PetIGA-MF: a multi-field high-performance toolbox for structure-preserving B-splines spaces

A.F. Sarmiento\textsuperscript{a,*}, A.M.A. C\'ortes\textsuperscript{b}, D.A. Garcia\textsuperscript{c}, L. Dalcin\textsuperscript{a,d}, N. Collier\textsuperscript{e}, V.M. Calo\textsuperscript{f,g}

\textsuperscript{a} Extreme Computing Research Center (ECRC)  
King Abdullah University of Science and Technology (KAUST)  
Thuwal, Saudi Arabia

\textsuperscript{b} Mathematics and Statistics Department  
Federal University of the State of Rio de Janeiro  
Rio de Janeiro, Brazil

\textsuperscript{c} Basque Center for Applied Mathematics (BCAM)  
Bilbao, Spain

\textsuperscript{d} Centro de Investigaci\'on de M\'etodos Computacionales (CIMEC)  
Consejo Nacional de Investigaciones Cient\'ificas y T\'ecnicas (CONICET)  
Universidad Nacional del Litoral (UNL)  
Santa Fe, Argentina

\textsuperscript{e} Computer Science and Mathematics Division  
Oak Ridge National Laboratory  
Oak Ridge, TN, USA

\textsuperscript{f} Applied Geology  
Western Australian School of Mines, Faculty of Science and Engineering, Curtin University  
Perth, WA, Australia 6845

\textsuperscript{g} Mineral Resources  
Commonwealth Scientific and Industrial Research Organization (CSIRO)  
Kensington, WA, Australia 6152

Abstract

We describe a high-performance solution framework for isogeometric discrete differential forms based on B-splines: PetIGA-MF. Built on top of PetIGA, an open-source library we have built and developed over the last decade, PetIGA-MF is a general multi-field discretization tool. To test the capabilities of our implementation, we solve different viscous flow problems such as Darcy, Stokes, Brinkman, and Navier-Stokes equations. Several convergence benchmarks based on manufactured solutions are presented assuring optimal convergence rates of the approximations, showing the accuracy and robustness of our solver.

Keywords: isogeometric analysis, discrete differential forms, structure-preserving discrete spaces, multi-field discretizations, PetIGA, high-performance computing

1. Introduction

The theory of finite element exterior calculus and the underlying concept of discrete differential forms surveyed in \cite{1} formalize the design of compatible discrete schemes. By compatibility we mean that the discretization preserves the mathematical structure of the partial differential equation and the functional spaces underlying them, from the continuous to the discrete setting. One example where such compatibility property is a necessary requirement for
the stability of the discrete scheme is the Maxwell equations system. In many cases, such compatibility condition is encoded on the commutativity of the de Rham diagram \([2,3]\).

Isogeometric Analysis (IGA)\([4]\) allows the definition of a family of discrete differential forms, based on splines functions, called isogeometric discrete differential forms. The isogeometric discrete differential forms theory, described in \([5]\), provides structure-preserving discrete spaces, specifically, gradient-, curl-, divergence- and integral-conforming spaces, which satisfy a discrete de Rham diagram. Curl-conforming spaces were first applied to approximate and solve the Stokes system in \([7]\). In a series of papers \([8–10]\) Evans and Hughes further developed the theory and the application of these spaces to approximate different incompressible viscous flow problems such as Darcy, Stokes, Brinkman and Navier-Stokes equations. In this case, the compatibility of the divergence- and integral-conforming B-spline spaces, when used as a discrete velocity-pressure pair, engenders two important properties of the scheme: the inf–sup stability and a point-wise divergence-free discrete velocity field.

Using these ideas, we build a high-performance solver called PetIGA-MF, as an extension of PetIGA \([11]\), a high-performance isogeometric discretization framework that simplifies modelling and simulation of problems using isogeometric analysis \([12,13]\). PetIGA-MF focuses on multiphysics and multi-field analyses using gradient-conforming spaces as well as curl-, divergence- and integral-conforming discretizations \([14,15]\).

The paper is organized as follows. In Section 2, we present the strong and weak forms of the generalized Navier-Stokes problem. Section 3 introduces B-spline basis functions, B-spline compatible spaces, and boundary condition imposition. In Section 4, we describe the implementation of our framework. In Section 5, we show the numerical results for all the test cases. We draw conclusions in Section 6.

2. Generalized Navier-Stokes problem

We start introducing the generalized Navier-Stokes problem to simplify the description of the incompressible flow problems we address in this paper, which are Darcy, Brinkman, Stokes and Navier-Stokes problems. The Darcy equation models viscous flows through porous media, whereas Brinkman equation models flow through porous media with an effective viscosity representing high permeability contrasts, for example, when large cavities are present in the medium. The Stokes equation models highly viscous flows, while the Navier-Stokes equation models flows where the advection is not negligible compared to the diffusivity. These generalizations result in a coupled nonlinear system of partial differential equations for the conservation of linear momentum and mass.

Assuming a steady state system in a bounded open domain \(\Omega \subset \mathbb{R}^d\) \((d = 2, 3)\), the problem in its strong form is to find \(U = [u, p]\), with \(u : \Omega \rightarrow \mathbb{R}^d\), and \(p : \Omega \rightarrow \mathbb{R}\) such that:

\[
\begin{align*}
\alpha \nabla \cdot (u \otimes u) + \beta u - \nabla \cdot \sigma(u, p) &= f & \text{in } \Omega \\
\nabla \cdot u &= 0 & \text{in } \Omega \\
\mathbf{u} &= \mathbf{g} & \text{on } \partial \Omega, 
\end{align*}
\]

where \(u\) is the fluid velocity field, \(p\) is the fluid pressure field, \(\sigma(u, p) = -\mu I + 2\nu \nabla \mathbf{u}\) is the Cauchy stress tensor for an incompressible fluid, with \(I\) being the identity matrix, and \(\nabla \mathbf{u}\) the symmetric part of the velocity gradient (strain rate), \(\nu\) is the kinematic viscosity, \(\beta\) is the reaction rate, \(f\) is the body force, and \(g\) is the Dirichlet boundary condition for the velocity. The remaining coefficient, \(\alpha\), is used to incorporate advection into the models. For \(\alpha = 0\) the flow field has no advection, like in the case of Stokes, Darcy, and Brinkman, whereas for \(\alpha = 1\) the flow has advection, as in the Navier-Stokes equations. The different models are recovered by varying the coefficients \(\alpha, \beta, \nu\). Having \(\alpha = 0\) and \(\beta \gg \nu\) represents the Darcy equations, \(\alpha = 0\) and \(\beta \approx \nu\) the Brinkman equations, \(\alpha = 0\) and \(\beta \ll \nu\) the Stokes equations, and \(\alpha = 1\) and \(\beta \ll \nu\) the Navier-Stokes equations.

