triangle $K^e \in \mathsf{T}_h$ and is associated to the exterior region K^e_{ext}. For a polynomial function p defined on K^e, we denote by $E(p)$ the extrapolation from K^e to K^e_{ext} obtained by extending the domain of definition of p to $K^e \cup K^e_{ext}$ while keeping the same polynomial form. Hence,

$$E(p) : K^e \cup K^e_{ext} \longrightarrow \mathbb{R}, \qquad E(p)(\boldsymbol{y}) := p(\boldsymbol{y}). \tag{5}$$

This definition provides a systematic way to extend a polynomial function defined only in the computational domain into the region Ω^h_{ext} enclosed by the physical boundary Γ and the computational boundary Γ_h.

Approximation of the boundary data. Let $\sigma_{\boldsymbol{t}}(\boldsymbol{x})$ be the transferring path from $\boldsymbol{x} = (r, z) \in \Gamma_h$ to $\bar{\boldsymbol{x}} = (\bar{r}, \bar{z}) \in \Gamma$. Multiplying (4a) by r, integrating along $\sigma_{\boldsymbol{t}}(\boldsymbol{x})$ and recalling that $\psi(\bar{\boldsymbol{x}}) = 0$, we obtain

$$\psi(\boldsymbol{x}) = -\int_0^{l(\boldsymbol{x})} r\,\boldsymbol{q}(\boldsymbol{x} + \boldsymbol{t}(\boldsymbol{x})s) \cdot \boldsymbol{t}(\boldsymbol{x})\,ds \;=:\; \varphi(\boldsymbol{x}). \tag{6}$$

In other words, φ is an *exact* representation of the boundary data at Γ_h that depends on the unknown $\boldsymbol{q}$. Let us emphasize that the numerical method will approximate the solution $(\boldsymbol{q}, \psi)$ in Ω^h, however the integral in the definition of φ is over a segment lying outside Ω^h where $\boldsymbol{q}$ is still unknown. That is why, instead of considering the exact boundary condition φ, we will use an approximation denoted by φ_h. To be more precise, if $\boldsymbol{q}_h$ is the approximation of $\boldsymbol{q}$ computed by the HDG method on K^e, then motivated by (6) we define

$$\varphi_h(\boldsymbol{x}) := -\int_0^{l(\boldsymbol{x})} rE(\boldsymbol{q}_h)(\boldsymbol{x} + \boldsymbol{t}(\boldsymbol{x})s) \cdot \boldsymbol{t}(\boldsymbol{x})\,ds, \tag{7}$$

where $E(\boldsymbol{q}_h)$ is the extension of the polynomial $\boldsymbol{q}_h|_{K^e}$ to the neighboring exterior element K^e_{ext} defined in (5).

3.2. The Hybridizable Discontinuous Galerkin Method

The problem to solve in the computational domain is

$$\begin{aligned}
\boldsymbol{q} - \frac{1}{r}\widetilde{\nabla}\psi &= \boldsymbol{0} && \text{in } \Omega^h,\\
-\widetilde{\nabla}\cdot\boldsymbol{q} &= \frac{F}{r} && \text{in } \Omega^h,\\
\psi &= \varphi_h && \text{on } \Gamma_h,
\end{aligned}$$

where φ_h is given by (7). The above problem can be thought of as a set of local boundary value problems defined on every element $K \in \mathsf{T}_h$. The local solutions are coupled through (a) the introduction of the *hybrid* unknown $\widehat{\psi} := \psi|_{\mathcal{E}_h}$ that corresponds to the restriction of ψ to the skeleton of the mesh $\mathcal{E}_h$, and (b) the requirement that for every pair of elements K^+ and K^- sharing an edge e and with exterior normal vectors given by $\boldsymbol{n}^+$, $\boldsymbol{n}^-$, the normal component of the flux be continuous across their interface $[\![\boldsymbol{q}]\!] := \boldsymbol{q}^+ \cdot \boldsymbol{n}^+ + \boldsymbol{q}^- \cdot \boldsymbol{n}^- = 0$. This equivalent form can be stated as

$$\begin{aligned}
\boldsymbol{q} - \frac{1}{r}\widetilde{\nabla}\psi &= \boldsymbol{0} && \text{in } K\ \forall\, K \in \mathsf{T}_h,\\
-\widetilde{\nabla}\cdot\boldsymbol{q} &= \frac{F}{r} && \text{in } K\ \forall\, K \in \mathsf{T}_h,\\
\psi &= \widehat{\psi} && \text{on } \partial K\ \forall\, K \in \mathsf{T}_h,\\
[\![\boldsymbol{q}]\!] &= 0 && \text{on } e\ \forall e \in \mathcal{E}_h^\circ,\\
\psi &= \varphi_h && \text{on } \Gamma_h.
\end{aligned}$$

The first three equations above define the local problems, while the last two define the global system that determines the hybrid unknown $\widehat{\psi}$. The HDG method [24] seeks approximations $(\boldsymbol{q}_h, \psi_h, \widehat{\psi}_h)$ to the solutions $(\boldsymbol{q}, \psi, \psi|_{\mathcal{E}_h})$ in the finite dimensional space $\boldsymbol{V}_h \times W_h \times M_h$ given by

$$\begin{aligned}
\boldsymbol{V}_h &= \{\boldsymbol{v} \in \boldsymbol{L}^2(\mathsf{T}_h) :\ \boldsymbol{v}|_K \in \mathbf{P}_k(K)\ \ \forall K \in \mathsf{T}_h\},\\
W_h &= \{w \in L^2(\mathsf{T}_h) :\ \ w|_K \in \mathbb{P}_k(K)\ \ \forall K \in \mathsf{T}_h\},\\
M_h &= \{\mu \in L^2(\mathcal{E}_h) :\ \ \mu|_e \in \mathbb{P}_k(e)\ \ \forall e \in \mathcal{E}_h\},
\end{aligned}$$

where $\mathbb{P}_k(K)$ is the space of polynomials of degree k defined on the triangle K, $\mathbf{P}(K) := [\mathbb{P}_k(K)]^2$, and $\mathbb{P}_k(e)$ is the space of polynomials of degree k defined on the edge e. Following the standard notation, we will denote by $(\cdot,\cdot)_K$ and $\langle\cdot,\cdot\rangle_{\partial K}$ the L^2 inner products on an element K and on its boundary ∂K respectively, and we will define

$$(\cdot,\cdot)_{\mathsf{T}_h} := \sum_{K\in\mathsf{T}_h} (\cdot,\cdot)_K \qquad\qquad \langle\cdot,\cdot\rangle_{\partial\mathsf{T}_h} := \sum_{K\in\mathsf{T}_h} \langle\cdot,\cdot\rangle_{\partial K}.$$

