A STABILIZED DISCONTINUOUS GALERKIN METHOD FOR VARIATIONAL EMBEDDING OF PHYSICS-BASED DATA

BY

SHOAIB AHMAD GORAYA

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Civil Engineering
with a concentration in Computational Science and Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2019

Urbana, Illinois

Adviser:

Professor Arif Masud

ABSTRACT

A stabilized variational framework that admits overlapping as well as non-overlapping coupling of domains for a variety of Partial Differential Equations (PDEs) is employed in this work. This method accommodates non-matching meshes across the interfaces between the subdomain boundaries and allows for sharp changes in mechanical material properties. Interface coupling operators that emanate via embedding of Discontinuous Galerkin ideas in the continuous Galerkin framework provide a unique avenue to embed physics-based data in the modeling and analysis of the system. Physics-based data, either in discrete or in distributed form can be embedded via the interface operators that are otherwise devised to enforce continuity of the fields across internal discontinuities. The least-squares form of the interface coupling operators is exploited for its inherent linear regression type structure, and it is shown that it helps improve the overall accuracy of the numerical solution. Method is applicable to multi-PDE class of problems wherein different PDEs are operational on adjacent domains across the common interface. The method also comes equipped with a residual based error estimation method which is shown to be applicable to test problems employed. Different test cases are employed to investigate the mathematical attributes of the method.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

Data Science is the extraction of 'features' or 'useful knowledge' from large volumes of unstructured data [1]. Typically, it is done through a combination of various tools, algorithms, and machine learning principles with the aim to get 'insights' from the raw data, which are then used for decision making in a variety of fields. Data Science allows computational models to learn representations of data with multiple levels of abstraction. In recent years, machine learning based computational models have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics [32].

Recently, there has been a trend of incorporating Data Science into computational mechanics, however its full potential is yet to be realized. In computational mechanics, machine learning has been used to formulate multiscale elements [31], to enhance the performance of traditional elements [45], to extract constitutive manifolds [28], to produce a data-driven solver [29], and to develop surrogate models for computing element internal forces [11].

The core of computational mechanics has been to find approximate solutions for a variety of partial differential equations, which describe natural, physical, and chemical phenomena mathematically. Finite Element Method (FEM) is one of the numerical methods that forms the basis of computational mechanics. It has been extremely popular and widely used for a variety of applications over the last half century due to its robustness and systematic approach. Unfortunately, the method has limitations for problems with discontinuities, singularities, and multiple scales. Some stabilization methods have been proposed to tackle these technical bottlenecks. Successful methods include the global-local methods [43], residual free bubbles [10],

the Variational Multiscale (VMS) method [25], the discontinuous enrichment method [15], the generalized finite element method [16], and multiscale methods [13]. One common feature of these methods is that they expand or modify the solution space to better fit the particular problem. Although these methods tend to have a better computational performance, the convergence rates are sub-optimal and get prohibitively expensive to achieve a certain accuracy.

In VMS method, a decomposition is applied to the solution field into resolved scales, which are captured by a given mesh, and unresolved scales, i.e., the finer features that are beyond the resolution capacity of the given spatial mesh, thus giving rise to discretization error right from the outset. This decomposition is introduced into the variational structure of the problem and gives rise to a system of coupled equations for the resolved and unresolved scales. This system of variational equations serves as a launching point for the derivation of enhanced numerical methods termed as VMS-based methods.

One of the applications of the VMS method is to use the fine-scale component either as a posteriori error estimator or as a driving component for adaptive meshing algorithms [47,31]. A unified, consistent framework for estimating discretization error through the modeling of fine-scale effects is presented in [39].

Another area where VMS has led to significant contributions is the development of stabilized methods. In recent years, stabilized formulations have been pursued through the application of the variational multiscale method to the mixed form of elasticity and the Stokes flow problem, which are form-equivalent in the incompressible limit. In [40,41], Masud and co-workers applied the variational multiscale method to the mixed form of elasticity and derived the structure of the stabilization matrix $\boldsymbol{\tau}$ by employing bubble functions to represent the fine scales. Employing the

VMS-based stabilization where the fine-scale problem is treated in a direct fashion via bubble functions, Masud and Franca [34] developed a hierarchical multiscale modeling framework for problems with multiscale source terms, and Masud and Scovazzi [37] developed a heterogeneous multiscale modeling framework for hierarchical systems of nested PDEs.

As discussed earlier, numerical methods have limitations when it comes to solving problems with discontinuous interfaces. The development of robust techniques for modeling interfaces within the finite element method has been an active area of research in recent years. Barbosa and Hughes [5] proposed Galerkin least-squares (GLS) stabilizing terms involving the Euler–Lagrange residual on the boundary to weakly enforce Dirichlet constraints. An alternative for imposing constraints is the consistent penalty formulation called the Nitsche method [42], in which numerical flux terms defined from the primary field stand-in for the multipliers. A limitation of the classical Nitsche or GLS approaches [42,5] is that the explicit definition of the stabilization parameter is left unspecified by the theory. Although simple approximations or global estimates for the stabilization parameter are satisfactory for most situations, robust estimates become crucial in the presence of sharply varying material properties, and anisotropic nonconforming meshes.