Let \((\cdot, \cdot)_{\Omega}\) denote the \(L_2\) inner product in \(\Omega\). The trial and weighting spaces for the velocity field are defined by \(\mathcal{V}_s = \{v \in H^1(\Omega) : v = 0 \text{ on } \partial \Omega\}\), respectively, where \(u \in \mathcal{V}_s\) is a lift of a function in \(\mathcal{V}_0\), that is, \(u = v \circ I\). The trial and weighting spaces for pressure are \(Q = L_2(\Omega)\). With these notations, the weak form of the problem is to find \((u, p)\) such that:\n
\[
(W, \mathcal{L}(U)) = B_1(W, U) + B_2(W, U, U) = L(W) 
\]

2
where

\[ B_1(W, U) = (\nabla^2 w, 2\nabla^2 u)_\Omega + (w, \beta u)_{\Omega} - (\nabla \cdot w, p)_{\Omega} + (q, \nabla \cdot u)_{\Omega} \]
\[ B_2(W, U, U) = - (\nabla w, \alpha(u \otimes u))_{\Omega} \]
\[ L(U) = (w, f)_{\Omega} \]

here the bilinear operator \( B_1(\cdot, \cdot) \) represents the diffusive and reactive terms of the problem, the trilinear operator \( B_2(\cdot, \cdot, \cdot) \) represents the advective term, and the linear operator \( L(\cdot) \) represents the forcing term.

3. Discretization

We discretize the weak form of the problem (2) with compatible B-spline spaces, using a divergence-conforming space for the velocity and an integral-conforming space for the pressure. To simplify the description of such discrete approximation spaces, we give a brief introduction to B-spline functions, and then describe the compatible B-spline space for the velocity and an integral-conforming space for the pressure.

3.1. B-splines basis functions

B-spline basis functions are defined by the tensor product of univariate ones as

\[ B_{i_1, i_2, i_3}^{p_1, p_2, p_3} := B_{i_1, i_2}^{p_1} \otimes B_{i_2, i_3}^{p_2} \otimes B_{i_3}^{p_3} , \quad i_1 = 1, \ldots, n_1; \quad i_2 = 1, \ldots, n_2; \quad i_3 = 1, \ldots, n_3. \]

Defining the regularity vectors \( \varsigma_1, \varsigma_2, \varsigma_3 \) in each direction, the trivariate B-spline space is defined by

\[ \mathcal{S}^{p}_{\varsigma_1, \varsigma_2, \varsigma_3} := \text{span} \left[ B_{i_1, i_2, i_3}^{p_1, p_2, p_3} \right]_{i_1, i_2, i_3 = 1}^{n_1, n_2, n_3}. \]

We assume that the regularity vectors \( \varsigma_i \) are constant, with components equal to \( \varsigma \) (except \( \varsigma_1 = \varsigma_m = 0 \)), unless stated otherwise.
3.2. Isogeometric (B-spline) differential forms

The discrete differential forms concept in the context of the finite element method, also known as finite element exterior calculus, is surveyed in [1]. The key aspect of the theory is the use of algebraic topology tools, realized by the existence of de Rham diagrams (exact sequences) relating functional spaces and the image and the kernel of a differential operator between them. These relations are known to hold on the continuous setting, but to inherit such relations at the discrete level is a challenging accomplishment since it requires the definition of interpolation and projection operators to transfer the commutativity of the de Rham diagrams from the continuous to the discrete setting.

Based on the isogeometric analysis discretization framework Buffa et al. first introduced the isogeometric differential forms in the context of Maxwell equations [5] and Stokes equations [7], and later developed the general theory in [5]. At the same time, Evans and Hughes [8,10] applied this framework to the Generalized Stokes and Navier-Stokes equations. Therein, the isogeometric differential forms, based on B-splines, generate an exact sequence of discrete gradient-, curl-, divergence-, and integral-conforming spaces, that together with the proper interpolation and projection operators, defined in [5], render a commutative de Rham diagram. For the construction of de Rham commuting diagram in the context of $hp$ finite elements see [3]. The novelty of using the isogeometric framework is the possibility of an exact description of the geometry [4].

We use divergence- and integral-conforming spaces for the velocity and pressure, respectively, to solve the generalized Navier-Stokes problem. These spaces are defined in the parametric domain as follows:

<table>
<thead>
<tr>
<th>Divergence-conforming</th>
<th>Integral-conforming</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D: ( S_{p_1+1,p_2+1} \times S_{p_1+1,p_2+1} )</td>
<td>( S_{p_1+1,p_2} )</td>
</tr>
<tr>
<td>3D: ( S_{p_1+1,p_2+p_3} \times S_{p_1+1,p_2+p_3} \times S_{p_1+1,p_2+1,p_3} )</td>
<td>( S_{p_1+p_2+p_3} )</td>
</tr>
</tbody>
</table>

The main consequence of those definitions is that one can prove that the divergence operator is surjective for each pair of spaces above. In order to move the definitions from the parametric to the physical domain, a preserving push-forward mapping is used for every space in the sequence, guaranteeing that the de Rham diagram for the spaces defined on the physical domain (see [5]) also commutes in the discrete setting, that is, the spaces mapped to the physical domain also define an exact sequence. In our case, the relevant preserving mappings for our spaces of interest are:

\[
\begin{align*}
&\iota_u(v) = \det (DF) (DF)^{-1} (v \circ F), & v \in H(\text{div}; \Omega), \\
&\iota_p(q) = \det (DF) (q \circ F), & q \in L^2(\Omega),
\end{align*}
\]

where \( F \) is the geometric mapping from the parametric domain \( \Omega \) onto the physical domain \( \Omega \) (Figure 2), and \( DF \) is the gradient of the geometric mapping. The divergence-preserving map, \( \iota_u \), is the Piola transformation [20], and \( \iota_p \) is the integral preserving transformation. As one can infer the polynomial order and continuity of our basis functions in the physical domain, depend not only on the basis functions adopted in the parametric space but also on the geometric mapping used.