The approximation $(\boldsymbol{q}_h, \psi_h, \widehat{\psi}_h) \in \boldsymbol{V}_h \times W_h \times M_h$ is the unique solution of

$$
\begin{aligned}
(r\boldsymbol{q}_h, \boldsymbol{v})_{\mathsf{T}_h} + (\psi_h, \widetilde{\nabla} \cdot \boldsymbol{v})_{\mathsf{T}_h} - \langle \widehat{\psi}_h, \boldsymbol{v} \cdot \boldsymbol{n} \rangle_{\partial \mathsf{T}_h} &= 0, && \text{(8a)}\\
(\boldsymbol{q}_h, \widetilde{\nabla} w)_{\mathsf{T}_h} - \langle \widehat{\boldsymbol{q}}_h \cdot \boldsymbol{n}, w \rangle_{\partial \mathsf{T}_h} &= (F/r, w)_{\mathsf{T}_h}, && \text{(8b)}\\
\langle \widehat{\psi}_h, \mu \rangle_{\Gamma_h} &= \langle \varphi_h, \mu \rangle_{\Gamma_h}, && \text{(8c)}\\
\langle \widehat{\boldsymbol{q}}_h \cdot \boldsymbol{n}, \mu \rangle_{\partial \mathsf{T}_h \setminus \Gamma_h} &= 0, && \text{(8d)}
\end{aligned}
$$

for all $(\boldsymbol{v}, w, \mu) \in \boldsymbol{V}_h \times W_h \times M_h$, where the *numerical flux* $\widehat{\boldsymbol{q}}_h$ is defined as

$$
\widehat{\boldsymbol{q}}_h \cdot \boldsymbol{n} := \boldsymbol{q}_h \cdot \boldsymbol{n} + \tau(\psi_h - \widehat{\psi}_h) \qquad \text{on } \partial \mathsf{T}_h, \tag{8e}
$$

and τ is a non-negative piecewise constant stabilization parameter defined on $\partial \mathsf{T}_h$. This simple choice for the numerical flux has become somewhat standard, but it is by no means unique. One of the advantages of the definition (8e) is that it keeps equations (8a) and (8b) completely in terms of local quantities (once the hybrid unknown has been determined). It has been shown (in [32] for polyhedral domains and in [31] for curved domains) that the method achieves optimal convergence order when τ is kept of order one. In our computations the value of the stabilization parameter was set to $\tau = 1$.

Once the solution $(\boldsymbol{q}_h, \psi_h)$ in Ω^h is computed by solving (8), it can be extrapolated to the entire domain Ω using (5). However, we can define a better approximation for ψ on the exterior domain by means of the transferring technique (6). More precisely, consider $\boldsymbol{y} \in K^e_{ext}$ and let $\sigma_t(\boldsymbol{y})$ be the transferring path from $\boldsymbol{y} = (r, z) \in \Gamma_h$ to $\bar{\boldsymbol{y}} = (\bar{r}, \bar{z}) \in \Gamma$. Then, recalling that $\psi(\bar{\boldsymbol{y}}) = 0$, the approximation of $\psi(\boldsymbol{y})$ is given by

$$
\psi_h(\boldsymbol{y}) := -\int_0^{l(\boldsymbol{y})} r\, \boldsymbol{q}_h(\boldsymbol{y} + \boldsymbol{t}(\boldsymbol{x})s) \cdot \boldsymbol{t}(\boldsymbol{y})\, ds. \tag{9}
$$

It is evident that HDG methods are related to the family of *mixed methods*, where the gradient of the scalar potential is introduced as an additional unknown of the problem. This reformulation results in an increased number of problem unknowns, but in return ensures that the order of accuracy provided by the discretization is the same for the original unknown and its partial derivatives.

As a matter of fact, the mixed structure of the problem, together with the hybridization, also has positive consequences from the computational point of view. For instance, the hybridization technique helps to overcome the increment on the number of degrees of freedom usually associated to discontinuous Galerkin methods and the resulting matrices have the same sparsity and size as the ones resulting from the hybrid *continuous* mixed methods [33]. This contrasts with the tendency of Discontinuous Galerkin methods to "use too many unknowns". Moreover, since the only global unknown is the numerical trace that lies on the edges of the triangulation, once it is determined, the other unknowns are locally computed in each element, allowing for parallelization.

The reader familiar with mixed methods may object that this family imposes very tough restrictions on the possible choices for the approximation spaces, but one can rest assured that yet another positive feature of HDG is providing with ample flexibility for the spaces used. We refer the reader interested in the programming aspects of HDG methods to [34], where a very detailed explanation and coding strategies and tools are given. Our own implementation is based on the tools provided in that reference.

3.3. Accelerated fixed-point iteration

In order to deal with the semi-linearity of the problem we will resort to an iterative strategy. Due to their simplicity and effectiveness, straightforward fixed-point iterations (also known as a Picard iterations) of the style

$$
\Delta^* \psi^n = F(r, z, \psi^{n-1})
$$

have been preferred in many applications [11, 12, 15, 16, 17, 18, 20]. We choose to follow a similar strategy, but enhance it with two simple yet effective acceleration methods.

The first method consists of a simple two-grid strategy that was already included in CHEASE [8] where the fixed-point iteration is carried out in a coarse grid $\tau_H(\Omega)$ until convergence is achieved to a prescribed tolerance. The

resulting coarse-grid solution is then projected onto a finer grid $\tau_h(\Omega)$ where it is used as initial guess for a second round of fixed-point iterations. The computations on the coarse grid are considerably less taxing and the resulting improved initial guess decreases the number of iterations required on the fine grid. The convergence properties of this two-grid fixed-point strategy have been analyzed in [35, 36], where rigorous conditions for optimal rates of convergence are discussed.

The second strategy pertains to the fixed-point iteration itself and is an optimized variation of the popular back-averaging method known as *Anderson acceleration* [37]. The idea is to use an optimized convex linear combination of a predetermined number of previous iterates as input for the next update. If we denote by $M(\cdot)$ the mapping whose fixed-point is being sought for and u_0 the initial input then, in its simplest form, the acceleration algorithm $\texttt{anderson}\big(m, u^0, M(\cdot), \epsilon\big)$ using m previous iterates can be described as follows:

Algorithm 1: $\texttt{anderson}\big(m, u^0, M(\cdot), \epsilon\big)$

Data:
m: Depth. u^0: Initial guess. M : Mapping. ϵ: Stopping tolerance.
Result:
u^*: Approximate fixed-point of M.
begin
  $n = 0$, $Res = 1$;
  $\widetilde{u}^1 = M(u^0)$;
  $G^1 = \widetilde{u}^1 - u^0$;
  $u^1 = \widetilde{u}^1$;
  while $Res \geq \epsilon$ **do**
    $n = n + 1$;
    $k = \min\{m, n\}$;
    $\widetilde{u}^{n+1} = M(u^n)$;
    $G^{n+1} = \widetilde{u}^{n+1} - u^n$;
    Find: $(\alpha_1, \ldots, \alpha_{k+1}) \in \mathbb{R}^{k+1}$ such that
      1. $\sum_{j=1}^{k+1} \alpha_j = 1$
      2. $(\alpha_1, \ldots, \alpha_{k+1}) = \operatorname{argmin} \| \sum_{j=1}^{k+1} \alpha_j G^{n+j-k} \|$
    $u^n = \sum_{j=1}^{k+1} \alpha_j \widetilde{u}^{n+j-k}$;
    $Res = \|u^n - u^{n-1}\| / \|u^n\|$;
  end
  $u^* = u^n$;
end

It is clear that the algorithm of depth $m = 0$ coincides with the simple Picard iterative scheme. By including information from more than one of the previous updates in this way, the convergence can be dramatically improved. Furthermore, in terms of the number of iterations needed to achieve a prescribed tolerance, the method can not do worse than Picard for linear problems and for non linear problems it often provides considerable improvement, as shown by Toth and Kelley [38]. In the same work, the authors report that, although no results are available regarding the optimal depth m, empirically there is no gain from choosing $m \geq 3$. This was consistent with our own experiments and therefore we settled for an acceleration of depth $m = 2$ in our implementation. A more elaborate version of the procedure allows for the mixing of previous instances of both $\tilde{u}$ and u [38, 39], but for our implementation we follow the simple procedure described above.