Truster and Masud [53] derived a Nitsche interface method from an underlying Lagrange multiplier method by applying a VMS approach. Starting from a Lagrange multiplier formulation for imposing continuity constraints weakly at the interface, a multiscale decomposition is applied to the primary field locally at the interface. Recognizing that the discretization process induces instabilities not present in the continuum problem, models for the fine-scale features are incorporated in order to enhance the stability in the neighborhood of the interface. By modeling these fine-scale features using edge bubble functions, consistent residual-driven terms at the interface that stabilize the mixed primal-multiplier formulation are obtained. By employing

discontinuous functions for the Lagrange multipliers along segments of the interface, a definition for the numerical flux emerges that, upon substitution into the mixed form, yields a DG interface method where the primary field is the only unknown.

The resulting DG interface method provides an avenue for mathematically linking hierarchal physical models in a consistent fashion [21]. A common example is flow through a porous media where regions with low permeability are modeled with Darcy equation while the regions with higher flow rates are modeled via Darcy-Stokes equations. The multiscale phenomenon needs exchange of information from one model to the other through coupling of models via a DG interface.

In this work, we have incorporated the observational data into the VMS based DG interface method [53] in order to enhance accuracy of the solution fields. The data is provided at nodal points in a small patch which is embedded as a discontinuity in the problem domain. This provides an avenue to introduce data in interface coupling terms, wherein least squares type linear regression function emerges. In this new framework, essential constraints and conservation laws such as compatibility and equilibrium remain unchanged, as do the spatial and temporal discretizations. Therefore, the variational consistency of the DG interface method carries over to the present data-driven paradigm. The goal of embedding data is to find the state satisfying the conservation laws that is closest to the data set. The resulting data-driven problem thus consists of the minimization of a distance function to the data set subject to the satisfaction of essential constraints and conservation laws. This is where the notion of physics informed data-driven modeling appears.

An outline of the thesis is as follows: Chapter 2 presents a general framework employing VMS ideas for deriving DG interface method to couple nonconforming meshes as well as different PDEs

across discrete interfaces, wherein consistent expressions for the stabilization parameter and weighted numerical flux emerge. Chapter 3 provides a derivation of the stabilized form for mixed form of elasticity and a residual based error estimation framework using scale split approach of VMS. Chapters 4 and 5 specialize the general DG interface framework presented in Chapter 2 for different solid and fluid mechanics PDEs, coupling of different PDEs, and linking of hierarchal physical models. Chapter 6 describes the approach for variationally embedding data in the DG interface method and the interpretation of the emerging terms in the context of Data Science. A series of numerical test cases for both solid and fluid mechanics problems with embedded data are presented in Chapter 7. Finally, Chapter 8 contains concluding remarks.

CHAPTER 2: GENERAL FRAMEWORK FOR COUPLING MULTIPLE SUBDOMAINS

This chapter presents the general framework that underlies the method. $\mathcal{A}^{(\alpha)} : \mathcal{V}^{(\alpha)} \to \mathcal{V}'^{(\alpha)}$ represents linear (differential) operators of the governing system of equations in subdomain $\Omega^{(\alpha)}$. We require that $\mathcal{A}^{(\alpha)}$ are such that the fields $\boldsymbol{u}^{(\alpha)}$ have the same physical connotation (e.g. displacement, velocity, or concentration) throughout Ω and that the local equations are well-posed in the sense of the Lax-Milgram theorem.

$$\mathcal{A}^{(\alpha)}\boldsymbol{u}^{(\alpha)} + \boldsymbol{f}^{(\alpha)} = \boldsymbol{0} \qquad \text{in } \Omega^{(\alpha)} \tag{1}$$

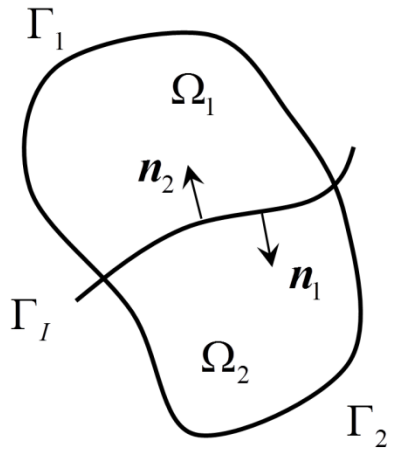


Figure 1: Domain cut into subdomains across the common interface

Figure 1 shows the domain cut into subdomain across which the functions are assumed to be discontinuous. To connect the two regions at the interface, the domain interior (1) is supplemented with the following continuity equations:

$$A^{(1)}\boldsymbol{u}^{(1)} - \boldsymbol{\lambda} = \boldsymbol{0} \qquad \text{on } \Gamma_I$$

$$A^{(2)}\boldsymbol{u}^{(2)} + \boldsymbol{\lambda} = \mathbf{0} \qquad \text{on } \Gamma_I \tag{2}$$

$$B^{(1)}\boldsymbol{u}^{(1)} = B^{(2)}\boldsymbol{u}^{(2)} \qquad \text{on } \Gamma_I \tag{3}$$

where the Lagrange multiplier field $\boldsymbol{\lambda} \in \mathcal{W}$ plays the role of the flux across the interface. In (2) and (3) the linear operators $A^{(\alpha)}$ are the flux operators associated with the interior operators $\mathcal{A}^{(\alpha)}$. The trace operators $B^{(\alpha)}$ map the primary field $\boldsymbol{u}^{(\alpha)}$ to its trace on Γ_I for pairing with the multiplier μ. Because the continuity condition (4) is weakly enforced, the finite element mesh is permitted to be nonconforming along the interface Γ_I.