Additionally, the commutativity of the de Rham diagram with respect to the discretization projections means the stability of the scheme is guaranteed when the discrete velocity and pressure pair satisfy the inf–sup condition. Furthermore, the satisfaction of the weak incompressibility condition implies it holds strongly, that is, \((\nabla \cdot u = 0)\) the discrete divergence is zero pointwise (for a proof see [3]).

3.3. Boundary Condition Imposition

We impose the normal boundary conditions on the velocity strongly but doing the same for the tangential boundary conditions on the velocity with divergence-conforming basis functions may lead to unstable discretizations in domains with corners. Thus, we use Nitsche’s method for weak boundary imposition to avoid this problem, alleviating the necessity for highly refined meshes to reproduce the layer effect on the no-slip boundary conditions [21][22]. The
weak imposition of the tangential boundary conditions modifies the operators $B_1(W, U)$ and $L(W)$, introducing the adjoint consistency and the penalization terms, we then get

\[
\begin{align*}
\tilde{B}_1(W, U) &= B_1(W, U) \\
&\quad - (w, 2\nu \nabla^\prime u \cdot n)_T \quad \text{Consistency} \\
&\quad - (u, 2\nu \nabla^\prime w \cdot n)_T \quad \text{Adjoint consistency} \\
&\quad + (w, 2\nu \nabla^\prime u \alpha_p g)_T \quad \text{Penalization},
\end{align*}
\]

\[
\begin{align*}
\tilde{L}(W) &= L(W) \\
&\quad - (g, 2\nu \nabla^\prime w \cdot n)_T \quad \text{Adjoint consistency} \\
&\quad + (w, \nu \alpha_p g)_T \quad \text{Penalization}.
\end{align*}
\]

where $\alpha_p = C_{pen}/h_f$, and $C_{pen} = 5(p + 1)$ is the penalty parameter which depends on the polynomial order $p$ of the discretization, and $h_f$ is the wall normal mesh size [21].

4. Implementation

In this section, we describe an extension of PetIGA, which adds a flexible and scalable parallel implementation of multi-field isogeometric discretizations, where any field can be discretized as a conforming space of the discrete de Rham sequence (i.e. gradient-, curl-, divergence- and integral-conforming spaces). We first introduce the basic structures of PetIGA, and then present PetIGA-MF and the new structures that implement multi-fields discretizations in a user-friendly manner.

4.1. PetIGA

PetIGA [11] is a framework based on PETSc [23], which uses its parallel tools to solve a discrete variational formulation (Galerkin or collocation method) of partial differential equations. The discretization is built using B-spline functions and a patch-wise isoparametric mapping. Different structures built in PetIGA contain all the information the user needs to code the discrete variational formulation at a quadrature/collocation point. For a structured mesh and its partitioning, PetIGA implements its data structures, in a similar fashion to a DM in the PETSc jargon. In this case, the IGA (a modified DM) is tailored to the specifics of isogeometric analysis, particularly the use of high continuous basis functions, with possibly arbitrary continuity orders across elements boundaries, and the respective connectivity array to promote the assembly of the global matrices from their local contributions. With respect to the synergy between geometry description and finite element analysis, this data-structure called IGA provides the abstraction of a spline patch, together with the elemental and quadrature information needed to integrate a variational form when we use Galerkin’s method or collocation schemes [24].
The mesh is split, to balance the workload between the processors, according to a calculation of a box stencil, distributing the elements through the grid of processors, and then assigning the degrees of freedom that lie on the interfaces to one of the neighboring processors. When having an uneven distribution of elements, PetIGA is programmed to assign the higher workload to the next processor in the grid, to the left or the bottom, depending on the interface. Figure 3 shows an example of a spline space \( S_{p_1,p_2}^{\varsigma_1,\varsigma_2} \) defined over a mesh of \( 4 \times 4 \) elements \( p_1 = p_2 = 2 \) and \( \varsigma_1 = \varsigma_2 = 1 \) basis, and its splitting through a grid of \( 2 \times 2 \) processors.

![Figure 3: Distribution of elements for a mesh of 4×4 elements on a grid of 2×2 processors, for a spline space with \( p_1 = p_2 = 2 \) and \( \varsigma_1 = \varsigma_2 = 1 \) regularity. Grey-filled nodes represent the basis functions with support on the dashed element.](image)

As shown in Figure 3 and also reproduced in Figure 4a, the basis functions are naturally ordered in a lexicographic way, called natural numbering in PETSc jargon. Once in parallel such numbering is not convenient anymore, and the mesh splitting among processors induces a new numbering where the degrees of freedom that belong to the same processor are numbered first (see Figure 4b). This ordering is referred to as global numbering. Global vectors (see Figure 5a) are associated with this numbering. For processors to be able to solve in parallel, the information of the “ghost degrees of freedom” must communicate from neighboring processors. For such task a local numbering is more convenient as Figure 4c shows. Local vectors are associated with this numbering as shown in Figure 5b where lighter colors represent the ghost degrees of freedom. The amount of communication between processors depends on the continuity of the basis. All processor communications are hidden from the user and managed internally by PetIGA and its data-structures.
4.2. PetIGA-MF

PetIGA-MF is a multi-field extension of PetIGA, where different discretization spaces can be used for each field, making it suitable for solving multi-physics problems. All scalar and vector structure-preserving B-spline discrete spaces mentioned in section 3 are available. To simplify the access to the information of the different fields, we create new structures on top of the ones already existent in PetIGA, combining the single field data-structures to work in a multi-field framework.