3.4. Summary of the solution method

Let $S_{h,F}$ be the operator that, for a given right hand side F, maps the initial guess ψ^0 to the HDG approximation $(\boldsymbol{q}_h, \psi_h)$ computed on a mesh with parameter h, and Π_h be he $L^2(\Omega)$ projector onto the space $V_h(\Omega)$. With this notation we can summarize the solution strategy algorithmically as follows:

Algorithm 2: `solveGS`$(F, \tau_H(\Omega), \tau_h(\Omega), \epsilon)$

Data:
F: Right hand side. $\tau_H(\Omega)$: Coarse triangulation. $\tau_h(\Omega)$: Fine triangulation.
ϵ: Stopping tolerance.
Result:
$(\boldsymbol{q}_h, \psi_h)$: Approximate HDG solutions.
begin
  ψ^0 ; `// Non-trivial initial guess`
  $(\boldsymbol{q}_H, \psi_H) =$ `anderson`$\left(2, \psi^0, S_{H,F}, \epsilon\right)$; `// Coarse grid`
  $\psi^0 = \Pi_h \psi_H$; `//` L^2 `Projection onto a fine grid`
  $(\boldsymbol{q}_h, \psi_h) =$ `anderson`$\left(2, \psi^0, S_{h,F}, \epsilon\right)$; `// Fine grid`
end

4. Numerical Experiments

The accuracy and convergence properties of the scheme are evaluated by comparing the approximations to the flux function and its gradient to some existing analytical solutions briefly presented below. The geometries used for the simulations are closely related to physically relevant configurations: the International Thermonuclear Experimental Reactor (ITER), the National Spherical Toroidal Experiment (NSTX), the Axially Symmetric Divertor Experiment (ASDEX upgrade), and Field Reversed Configurations (FRC). We measure error in the standard L^2 norm, but also verify the performance "off the grid" in an L^∞-related norm by sampling on random non-grid points in the computational domain and considering the maximum discrepancy as the error measure. Convergence as a function of the grid size and the polynomial degree are tested independently and the results of the numerical experiments on the different geometries are presented at the end of the section.

4.1. Analytical solutions in free space

For particular pressure and poloidal current profiles some exact solutions have been derived. These profiles determine the right hand side of equation (3) that we recall here for convenience:

$$-\widetilde{\nabla} \cdot \left(\frac{1}{r}\widetilde{\nabla}\psi\right) = \frac{F(r,z,\psi)}{r}, \qquad F(r,z,\psi) := \mu_0 r^2 \frac{dp}{d\psi} + g\frac{dg}{d\psi}.$$

The concrete expressions of the solutions we use as benchmarks (and the right hand sides they give rise to) are described briefly in what follows.

Solov'ev profiles. These profiles for $p(\psi)$ and $g(\psi)$ arise when the terms on the right hand side of (1) are such that

$$\mu_0 \frac{dp}{d\psi} = A\,, \qquad g\frac{dg}{d\psi} = C,$$

for some constant values A and C. If the flux is normalized so that $A + C = 1$ the right hand side function becomes

$$F(r,z,\psi) = -\Big((1-A)r^2 + A\Big). \tag{10}$$

The solution of this equation can be split into a homogeneous part, ψ_H, and particular part ψ_P, so that $\psi = \psi_H + \psi_P$. A particular solution is

$$\psi_P(r,z) = \frac{r^4}{8} + A\left(\frac{1}{2}r^2 \ln r - \frac{r^4}{8}\right), \tag{11}$$

while the homogeneous solution can be a linear combination the terms:

$$
\begin{aligned}
\psi_1 &= 1, & \psi_7 &= 8z^6 - 140z^4r^2 + 75z^2r^4 - 15r^6 \ln r \\
\psi_2 &= r^2, & &\quad + 180r^4z^2 \ln r - 120r^2z^4 \ln r, \\
\psi_3 &= z^2 - r^2 \ln r, & \psi_8 &= z, \\
\psi_4 &= r^4 - 4r^2z^2, & \psi_9 &= zr^2, \\
\psi_5 &= 2z^4 - 9z^2r^2 + 3r^4 \ln r & \psi_{10} &= z^3 - 3zr^2 \ln r, \\
&\quad - 12r^2z^2 \ln r, & \psi_{11} &= 3zr^4 - 4z^3r^2, \\
\psi_6 &= r^6 - 12r^4z^2 + 8r^2z^4, & \psi_{12} &= 8z^5 - 45zr^4 - 80z^3r^2 \ln r + 60zr^4 \ln r.
\end{aligned} \tag{12}
$$

Following the procedure carefully derived in [40] to choose the coefficients $(c_1, \ldots, c_{12})$ corresponding to each of these functions, the solution ψ can be written in terms of three free parameters $(\epsilon, \delta, \kappa)$ that describe the cross section of the configuration. By adjusting the values of ϵ, δ, and κ it is possible to obtain profiles that correspond to relevant physical configurations. The interested reader is referred to the above reference for more details on the parametrization of the plasma boundary.

Configurations with dissimilar source functions. If, as proposed in [41], the pressure and toroidal flux are set to

$$
p = \frac{S}{\mu_0}\psi\,, \qquad g^2 = T\psi^2 + 2U\psi + g_0^2
$$

for some constants S, T, U, and g_0, the right hand side function of (1) becomes

$$
F(r, z, \psi) = T\psi + S\,r^2 + U. \tag{13}
$$

The solution to the Grad-Shafranov equation for these sources can be obtained by a similar procedure as before; by splitting the solution into homogeneous and particular parts $\psi = \psi_h + \psi_p$ where

$$
\psi_h = -\frac{1}{T}\left(U + S\,r^2\right) \quad \text{and} \quad -\Delta^*\psi_p = T\psi_p.
$$

Eight different families of solutions ψ_p to the above eigenvalue problem are found in [41]. Here we will focus on one particular linear combination of them that gives rise to the solution

$$
\begin{aligned}
\psi = {} & c_1 + c_2r^2 + rJ_1(pr)\,(c_3 + c_4z) + c_5\cos(pz) + c_6\sin(pz) \\
& + r^2\,(c_7\cos(pz) + c_8\sin(pz)) + c_9\cos\left(p\sqrt{r^2+z^2}\right) + c_{10}\sin\left(p\sqrt{r^2+z^2}\right) \\
& + rJ_1(\nu r)\,(c_{11}\cos(qz) + c_{12}\sin(qz)) + rJ_1(qr)\,(c_{13}\cos(\nu z) + c_{14}\sin(\nu z)) \\
& + rY_1(\nu r)\,(c_{15}\cos(qz) + c_{16}\sin(qz)) + rY_1(qr)\,(c_{17}\cos(\nu z) + c_{18}\sin(\nu z))\,,
\end{aligned} \tag{14}
$$

where $p = \sqrt{T}$, $q = p/2$, $\nu = \sqrt{3/4}p$, and J_1 (resp. Y_1) is the Bessel function of the first (resp. second) kind. The first two terms of the above expression correspond to the homogeneous solution ψ_h where we have set $U = -c_1T$ and $S = -c_2T$.