PETSc provides a data management subclass, called DMComposite, which allows one to pack several fields in a monolithic block for multi-field and multi-physics discretizations. A new IGAM class packs the IGAs for each field together with an instance of a DMComposite. Once we create an IGAM object, the discrete spaces are set by assigning a type of structure-preserving space (gradient-conforming is the default type), and the corresponding fields of it. Figures 4 and 6 gives a schematic representation for the case of a two-dimensional divergence- and integral-conforming velocity-pressure pair of B-spline spaces. Two constraints built into PetIGA-MF are that all the fields, that are B-spline spaces, need to be defined on a mesh with the same number of elements and to use the same number of quadrature points per element. Figure 7 illustrates the natural numbering of the three fields for the divergence- and integral-conforming pair of spaces shown in Figure 6 defined on a mesh of $4 \times 4$ elements (see Figure 7a) with $p_1 = p_2 = 1$ and $\varsigma_1 = \varsigma_2 = 0$. Figures 7b, 7c and 7d emphasize the basis functions with support on the dashed element in Figure 7a for every field, these have a direct impact on the parallel partitioning of the degrees of freedom of each field.
\[ \text{field}[0] \left( S^{p+1}_{p+1, \alpha+1} \right) \]
\[ \text{field}[1] \left( S^p_{p, \alpha+1} \right) \]
\[ \text{field}[2] \left( S^p_{p, \alpha} \right) \]

Figure 6: Discrete velocity and pressure spaces abstraction used in PetIGA-MF.

\[ (a) \text{4x4 mesh used to define all the fields} \]

\[ (b) \text{Natural numbering for the space } S^{1}_{1,0} (\text{field}[0]). \]
\[ (c) \text{Natural numbering for the space } S^{2}_{0,1} (\text{field}[1]). \]
\[ (d) \text{Natural numbering for the space } S^{1}_{0,0} (\text{field}[2]). \]

Figure 7: Natural numbering for the degrees of freedom of all fields and basis functions numbers with support on the dashed element.
In PetIGA-MF every processor owns the part of every field that corresponds to its part of the mesh, and with respect to the global numbering the vector for a multi-field problem is schematically represented as in Figure 8a. To solve the fields in parallel, a first step is to create an independent vector for each field. We create these vectors by splitting the global vector into fields, obtaining the split global vectors as Figure 8b shows. The second step is to follow the same procedure as in PetIGA, for every split global vector, we obtain the split local vectors, that incorporate the ghost degrees of freedom as shown in Figure 8c.

4.2.1. Mapped basis functions.

We adhered to PetIGA’s philosophy that the framework delivers to the user the basis functions and their derivatives mapped to the physical space, called the shape functions. In this way, the user can directly code the variational formulation. Since in the multi-field setting we mix scalar and vector discrete spaces, we create a three indexed array of pointers, \( \text{shape}[d][i][j] \), to store the shape functions and their derivatives evaluated at the quadrature points of an element. The index \( d = 0, 1, 2 \) selects: the shape function, \( d = 0 \), its first derivative, \( d = 1 \), and its second derivative, \( d = 2 \). The indices \( i \) and \( j \) stand for the field components. Such an indexing is needed because of the use of mapped vector basis functions, for example, the divergence-conforming space on the physical domain.

Given the nature of the mappings used for the discrete vector spaces, for example the Piola transformation in the case of the divergence-conforming spaces, a component of a vector basis function in the parametric space, is mapped into a linear combination of all the parametric components, coupling them all in the physical space. Indeed, consider the example of the divergence-conforming space depicted in Figure 6 and let \( (\tilde{N}_1^u, \tilde{N}_2^u, \ldots, \tilde{N}_n^u) \) and \( (\tilde{N}_1^v, \tilde{N}_2^v, \ldots, \tilde{N}_n^v) \) represent the basis functions with support on an element for the spaces \( S_{p_1+1,p_2}^1 \times S_{p_1+1,p_2+1}^1 (\text{field}[0]) \) and \( S_{p_1+1,p_2+1}^1 (\text{field}[1]) \) respectively. The vector basis functions of the parametric space \( S_{p_1+1,p_2}^1 \times S_{p_1+1,p_2+1}^1 \) with support on the same element will be

\[
\left\{ (\tilde{N}_1^u, 0), (\tilde{N}_2^u, 0), \ldots, (\tilde{N}_n^u, 0), (0, \tilde{N}_1^v), (0, \tilde{N}_2^v), \ldots, (0, \tilde{N}_n^v) \right\}.
\]

Applying the push-forward transformation \( \iota^{-1}_u (u) = \text{det} (DF)^{-1} (DF) (u) \) to the set of parametric basis functions, we obtain the mapped basis function.
showing optimal convergence rates for both parametric and physical domains. Results for different Reynolds numbers

5. Numerical Results

Here we present the solution of the two-dimensional flow in a unitary square shown in [7]. We compute $u$ and $p$, when a body force $f$ is imposed, and compare the numerical solution with the prescribed manufactured solution $\bar{u}$ and $\bar{p}$ (Figure 9).

\[
\bar{u} := \begin{bmatrix}
2e^t(-1+x)x^2(y^2-y)(-1+2y) \\
(-e^t(-1+x)x(-2+x(3+x))(-1+y)^2y^2)
\end{bmatrix},
\]

\[
\bar{p} := -424 + 156e^t + (y^2-y)(-456 + e^t(456(1-x) + 228x^2 - 72x^4 + 12x^3 + 2(2x - 5x^2 + 2x^3 + x^3)(y^2-y))).
\]
The body force \( f \) is then defined for each test case by introducing the corresponding coefficients for \( \alpha \), \( \beta \) and \( \nu \), and the prescribed expressions for \( \bar{u} \) and \( \bar{p} \) into the momentum equation (1) as follows:

\[
f := \alpha \nabla \cdot (\bar{u} \otimes \bar{u}) + \beta \bar{u} - \nabla \cdot (2\nu \nabla \bar{u} - \bar{p} I). \tag{6}
\]

By computing the \( L^2 \) norm of the error we verify the convergence rates of the method against the theoretical estimates. We solve for nested meshes from 16\times16 to 512\times512 elements, doubling the number of elements at each step in each direction, using the undistorted and distorted meshes seen in Figure 10 to prove convergence in the parametric and physical domains. The distorted mesh used for the convergence tests is created by moving the control points of a mesh with one element, polynomial order \( p = 2 \) and continuity order \( \varsigma = 1 \), a distance \( d \) as shown in Figure 10(b), and then performing an \( h \)-refinement of the element. Results for three different polynomial orders with maximum continuity are shown in Figures 11 and 12, where the solid lines represent the results of the uniform meshes, and the dashed lines show the results of the distorted meshes. The asymptotic convergence rate \( r \) is given for every mesh and discretization.

Table 2: Asymptotic convergence rates for pressure in the unitary square case.

<table>
<thead>
<tr>
<th>Model</th>
<th>( p=1, \varsigma=0 )</th>
<th>( p=2, \varsigma=1 )</th>
<th>( p=3, \varsigma=2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( d=0 )</td>
<td>( d=0.45 )</td>
<td>( d=0 )</td>
</tr>
<tr>
<td>Stokes</td>
<td>1.99</td>
<td>1.47</td>
<td>3.02</td>
</tr>
<tr>
<td>Brinkman</td>
<td>1.99</td>
<td>1.47</td>
<td>3.02</td>
</tr>
<tr>
<td>Darcy</td>
<td>1.97</td>
<td>1.49</td>
<td>3.49</td>
</tr>
<tr>
<td>Navier-Stokes ( Re = 1 )</td>
<td>1.99</td>
<td>1.47</td>
<td>3.02</td>
</tr>
<tr>
<td>Navier-Stokes ( Re = 1000 )</td>
<td>2.00</td>
<td>2.00</td>
<td>3.00</td>
</tr>
</tbody>
</table>

Table 2 shows convergence rates for the error in the pressure. These convergence rates are equal to \( p + 1 \) when using uniform meshes, whereas they deteriorate to \( p \) when using a distorted mesh, with the exception of the Navier-Stokes and high Reynolds number case. This deterioration is more notorious in the cases with high polynomial orders. The loss of convergence for the error in the pressure when using distorted meshes corroborates the a priori error estimates presented in [8,9]. These results show that the theory is rigorous and that there is no superconvergence in the pressure. A table for convergence rates for the error in the velocity is not included, since these are consistently equal to \( p + 1 \) for all equations, and are not affected by the mesh distortion. Figures 11 and 12 show the convergence of velocity and pressure in the cases of Stokes and Navier-Stokes with a Reynolds number of one thousand, respectively.
similar convergence results were found for the other equations.

Figure 10: Meshes of 16×16 elements used to discretize the square physical domain. The undistorted mesh is used to test the convergence in the parametric domain $\hat{\Omega}$, and the distorted mesh to test the convergence in the physical domain $\Omega$.

Figure 11: Convergence test results for Stokes in the square problem.
5.1.1. Reduced quadrature schemes

We evaluate the convergence rates using two reduced quadrature schemes, one in which we keep the exact quadrature of $p+2$ points for the elements at the boundaries, and gradually reduce the number of quadrature points by one at contiguous elements, as they approach the center of the domain, until they reach a given minimum number of quadrature points per direction as shown in Figure 13(b). We also consider a homogeneous reduction of quadrature points. Both reduction schemes using $p+1$ quadrature points in every direction produce the same convergence rates as the exact quadrature, and no deterioration on the convergence constant. When both schemes reduce the number of quadrature points to $p$, velocity convergence remains equal to the exact quadrature, but the pressure convergence rate and constant worsen. The matrix is not invertible in the case with the lowest order discretization ($p=1$, $\varsigma=0$) and the homogeneous reduction scheme using $p$ quadrature points. Any reduction beyond $p$ quadrature points deteriorates the convergence of both velocity and pressure.

5.2. Two-dimensional lid-driven square cavity

We solve the two-dimensional lid-driven cavity test for the Stokes and Navier-Stokes equations, using the same set of nested meshes from 16 to 512 elements per side, as in the previous example to compare the solutions. The solutions found for the Stokes problem are compared to a spectral approach [25] in Table 3, comparing the value of the vorticity at a specified point near the top right corner ($x=(1,0.95)$), for discretizations using $p=1, 2, 3$ and maximum continuity. The solution for the Navier-Stokes problem uses two different Reynolds numbers $Re=100$ and...
Re = 400, and we compare with the results presented by Ghia in [26] and the spectral approach [25]. Tables 4 and 5 compare the value and position of the minimum horizontal velocity along the vertical centerline (x = 0.5), and the value and position of the minimum and maximum vertical velocity along the horizontal centerline (y = 0.5). Figures 14 and 15 illustrate the effect of the mesh distortion for the case of Navier-Stokes, where the results found with the coarsest mesh (h = $\frac{1}{16}$, $p = 1, \varsigma = 0$) without distortion, are compared to the results found using a distorted mesh (d = 0.45) with the same discretization.

Table 3: Convergence of the vorticity @ x= (1, 0.95) for the Stokes problem.

<table>
<thead>
<tr>
<th>Method</th>
<th>h</th>
<th>d = 0</th>
<th>d = 0.45</th>
<th>p = 1, $\varsigma = 0$</th>
<th>d = 0</th>
<th>d = 0.45</th>
<th>p = 2, $\varsigma = 1$</th>
<th>d = 0</th>
<th>d = 0.45</th>
<th>p = 3, $\varsigma = 2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\frac{1}{6}$</td>
<td>-0.528094</td>
<td>1.273373</td>
<td>12.947509</td>
<td>11.517694</td>
<td>32.790408</td>
<td>22.523328</td>
<td>25.422708</td>
<td>24.767150</td>
<td>27.342346</td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{5}$</td>
<td>18.075800</td>
<td>9.838386</td>
<td>33.277310</td>
<td>23.387877</td>
<td>22.522894</td>
<td>29.289110</td>
<td>23.479465</td>
<td>22.074412</td>
<td>25.848790</td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{4}$</td>
<td>19.186815</td>
<td>17.129618</td>
<td>35.017081</td>
<td>27.772500</td>
<td>30.291823</td>
<td>28.185708</td>
<td>23.479465</td>
<td>22.074412</td>
<td>25.848790</td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{3}$</td>
<td>23.479465</td>
<td>22.074412</td>
<td>25.848790</td>
<td>27.650488</td>
<td>29.324554</td>
<td>27.356594</td>
<td>23.479465</td>
<td>22.074412</td>
<td>25.848790</td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{2}$</td>
<td>25.425581</td>
<td>24.767150</td>
<td>27.342346</td>
<td>27.378879</td>
<td>27.642689</td>
<td>27.286833</td>
<td>25.425581</td>
<td>24.767150</td>
<td>27.342346</td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{1}$</td>
<td>26.371713</td>
<td>26.060740</td>
<td>27.294087</td>
<td>27.303925</td>
<td>27.278365</td>
<td>27.279689</td>
<td>26.371713</td>
<td>26.060740</td>
<td>27.294087</td>
</tr>
<tr>
<td>Spectral [23]</td>
<td>27.27901</td>
<td>-</td>
<td>27.27901</td>
<td>-</td>
<td>27.27901</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4 shows how the values found for the vorticity at a point that is near to the discontinuity of the velocity, located at the top right corner of the cavity, converge to the results found with a highly accurate spectral method of order 48 [25]. Results found with a discretization of $p = 1$ and $\varsigma = 0$ converge to the benchmark solution at a slower rate than the other discretizations, a finer mesh than the ones tested here is needed to resolve the corner singularity in this case. Higher order discretizations converge to within the first two significant digits of the spectral solution with a mesh of 256 × 256 elements, and within four significant digits with $p = 3$ and $\varsigma = 2$ when using a mesh of 512 × 512 elements. Results found with the distorted mesh converge to those of the uniform mesh showing the robustness of the discretization used for the velocity field.