4.2. Convergence studies

Refinement strategies. In the numerical experiments both mesh and polynomial refinement (commonly referred to as h and p refinements respectively) tests are considered. The computational grids are built following the steps given in Section 3.1. Starting from a background shape-regular triangulation with diameter h, the associated computational domain Ω^h and the region Ω^h_{ext} are determined for every geometry. Mesh refinement is performed by uniform subdivision of the background mesh, which results in a finer mesh with diameter $h/2$; the computational domain and the extension are updated accordingly. For the cases requiring two-grid iterations, the coarse and fine meshes correspond to adjacent levels of refinements with diameters $h/2^n$ and $h/2^{n+1}$. In the case of p-refinements, the grids mentioned

above were held fixed and the refinement was carried out by increasing the polynomial degree of the approximation space.

On the convergence plots, the polynomial degree is varied along the horizontal axis, while the convergence curves for meshes with different diameters is superimposed with a different color on the same graph. The reader is referred to the electronic version of the paper for the color reference.

Error measures. We consider two different error measures for both the poloidal flux ψ and its gradient $\nabla\psi$. First, for a function f and its HDG approximation f_h we define the usual mean square error as

$$E_2(f) := \left(\|f - f_h\|^2_{L^2(\Omega^h)} + \|f - f_h\|^2_{L^2(\Omega^h_{ext})} \right)^{1/2}.$$

This expression includes the error contributions from the computational domain Ω_h and from the corresponding exterior region Ω^h_{ext}– where the value of $\boldsymbol{q}_h$ is defined by the extrapolation in (5) and the value of ψ_h by the transference defined in (9).

In many applications, the output of a plasma equilibrium code is not the final result of a simulation, but only a part of a much bigger calculation involving different physical processes simulated by several computational codes coupled together. In these cases the approximation of the magnetic field obtained is sampled and used as input for the computation of other physical processes of interest. The points where the approximation is evaluated may or may not coincide with nodes of the original mesh and therefore it is important to have a measure of the point-wise convergence of the approximation. As an estimate of the point-wise accuracy of the algorithm we sample the solution at random points in the interior of Ω and consider the maximum discrepancy with the analytic solution. This leads to the definition

$$E_\infty(f) := \max_{\boldsymbol{x}\in S} |f(\boldsymbol{x}) - f_h(\boldsymbol{x})|,$$

where S is comprised of five random points from every element of the discretization of both Ω^h and Ω^h_{ext}.

Stopping criteria. In all the following examples the stopping criteria for the iterative algorithm was set to a relative difference of 10^{-12} or below between two consecutive iterations. This effectively sets the maximum possible accuracy of the algorithm.

Solov'ev profiles in smooth geometries. We start by testing the performance of the method on a smooth geometry corresponding to a Field Reversed Configuration device modeled using the parametrization given in [40]. The parametrization of the plasma-vacuum boundary uses parameter values $\epsilon = 0.99, \delta = 0.7, \kappa = 10$, and $A = 0$ models an FRC, the coarsest mesh for this geometry is characterized by $h = 1.25$.

For this configuration, Example 1, the exact Solov'ev solution is a linear combination of the functions ψ_p from equation (11) with $A = 0$ and ψ_1, ψ_2, and ψ_4 from equation (12), yielding an up-down symmetric configurations which is a bivariate polynomial of fourth degree. This explains the sharp drop of the L^2 approximation error when the polynomial degree reaches 4 (Figure 4).

Despite the smoothness of the geometry and the relative simplicity of the exact solution, configurations like this can present challenges to solvers due two factors. The first one, and perhaps the most evident, is the proximity of the plasma boundary to the origin, where the toroidal operator Δ^* becomes singular. The second one is the high the elongation of the confinement region which can lead to overcrowding in the regions of high curvature for methods relying on conformal mapping, or to meshes with elements that are either too large and resolve the geometry poorly, or too many and result in large numbers of unknowns.

As it is shown in Figure 4, the algorithm performs remarkably well despite the geometrical challenges mentioned above and the comparatively large mesh diameters. For smooth geometries and Solov'ev profiles this kind of performance was also observed on other geometries such as a Spheromak and ITER-like and NSTX-like configurations without an x-point; for the sake of conciseness these examples were left out of the paper.

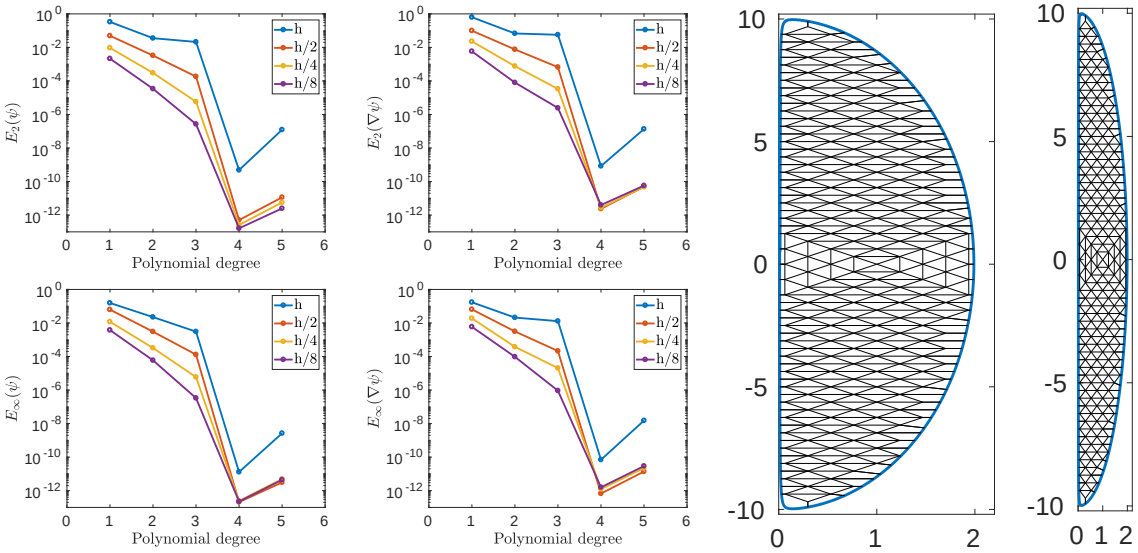

Figure 4: Convergence plots for Example 1 (FRC) for successive refinements of the computational grid and increasingly higher polynomial degrees. Left column: E_2 (top) and E_∞ (bottom) errors for the poloidal flux ψ. Center column: E_2 (top) and E_∞ (bottom) errors for $\nabla\psi$. Right column: Confinement region and sample grid corresponding the the second level of refinement (red curve in the convergence plot). The geometry is shown on 1:1 scale on the right end to stress the elongation of the geometry. The reader is referred to the on-line version of the manuscript for the color scheme.