Table 4: Velocity extrema for the Navier-Stokes problem (Re = 100) using uniform meshes (d = 0).

<table>
<thead>
<tr>
<th>Discretization</th>
<th>h</th>
<th>$\mu_{\text{min}}$</th>
<th>$y_{\text{min}}$</th>
<th>$\nu_{\text{min}}$</th>
<th>$\mu_{\text{max}}$</th>
<th>$\nu_{\text{max}}$</th>
<th>$x_{\text{min}}$</th>
<th>$\varsigma_{\text{max}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 1, \varsigma = 0$</td>
<td>$\frac{1}{6}$</td>
<td>-0.2201506</td>
<td>0.43750</td>
<td>-0.2605222</td>
<td>0.81249</td>
<td>0.1851086</td>
<td>0.25000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{5}$</td>
<td>-0.2164707</td>
<td>0.45703</td>
<td>-0.2538092</td>
<td>0.80859</td>
<td>0.1795948</td>
<td>0.23828</td>
<td></td>
</tr>
<tr>
<td>$p = 2, \varsigma = 1$</td>
<td>$\frac{1}{6}$</td>
<td>-0.2142675</td>
<td>0.45766</td>
<td>-0.2537870</td>
<td>0.81140</td>
<td>0.1797504</td>
<td>0.23706</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{5}$</td>
<td>-0.2140423</td>
<td>0.45808</td>
<td>-0.2538029</td>
<td>0.81042</td>
<td>0.1795728</td>
<td>0.23698</td>
<td></td>
</tr>
<tr>
<td>$p = 3, \varsigma = 2$</td>
<td>$\frac{1}{6}$</td>
<td>-0.2140613</td>
<td>0.45808</td>
<td>-0.2539128</td>
<td>0.81026</td>
<td>0.1796009</td>
<td>0.23679</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{5}$</td>
<td>-0.2140423</td>
<td>0.45808</td>
<td>-0.2538029</td>
<td>0.81042</td>
<td>0.1795728</td>
<td>0.23698</td>
<td></td>
</tr>
<tr>
<td>Spectral [23]</td>
<td>-0.2140424</td>
<td>0.4581</td>
<td>-0.2538030</td>
<td>0.8104</td>
<td>0.1795728</td>
<td>0.237</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Finite differences [26]</td>
<td>-0.210990</td>
<td>0.4531</td>
<td>-0.24533</td>
<td>0.8047</td>
<td>0.17527</td>
<td>0.2344</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Tables 4 and 5 compare the value of the velocity extrema when solving the Navier-Stokes system with a Reynolds number of one hundred, and four hundred, respectively, when using undistorted meshes, against the solution using a spectral method of order 96 [25] for the case of Re = 100 and a second order upwind finite differences method using 129 × 129 points [26] for both Reynolds numbers. We compare the values of the maximum horizontal velocity and its position along the vertical center line, and the values of the maximum and minimum vertical velocity along the horizontal center line, for the coarsest (h = $\frac{1}{16}$) and the finest meshes (h = $\frac{1}{256}$) used, and discretizations of $p = 1, 2, 3$ and maximum continuity. For both Reynolds numbers considered, all the results are reasonably close to the benchmark values, with the exception of the coarsest mesh when using the $p = 1, \varsigma = 0$ discretization, which is the only one that differs noticeably from the others. When using discretizations of $p > 1$ the differences between the results from
the coarsest and finest meshes become small, suggesting that a high order discretization with a coarse mesh may be enough to capture most of the features of the flow inside the domain.

Table 5: Velocity extrema for the Navier-Stokes problem \((Re = 400)\) using uniform meshes \((d = 0)\).

<table>
<thead>
<tr>
<th>Discretization</th>
<th>(h) (\frac{1}{16})</th>
<th>(h_{\text{min}})</th>
<th>(y_{\text{min}})</th>
<th>(v_{\text{min}})</th>
<th>(x_{\text{min}})</th>
<th>(v_{\text{max}})</th>
<th>(x_{\text{max}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(p=1, \varsigma = 0)</td>
<td>(\frac{1}{16})</td>
<td>-0.3523864</td>
<td>0.25000</td>
<td>-0.4920310</td>
<td>0.87499</td>
<td>0.3312674</td>
<td>0.24999</td>
</tr>
<tr>
<td></td>
<td>(\frac{1}{256})</td>
<td>-0.3288927</td>
<td>0.28124</td>
<td>-0.4542830</td>
<td>0.86328</td>
<td>0.3039886</td>
<td>0.22656</td>
</tr>
<tr>
<td>(p=2, \varsigma = 1)</td>
<td>(\frac{1}{16})</td>
<td>-0.3337101</td>
<td>0.28140</td>
<td>-0.4547631</td>
<td>0.85979</td>
<td>0.3078021</td>
<td>0.22429</td>
</tr>
<tr>
<td></td>
<td>(\frac{1}{256})</td>
<td>-0.3287303</td>
<td>0.28002</td>
<td>-0.4540652</td>
<td>0.86220</td>
<td>0.3038326</td>
<td>0.22530</td>
</tr>
<tr>
<td>(p=3, \varsigma = 2)</td>
<td>(\frac{1}{16})</td>
<td>-0.3298355</td>
<td>0.28047</td>
<td>-0.4550065</td>
<td>0.86134</td>
<td>0.3047172</td>
<td>0.22599</td>
</tr>
<tr>
<td></td>
<td>(\frac{1}{256})</td>
<td>-0.3287302</td>
<td>0.28002</td>
<td>-0.4540654</td>
<td>0.86221</td>
<td>0.3038325</td>
<td>0.22530</td>
</tr>
</tbody>
</table>

Finite differences \([26]\)

-0.32726 | 0.2813 | -0.44993 | 0.8594 | 0.30203 | 0.2266

Figures 14 and 15 illustrate the effect of the mesh distortion when using a \(p = 1, \varsigma = 0\) discretization and the coarsest mesh, by comparing the results found when solving the Navier-Stokes problem for two Reynolds numbers \((Re = 100 \text{ and } Re = 400)\) with the undistorted and the distorted mesh against the results found with the second order upwind finite differences method using \(129 \times 129\) points \([26]\). These comparisons show that the discretization is robust with respect to the mesh distortion and that even the coarsest discretization provides a fair approximation to the benchmark.