Solov'ev profiles on geometries with an x-point. The second set of test problems consists of Solov'ev profiles in geometries that present an x-point. The geometries are up-down asymmetric with a downwards oriented x-point and are modeled by the parametrization detailed in [40]. The exact solutions for these configurations involve all the terms in equations (11) and (12) and are therefore more challenging tests not only from the geometrical point of view.

Example 2 corresponds to an ITER-like configuration with parameters $\epsilon = 0.32, \delta = 0.33, \kappa = 2$, and $A = -0.115$ the mesh on the computational domain has an initial diameter of $h = 0.175$ that is successively halved. Example 3 corresponds to an NSTX-like configuration modeled using the values $\epsilon = 0.78, \delta = 0.335, \kappa = 1.7$, and $A = -0.115$. In this case the coarsest computational mesh has diameter $h = 0.5$.

The presence of an x-point in these configurations makes the problem challenging for many available solvers, nevertheless as can be seen in Figure 5 for the ITER geometry and Figure 6 for the NSTX configuration, our algorithm performs satisfactorily both on and off the grid, with the NSTX geometry presenting a bigger challenge due to its larger elongation and proximity to the origin as compared to the ITER configuration.

Dissimilar source terms on geometries with an x-point. The final test chosen for Example 5 is based on the model for the ASDEX upgrade experiment discussed in [41] and described by equation (14). With the pressure and current profiles determined by this model, the source term of the Grad-Shafranov equation depends linearly on ψ. This kind of source terms can be absorbed directly into a modified bilinear form and dealt with computationally by introducing a mass matrix in the system. However, our goal is to provide a solver that relies only on the discretization of the toroidal operator Δ^* and allows the user to provide the source terms as problem data. In this framework, a general source term is always dealt with iteratively.

The exact solution associated to this configuration of the ASDEX Upgrade experiment corresponds to equation (14) with the set of coefficients given by:

$$
\begin{aligned}
c_1 &= 0.17795 & c_6 &= -0.162 & c_{11} &= 1.5820 & c_{16} &= -0.4265 \\
c_2 &= -0.03291 & c_7 &= 0.3722 & c_{12} &= -0.009059 & c_{17} &= 0.8057 \\
c_3 &= 1.4934 & c_8 &= 0.07697 & c_{13} &= 2.2388 & c_{18} &= -0.004804 \\
c_4 &= -0.4818 & c_9 &= 1.2959 & c_{14} &= 0.4186 & T &= 17.8116. \\
c_5 &= -1.1759 & c_{10} &= 0.5881 & c_{15} &= 1.195 &
\end{aligned}
$$

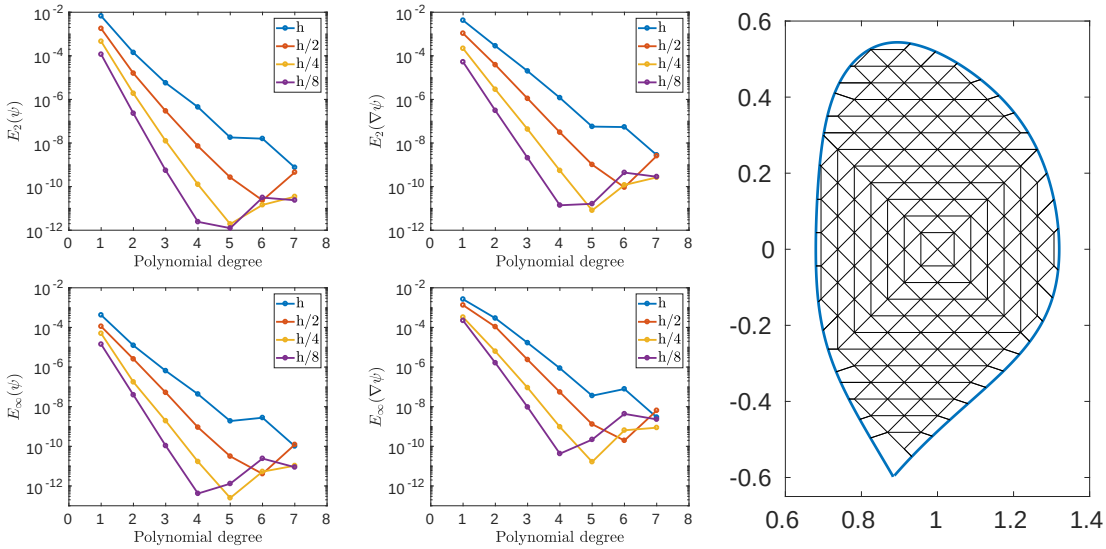

Figure 5: Convergence plots for Example 3 (ITER) for successive refinements of the computational grid and increasingly higher polynomial degrees. Leftt column: E_2 (top) and E_∞ (bottom) errors for the poloidal flux ψ. Center column: E_2 (top) and E_∞ (bottom) errors for $\nabla\psi$. Right column: Confinement region and sample grid corresponding the the second level of refinement (red curve in the convergence plot). The reader is referred to the on-line version of the manuscript for the color scheme.