Figure 14: Comparison of vertical and horizontal velocities along the horizontal and vertical centerlines, respectively, for uniform \((d = 0.0)\) and distorted \((d = 0.45)\) meshes, solving the Navier-Stokes problem in a unitary square with \(Re = 100\), using \(h = \frac{1}{16}\) and \(p = 1, \varsigma = 0\). Our numerical results with the two different meshes compare favorably to Ghia’s benchmark \([26]\).
Figure 15: Comparison of vertical and horizontal velocities along the horizontal and vertical centerlines, respectively, for uniform \((d = 0.0)\) and distorted \((d = 0.45)\) meshes, solving the Navier-Stokes problem in a unitary square with \(Re = 400\), using \(h = \frac{1}{16}\) and \(p = 1, \zeta = 0\). Our numerical results with the two different meshes compare favorably to Ghia’s benchmark \([26]\).

5.3. Cylindrical Couette flow

We present results for a Couette flow in an annulus to test convergence in a physical domain different than a square. The flow is driven by a boundary condition of a unitary tangential velocity on the inner face of the annulus. We test the solutions for the Navier-Stokes equation for two different Reynolds numbers \(Re = 1\) and \(Re = 100\). The analytical solution for the velocity and pressure when considering \(Da = 0\), as shown in Figure 16 are given by the following expression:

\[
\bar{u} = \begin{bmatrix} (Ar + \frac{B}{r}) \sin(\theta) \\ (Ar + \frac{B}{r}) \cos(\theta) \end{bmatrix}, \quad \frac{\partial p}{\partial r} = \left(\frac{Ar + \frac{B}{r}}{r}\right)^2
\]

where \(r\) and \(\theta\) correspond to the polar coordinates, and

\[
A = -\frac{U \delta^2}{r_{in}(1-\delta^2)}, \quad B = \frac{U r_{in}}{(1-\delta^2)}, \quad \delta = \frac{r_{in}}{r_{out}}.
\]

The domain is defined by the inner radius \(r_{in} = 1\) and the outer radius \(r_{out} = 2\). The simulations use the analytical mapping described in equation (7), where \(d\) indicates the distortion from the polar mapping, generating a mesh as shown in Figure 17. The results found using the analytical mapping are shown in Figures 18 and 19 for the Navier-Stokes one.

\[
F(\xi_1, \xi_2) = \begin{cases} (d \cos(2\alpha \xi_2) (\xi_1^2 - \xi_1) + \xi_1 + 1) \cos(2\pi \xi_2) \\ (d \cos(2\alpha \xi_2) (\xi_1^2 - \xi_1) + \xi_1 + 1) \sin(2\pi \xi_2) \end{cases}, \forall (\xi_1, \xi_2) \in \bar{\Omega}, \alpha \in \mathbb{Z}, d \in [-1, 1]
\]
(a) Velocity magnitude.  
(b) Pressure.  

Figure 16: Analytical solution for the Couette flow for the Navier-Stokes system.

Figure 17: Meshes of 4×16 elements used to discretize the domain using the analytical mapping. Mesh with $d=0$ on the left, and $a=5, d=0.5$ on the left.

Figure 18: Convergence test results for Navier-Stokes $Re=1$ in the Couette flow problem using an analytical mapping.
Figures 18 and 19 show that in this case, both velocity and pressure converge at a rate \( r = p + 1 \), for uniform and distorted meshes.

5.4. Solution in a unitary cube

Here we test our 3D implementation of the Darcy, Stokes, Brinkman, and Navier-Stokes problems against a three-dimensional manufactured solution. The body force \( \mathbf{f} \) is defined by introducing the coefficients \( \alpha, \beta \) and \( \nu \), and the prescribed solution for \( \mathbf{u} \) and \( p \) into equation (6). The expressions for \( \mathbf{u} \) and \( p \) as shown in Figure 20, are defined as

\[
\mathbf{u} := \nabla \times \phi, \quad p := \sin(\pi x) \sin(\pi y) - \frac{4}{\pi^2},
\]

where

\[
\phi := \begin{bmatrix}
 x(x-1)y^2(y-1)^2(z-1)^2 \\
 0 \\
 x^2(x-1)^2y^2(y-1)^2(z-1)
\end{bmatrix}.
\]

To test our implementation in the parametric and physical domains a set of nested uniform and distorted meshes, from \( 4 \times 4 \times 4 \) to \( 64 \times 64 \times 64 \) elements, were considered for this case. The distorted mesh was built by moving the central control point of a single second order element, a distance \( d \) in the positive direction of every axis and then performing an \( h \)-refinement. An example of the meshes used is shown in Figure 21.
Table 6 shows convergence rates for the error in the pressure. These rates are equal to $p + 1$, when using both uniform and distorted meshes, with the exception of a uniform mesh and a discretization of $p = 1$, when the convergence rate falls between $p + 1$ and $p + 2$. Convergence rates for the error in the velocity are not presented in a table, since these are consistently equal to $p + 1$ for all equations, and are not affected by the mesh distortion. Figure 22 shows the convergence of velocity and pressure in the case of Stokes, similar convergence results were found for the other equations. Uniform meshes using a discretization with $p = 3$ solve exactly the fourth order polynomial given for the prescribed solution of the velocity, hence the velocity convergence results for $p = 3$ and $d = 0$ are not shown in Figure 22. While the same order discretization, $p = 3$, using the distorted mesh has a convergence order of $p + 1$, due to the B-spline based mapping.
5.5. Three-dimensional lid-driven cavity

We solve the three-dimensional lid-driven cavity test for the Stokes and Navier-Stokes systems, using a set of nested meshes, with and without distortion, from $8 \times 8 \times 8$ to $32 \times 32 \times 32$ elements. We compare our solution for the Stokes problem with a differential quadrature method using a mesh of $25 \times 25 \times 25$ [27]. We solve the Navier-Stokes problem for two different Reynolds numbers $Re = 100$ and $Re = 400$, and compared with the results presented by Lo in [28] using a finite difference method, solving the case for $Re = 100$ and $Re = 400$ with a $51 \times 51 \times 51$ and $101 \times 101 \times 101$ mesh, respectively. We compare our simulation results with those presented by Wong in [29], using the finite element method to solve the velocity-vorticity formulation with $48 \times 48 \times 48$ elements.