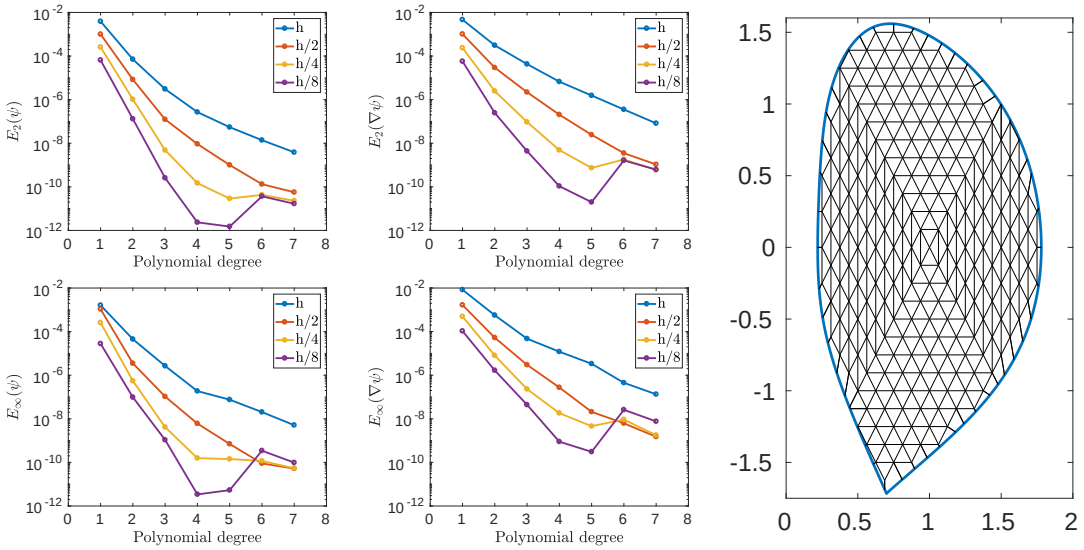

Figure 6: Convergence plots for Example 4 (NSTX) for successive refinements of the computational grid and increasingly higher polynomial degrees. Left column: E_2 (top) and E_∞ (bottom) errors for the poloidal flux ψ. Center column: E_2 (top) and E_∞ (bottom) errors for $\nabla\psi$. Right column: Confinement region and sample grid corresponding the the second level of refinement (red curve in the convergence plot). The reader is referred to the on-line version of the manuscript for the color scheme.

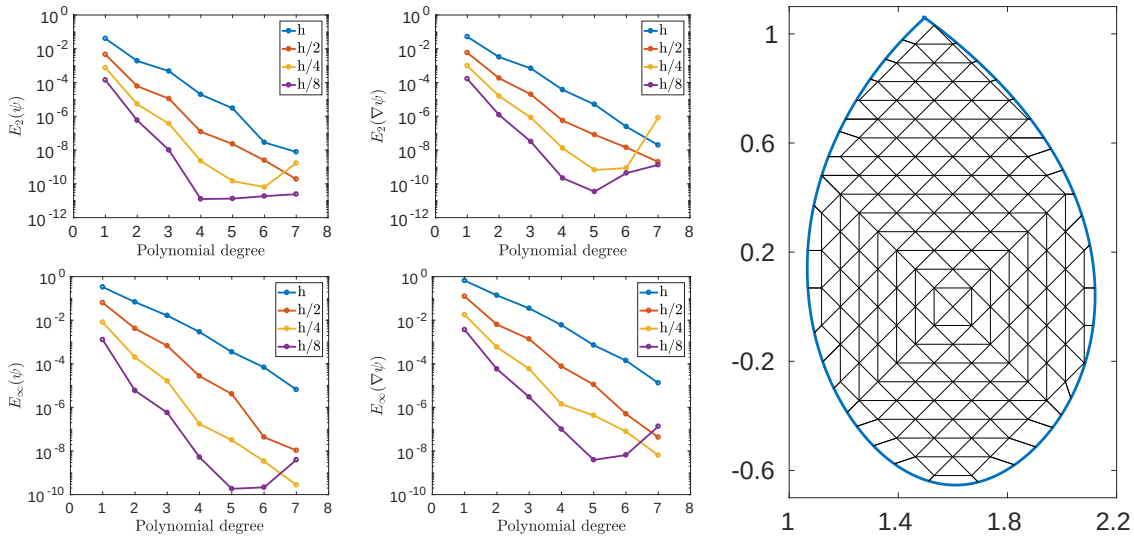

Figure 7: Convergence plots for Example 5 (ASDEX upgrade with an upwards oriented x-point) for successive refinements of the computational grid and increasingly higher polynomial degrees. Left column: E_2 (top) and E_∞ (bottom) errors for the poloidal flux ψ. Center column: E_2 (top) and E_∞ (bottom) errors for $\nabla\psi$. Right column: Confinement region and sample grid corresponding the the second level of refinement (red curve in the convergence plot). The reader is referred to the on-line version of the manuscript for the color scheme.

These values give rise to a configuration with an upwards-oriented x-point as shown in Figure 7. The background mesh at the coarsest level of refinement had diameter $h = 0.275$. As confirmed by the convergence plots on Figure 7, the iterative process performs well both in the point-wise and mean square senses.

5. Discussion

We have presented a high order accurate solver for the Grad-Shafranov equation based on the Hybridizable Discontinuous Galerkin method. As the numerical examples show, the method is very robust with respect to the geometrical properties of the confinement region and provides competitive convergence rates for the poloidal flux function and, more importantly, for the magnetic field even for relatively coarse grids.

The use of HDG provides an ideal framework for parallel computation, due to the fact that, once the global hybrid unknown has been determined, the local problems decouple and can be solved independently from each other. This feature enhances the speed of the calculations specially in fine grids with a large number of elements. The transferring path method used to enforce Dirichlet boundary conditions on a polygonal subdomain gives the algorithm great geometric flexibility and avoids the need for constant re-meshing if the geometry has to be adjusted, as it is often the case for real time monitoring or free boundary applications. Moreover, the technique can be applied to both fitted and unfitted meshes.

On the downside, the extrapolation strategy used to define the approximation in the extension Ω^h_{ext} may fail to resolve possible boundary layers close to the separatrix. In order to address this possibility, the authors are currently working on an adaptive mesh refinement strategy that when combined with the transferring technique can resolve the fine-scale structure up to a prescribed tolerance without increasing the computational cost excessively.

Other areas of improvement include the implementation of a combined two-grid fixed-point + Newton step strategy, where accelerated fixed point iterations are carried out on the coarse grid to generate a good initial guess which is used as the starting point for a Newton iteration on the fine grid. This technique promises to dramatically decrease the number of iterations required on the fine grid, speeding up the computation even further. All these enhancements are the subject of ongoing work and will be implemented in a subsequent stage in order to enable the treatment of the free boundary problem and $q-$solves.

6. Acknowledgements

The authors are deeply thankful to Antoine Cerfon for his numerous suggestions and remarks that contributed to greatly improve the quality of the work and to point out several possible paths for future and current improvements. Tonatiuh Sánchez-Vizuet was partially supported by the US Department of Energy. Grant No. DE-FG02-86ER53233. Manuel Solano was partially supported by CONICYT–Chile through FONDECYT project No. 1160320, by BASAL project CMM, Universidad de Chile and by Centro de Investigación en Ingeniería Matemática (CI^{2}MA), Universidad de Concepción.

[1] S. Jardin, Computational methods in plasma physics, Chapman & Hall/CRC Computational Science Series, CRC Press, Boca Raton, FL, 2010. doi:10.1201/EBK1439810958.
URL http://dx.doi.org/10.1201/EBK1439810958

[2] T. Takeda, S. Tokuda, Computation of MHD equilibrium of tokamak plasma, Journal of Computational Physics 93 (1) (1991) 1 – 107. doi:https://doi.org/10.1016/0021-9991(91)90074-U.
URL http://www.sciencedirect.com/science/article/pii/002199919190074U

[3] H. Grad, H. Rubin, Hydromagnetic equilibria and force-free fields, in: Proc. Second international conference on the peaceful uses of atomic energy, Geneva, Vol. 31,190, United Nations, New York, 1958.

[4] V. D. Shafranov, On magnetohydrodynamical equilibrium configurations, Soviet Physics JETP 6 (1958) 545–554.

[5] R. Lüst, A. Schlüter, Axialsymmetrische magnetohydrodynamische Gleichgewichtskonfigurationen, Z. Naturf 12a (1957) 850–854.

[6] G. Janeschitz, K. Borrass, G. Federici, Y. Igitkhanov, A. Kukushkin, H. D. Pacher, G. W. Pacher, M. Sugihara, The ITER divertor concept, Journal of Nuclear Materials 220 (1995) 73–88. doi:10.1016/0022-3115(94)00447-1.

[7] K. Dietz, S. Chiocchio, A. Antipenkov, G. Federici, G. Janeschitz, E. Martin, R. Parker, R. Tivey, Engineering and design aspects related to the development of the iter divertor, Fusion Engineering and Design 27 (Supplement C) (1995) 96 – 108, proceedings of the Third International Symposium on Fusion Nuclear Technology. doi:https://doi.org/10.1016/0920-3796(95)90121-3.
URL http://www.sciencedirect.com/science/article/pii/0920379695901213

[8] H. Lütjens, A. Bondeson, A. Roy, Axisymmetric MHD equilibrium solver with bicubic Hermite elements, Computer Physics Communications 69 (2) (1992) 287 – 298. doi:https://doi.org/10.1016/0010-4655(92)90167-W.
URL http://www.sciencedirect.com/science/article/pii/001046559290167W

[9] G. Huysmans, J. Goedbloed, W. Kerner, Isoparametric bicubic Hermite elements for solution of the Grad-Shafranov equation, International Journal of Modern Physics C 02 (01) (1991) 371–376. arXiv:http://www.worldscientific.com/doi/pdf/10.1142/S0129183191000512, doi:10.1142/S0129183191000512.
URL http://www.worldscientific.com/doi/abs/10.1142/S0129183191000512

[10] C. Konz, R. Zille, Manual of HELENA Fixed Boundary Equilibrium Solver, Max-Planck Institute for Plasma Physics, Garching, Germany (2007).

[11] E. Howell, C. Sovinec, Solving the Grad-Shafranov equation with spectral elements, Computer Physics Communications 185 (5) (2014) 1415 – 1421. doi:https://doi.org/10.1016/j.cpc.2014.02.008.
URL http://www.sciencedirect.com/science/article/pii/S001046551400040X

[12] A. Palha, B. Koren, F. Felici, A mimetic spectral element solver for the Grad-Shafranov equation, Journal of Computational Physics 316 (Supplement C) (2016) 63 – 93. doi:https://doi.org/10.1016/j.jcp.2016.04.002.
URL http://www.sciencedirect.com/science/article/pii/S0021999116300341

[13] B. J. Braams, Magnetohydrodynamic equilibrium calculations using multigrid, in: W. Hackbusch, U. Trottenberg (Eds.), Multigrid Methods II: Proceedings of the 2nd European Conference on Multigrid Methods held at Cologne, October 1–4, 1985, Springer, Berlin, Heidelberg, 1986, pp. 38–51. doi:10.1007/BFb0072640.
URL https://doi.org/10.1007/BFb0072640

[14] T. Hanawa, S. Ikuno, A. Kamitani, Application of parallelized multigrid method to solution of MHD equilibrium with MPI, IEEE Transactions on Magnetics 40 (2) (2004) 1005–1008. doi:10.1109/tmag.2004.825440.
URL https://doi.org/10.1109/tmag.2004.825440

[15] P.-A. Gourdain, J.-N. Leboeuf, R. Neches, High-resolution magnetohydrodynamic equilibrium code for unity beta plasmas, Journal of Computational Physics 216 (1) (2006) 275 – 299. doi:https://doi.org/10.1016/j.jcp.2005.12.005.
URL http://www.sciencedirect.com/science/article/pii/S0021999105005528

[16] A. Pataki, A. J. Cerfon, J. P. Freidberg, L. Greengard, M. ONeil, A fast, high-order solver for the Grad-Shafranov equation, Journal of Computational Physics 243 (Supplement C) (2013) 28 – 45. doi:https://doi.org/10.1016/j.jcp.2013.02.045.
URL http://www.sciencedirect.com/science/article/pii/S0021999113001721

[17] J. Lee, A. Cerfon, ECOM: A fast and accurate solver for toroidal axisymmetric {MHD} equilibria, Computer Physics Communications 190 (2015) 72 – 88. doi:https://doi.org/10.1016/j.cpc.2015.01.015.
URL http://www.sciencedirect.com/science/article/pii/S0010465515000351

[18] X. Li, L. E. Zakharov, V. V. Drozdov, Edge equilibrium code for tokamaks, Physics of Plasmas 21 (1) (2014) 012505. doi:10.1063/1.4861369.
URL https://doi.org/10.1063/1.4861369

[19] R. Imazawa, Y. Kawano, K. Itami, Meshless method for solving fixed boundary problem of plasma equilibrium, Journal of Computational Physics 292 (Supplement C) (2015) 208 – 214. doi:https://doi.org/10.1016/j.jcp.2015.03.035.
URL http://www.sciencedirect.com/science/article/pii/S0021999115001904

[20] M. Ghasemi, R. Amrollahi, Numerical solution of ideal MHD equilibrium via radial basis functions collocation and moving least squares approximation methods, Engineering Analysis with Boundary Elements 67 (Supplement C) (2016) 126 – 137. doi:https://doi.org/