Table 7: Convergence of the velocity extrema for the Stokes and Navier-Stokes problem in a cube using uniform meshes ($d=0$).

| Discretization | $h$ | Stokes $Re=100$ | | | Stokes $Re=400$ | | |
|----------------|-----|----------------|----------------|----------------|----------------|----------------|
|                | $u_{\min}$ | $z_{\min}$ | $u_{\min}$ | $z_{\min}$ | $u_{\min}$ | $z_{\min}$ |
| $p=1$, $\varsigma=0$ | $\frac{1}{8}$ | -0.21946 | 0.50000 | -0.24737 | 0.50000 | -0.34257 | 0.25000 |
| | $\frac{1}{16}$ | -0.21070 | 0.56249 | -0.22176 | 0.50000 | -0.27280 | 0.25000 |
| | $\frac{1}{32}$ | -0.20868 | 0.53125 | -0.21764 | 0.46875 | -0.24539 | 0.25000 |
| $p=2$, $\varsigma=1$ | $\frac{1}{8}$ | -0.20796 | 0.53237 | -0.21774 | 0.46856 | -0.25739 | 0.24299 |
| | $\frac{1}{16}$ | -0.20771 | 0.53468 | -0.21575 | 0.46830 | -0.24086 | 0.24008 |
| | $\frac{1}{32}$ | -0.20776 | 0.53605 | -0.21560 | 0.46923 | -0.23702 | 0.23885 |
| Differential quadrature [27] | -0.231 | - | -0.215 | - | -0.236 | - |
| Finite differences [28] | - | - | -0.2163 | 0.46 | -0.2334 | 0.26 |
| Finite elements [29] | - | - | -0.2154 | 0.4592 | -0.2349 | 0.2509 |

Table 7 compares the minimum horizontal velocity $u$ and its position over the vertical centerline ($x=0.5$, $y=0.5$), against solutions using differential quadrature [27], finite differences [28] and finite element method [29]. For these Reynolds numbers the simulation results are reasonably close to the benchmark values when using meshes with $h = \frac{1}{8}$ and $p=1$, especially for the case of $p=2$, $\varsigma=1$. When using discretizations of $p=2$ the differences between the results of using meshes with $h = \frac{1}{8}$ and $h = \frac{1}{16}$, are small, suggesting that a mesh $h = \frac{1}{16}$ is enough to represent the flow inside the domain.

Figure 23: Comparison of horizontal velocity $u$ along the vertical centerline for uniform ($d=0.0$) and distorted ($d=0.45$) meshes, solving the Stokes problem in a cube, using $h = \frac{1}{16}$ and $p=1$, $\varsigma=0$. 

Figure 24: Comparison of horizontal velocity along the vertical centerline for uniform ($d = 0.0$) and distorted ($d = 0.45$) meshes, solving the Navier-Stokes problem in a cube with $Re = 100$ and $Re = 400$, using $h = \frac{1}{16}$ and $p = 1$, $\varsigma = 0$. Our numerical results with the two different meshes compare favorably to Wong’s benchmark [29].

Figures 23 and 24 show the results of the horizontal velocity along the vertical centerline, found with a discretization using $h = \frac{1}{16}$, $p = 1$, $\varsigma = 0$ with uniform and distorted meshes, and compares them with the ones reported by Wong [29]. In the three cases, the results obtained using a distorted mesh are very similar to those of the uniform meshes, showing no major effect of the mesh distortion over the velocity convergence. Results obtained for the Navier-Stokes equations compare well to those of Wong, being indistinguishable from the benchmark in the case of $Re = 100$.

6. Conclusions

We introduce a framework, called PetIGA-MF, for multi-field high-performance isogeometric analysis. PetIGA-MF provides structure-preserving vector field discretizations to solve multi-physics problems. This framework allows us to use different approximation spaces for each component of the discrete fields. This flexibility simplifies the implementation and discretization of complex multi-field problems while guaranteeing stability. We extend PetIGA and adapt PETSc to manage the parallelism, and offer access to a significant variety of solvers and preconditioners.

We test our simulation framework with numerous benchmarks and evaluate the effect of distorting the mesh on the convergence rates, finding optimal convergence rates for the velocity and pressure in all cases. When using uniform meshes the convergence rates for the pressure are equal to those of the velocity, although its discretization uses spaces with one order lower polynomials. Under mesh distortion, the convergence rates for the pressure decreases, by almost one order, which is closer to the predicted limit of the a priori error estimates. These results lead us to conclude that the error estimates for pressure presented in [8] are not conservative, but only distorted meshes and non-trivial geometries may cause convergence rates to decrease to the predicted limit. An in-depth analysis of the effects of mesh distortion on the error estimates is required to understand better the circumstances under which this loss of superconvergence may occur.

Our choice of discrete divergence-conforming spaces for velocity and pressure guarantee an accurate solution of the flow and pointwise conservation of mass. Weak imposition of boundary conditions yields accurate results when focusing on flow near boundaries while avoiding instabilities due to over restricting the velocity space.

7. Acknowledgments

This publication was made possible in part by a National Priorities Research Program grant 7-1482-1-278 from the Qatar National Research Fund (a member of The Qatar Foundation), by the European Union’s Horizon 2020 Research and Innovation Program of the Marie Skłodowska-Curie grant agreement No. 644602 and the Center for Numerical
Porous Media at King Abdullah University of Science and Technology (KAUST). L. Dalcin was partially supported by Agencia Nacional de Promoción Científica y Tecnológica grants PICT 2014–2660 and PICT-E 2014–0191. The J. Tinsley Oden Faculty Fellowship Research Program at the Institute for Computational Engineering and Sciences (ICES) of the University of Texas at Austin has partially supported the visits of VMC to ICES.

8. References


[27] URL http://www.mcs.anl.gov/petsc

[28] [arXiv:1308.3339]