10.1016/j.enganabound.2016.02.008.
URL http://www.sciencedirect.com/science/article/pii/S0955799716300248
[21] D. Nath, M. Kalra, P. Munshi, Computation of fixed boundary tokamak equilibria using a method based on approximate particular solutions, Computers & Mathematics with Applications 70 (5) (2015) 1220 – 1233. doi:https://doi.org/10.1016/j.camwa.2015.07.013.
URL http://www.sciencedirect.com/science/article/pii/S089812211500348X
[22] D. Nath, M. S. Kalra, Solution of Grad-Shafranov equation by the method of fundamental solutions, Journal of Plasma Physics 80 (3) (2014) 477494. doi:10.1017/S0022377814000026.
[23] L. Ricketson, A. Cerfon, M. Rachh, J. Freidberg, Accurate derivative evaluation for any gradshafranov solver, Journal of Computational Physics 305 (2016) 744 – 757. doi:https://doi.org/10.1016/j.jcp.2015.11.015.
URL http://www.sciencedirect.com/science/article/pii/S002199911500755X
[24] B. Cockburn, The hybridizable discontinuous Galerkin methods, in: Proceedings of the International Congress of Mathematicians. Volume IV, Hindustan Book Agency, New Delhi, 2010, pp. 2749–2775.
[25] R. M. Kirby, S. J. Sherwin, B. Cockburn, To CG or to HDG: a comparative study, J. Sci. Comput. 51 (1) (2012) 183–212. doi:10.1007/s10915-011-9501-7.
URL http://dx.doi.org/10.1007/s10915-011-9501-7
[26] B. Cockburn, M. Solano, Solving Dirichlet boundary-value problems on curved domains by extensions from subdomains, SIAM J. Sci. Comput. 34 (1) (2012) A497–A519. doi:10.1137/100805200.
URL http://dx.doi.org/10.1137/100805200
[27] M. Lenoir, Optimal isoparametric finite elements and error estimates for domains involving curved boundaries, SIAM Journal on Numerical Analysis 23 (3) (1986) 562–580.
URL http://www.jstor.org/stable/2157524
[28] J. H. Bramble, J. T. King, A finite element method for interface problems in domains with smooth boundaries and interfaces, Advances in Computational Mathematics 6 (1) (1996) 109–138. doi:10.1007/BF02127700.
URL https://doi.org/10.1007/BF02127700
[29] C. Peskin, Flow patterns around heart valves: A numerical method, Journal of Computational Physics 10 (2) (1972) 252–271. doi:10.1016/0021-9991(72)90065-4.
[30] J. H. Bramble, T. Dupont, V. Thomée, Projection methods for Dirichlet's problem in approximating polygonal domains with boundary-value corrections, Mathematics of Computation 26 (120) (1972) 869–879.
URL http://www.jstor.org/stable/2005869
[31] B. Cockburn, W. Qiu, M. Solano, A priori error analysis for hdg methods using extensions from subdomains to achieve boundary conformity, Math. Comp. 83 (2014) 665–699. doi:10.1090/S0025-5718-2013-02747-0.
URL https://doi.org/10.1090/S0025-5718-2013-02747-0
[32] B. Cockburn, J. Gopalakrishnan, F.-J. Sayas, A projection-based error analysis of HDG methods, Math. Comp. 79 (271) (2010) 1351–1367. doi:10.1090/S0025-5718-10-02334-3.
URL http://dx.doi.org/10.1090/S0025-5718-10-02334-3
[33] B. Cockburn, J. Gopalakrishnan, A characterization of hybridized mixed methods for second order elliptic problems, SIAM Journal on Numerical Analysis 42 (1) (2004) 283–301. arXiv:https://doi.org/10.1137/S0036142902417893, doi:10.1137/S0036142902417893.
URL https://doi.org/10.1137/S0036142902417893
[34] Z. Fu, L. F. Gatica, F.-J. Sayas, Algorithm 949: MATLAB tools for HDG in three dimensions, ACM Trans. Math. Software 41 (3) (2015) Art. 20, 21. doi:10.1145/2658992.
URL http://dx.doi.org/10.1145/2658992
[35] J. Xu, A novel two-grid method for semilinear elliptic equations, SIAM J. Sci. Comput. 15 (1) (1994) 231–237. doi:10.1137/0915016.
URL http://dx.doi.org/10.1137/0915016
[36] J. Xu, Two-grid discretization techniques for linear and nonlinear PDEs, SIAM J. Numer. Anal. 33 (5) (1996) 1759–1777. doi:10.1137/S0036142992232949.
URL http://dx.doi.org/10.1137/S0036142992232949
[37] D. G. Anderson, Iterative procedures for nonlinear integral equations, J. Assoc. Comput. Mach. 12 (1965) 547–560. doi:10.1145/321296.321305.
URL http://dx.doi.org/10.1145/321296.321305
[38] A. Toth, C. T. Kelley, Convergence analysis for Anderson acceleration, SIAM J. Numer. Anal. 53 (2) (2015) 805–819. doi:10.1137/130919398.
URL http://dx.doi.org/10.1137/130919398
[39] H. F. Walker, P. Ni, Anderson acceleration for fixed-point iterations, SIAM J. Numer. Anal. 49 (4) (2011) 1715–1735. doi:10.1137/10078356X.
URL http://dx.doi.org/10.1137/10078356X
[40] A. J. Cerfon, J. P. Freidberg, one size fits all analytic solutions to the gradshafranov equation, Physics of Plasmas 17 (3) (2010) 032502. doi:10.1063/1.3328818.
URL http://dx.doi.org/10.1063/1.3328818
[41] P. J. M. Carthy, Analytical solutions to the grad-Shafranov equation for tokamak equilibrium with dissimilar source functions, Physics of Plasmas 6 (9) (1999) 3554–3560. doi:10.1063/1.873630.
URL http://dx.doi.org/10.1063/1.873630

Centro de Investigación en Ingeniería Matemática (CI^{2}MA)

PRE-PUBLICACIONES 2017

2017-19 Raimund Bürger, Ilja Kröker: *Computational uncertainty quantification for some strongly degenerate parabolic convection-diffusion equations*

2017-20 Gabriel N. Gatica, Bryan Gomez-Vargas, Ricardo Ruiz-Baier: *Mixed-primal finite element methods for stress-assisted diffusion problems*

2017-21 Sergio Caucao, Gabriel N. Gatica, Ricardo Oyarzúa: *Analysis of an augmented fully-mixed formulation for the non-isothermal Oldroyd-Stokes problem*

2017-22 Rolando Biscay, Joaquin Fernández, Carlos M. Mora: *Numerical solution of stochastic master equations using stochastic interacting wave functions*

2017-23 Gabriel N. Gatica, Mauricio Munar, Filander A. Sequeira: *A mixed virtual element method for the Navier-Stokes equations*

2017-24 Nicolas Barnafi, Gabriel N. Gatica, Daniel E. Hurtado: *Primal and mixed finite element methods for deformable image registration problems*

2017-25 Sergio Caucao, Gabriel N. Gatica, Ricardo Oyarzúa: *A posteriori error analysis of an augmented fully-mixed formulation for the non-isothermal Oldroyd-Stokes problem*

2017-26 Raimund Bürger, Stefan Diehl, María Carmen Martí: *A conservation law with multiply discontinuous flux modelling a flotation column*

2017-27 Antonio Baeza, Raimund Bürger, Pep Mulet, David Zorío: *Central WENO schemes through a global average weight*

2017-28 Rodolfo Araya, Manuel Solano, Patrick Vega: *Analysis of an adaptive HDG method for the Brinkman problem*

2017-29 Sergio Caucao, Marco Discacciati, Gabriel N. Gatica, Ricardo Oyarzúa: *A conforming mixed finite element method for the Navier-Stokes/Darcy-Forchheimer coupled problem*

2017-30 Tonatiuh Sanchez-Vizuet, Manuel Solano: *A Hybridizable Discontinuous Galerkin solver for the Grad-Shafranov equation*

Para obtener copias de las Pre-Publicaciones, escribir o llamar a: Director, Centro de Investigación en Ingeniería Matemática, Universidad de Concepción, Casilla 160-C, Concepción, Chile, Tel.: 41-2661324, o bien, visitar la página web del centro: `http://www.ci2ma.udec.cl`







Centro de Investigación en
Ingeniería Matemática (CI²MA)
Universidad de Concepción

Casilla 160-C, Concepción, Chile
Tel.: 56-41-2661324/2661554/2661316
http://www.ci2ma.udec.cl