In (1) – (4) the unknown fields are $\boldsymbol{u}^{(\alpha)}$ and $\boldsymbol{\lambda}$. While the continuum formulation (1) – (4) is well-posed, its discrete counterpart does not inherit the same stability properties, which leads to technical issues when the classical continuous Galerkin approach is applied to the individual subproblems in (1) for $\alpha \in \{1,2\}$. Namely, the formulation can be characterized as a mixed field problem, and therefore the selection of piecewise continuous or discontinuous multipliers as well as the polynomial order has significant implications on the stability of the discrete problem. In particular, employing combinations which do not satisfy the Babuška–Brezzi condition may lead to unstable or oscillatory numerical results.

Secondly, in the context of mathematical physics, Lagrange multiplies has a force connotation, and it provides the magnitude of force required to keep the two pieces together. For the case where interfacial kinematics goes beyond gluing the two pieces and may encompass relative slipping or sliding of the interfaces, Lagrange multiplies methods does not provide provision to embed interfacial kinematic models. Furthermore, this problem cannot be modeled via Penalty function

formulation as well, as penalty parameter is problem dependent, and depending on its numeric value employed, the constraint is either loosely enforced, or a too large a value can lead to ill-conditioning of the discrete system of equations.

In this context, Nitsche method provides another avenue to couple the subdomains while accommodating interfacial kinematics. While the relationship between the Nitsche method and stabilized Lagrange multiplier methods has been established, this section presents a systematic approach for deriving methods associated with a wider class of linear PDEs, including the capability of treating different governing equations in each of the subregions.

The underlying variational structure is that of the VMS method. The guiding philosophy of this method is that the weak continuity of the fields can introduce error in the discrete representation and the lack of resolution of the fine-scale features by a given discretization is the major factor in the loss of stability for many numerical techniques.

We remark that for mixed field problems, a decomposition may not be required for each primary field in order to achieve a stable method. Secondly, no continuity requirements regarding the interface are placed on the coarse or fine scales. The coarse scales $\hat{\boldsymbol{u}}^{(1)}$, $\hat{\boldsymbol{u}}^{(2)}$ are associated with finite element spaces $\mathcal{V}^{h(1)}$ and $\mathcal{V}^{h(2)}$, respectively, and the fine scales are taken to lie in the complement spaces $\mathcal{V}^{(1)} \setminus \hat{\mathcal{V}}^{(1)}$ and $\mathcal{V}^{(2)} \setminus \hat{\mathcal{V}}^{(2)}$.

Segregation of spaces of functions leads to coarse-scale problem and a fine-scale problem. The objective at this point is to derive an analytical expression for the fine scales in terms of the Euler-Lagrange equations for the coarse-scale fields $\hat{\boldsymbol{u}}^{(\alpha)}$ and $\boldsymbol{\lambda}$.

Secondly, for computational expediency we assume that the method relies upon the assumptions that the fine-scale field is localized to the elements adjacent to Γ_I and does not vanish along Γ_I. Upon substitution of this expression into the coarse-scale problem, the fine-scale effects at the interface produce a stabilized mixed formulation. With the additional assumption of discontinuous Lagrange multipliers, a stabilized weak form for the primary fields $\boldsymbol{u}^{(\alpha)}$ alone is obtained in which consistent definitions emerge for the weighted numerical flux and the penalty parameter.

To facilitate the derivation of the fine-scale models at the interface, we first introduce notation associated with the partition of the interface Γ_I into segments that are induced by the coarse-scale discretizations $\mathcal{T}^{(1)}$ and $\mathcal{T}^{(2)}$. Next, we approximate the fine-scale fields using edge bubble functions. However, the support of the bubble $b_s^{(\alpha)}$ is taken to be the tributary area or sector $\omega_s^{(\alpha)} \subseteq \Omega_{e_s}^{(\alpha)}$ which respects the interface segment γ_s, namely $\partial\omega_s^{(1)} \bigcap \tilde{\Gamma}_I = \partial\omega_s^{(2)} \bigcap \tilde{\Gamma}_I = \gamma_s$. The concept of segments and sectors associated with a discretized interface is illustrated in Figure 2. The specific form of the edge bubble function for various element types is provided in Table 3 and an example is shown in Figure 3.

In summary, the fine scales in the neighborhood of segment γ_s are represented as follows:

$$\tilde{\boldsymbol{u}}^{(\alpha)}\Big|_{\omega_s^{(\alpha)}} = \sum_{k=1}^{\tilde{n}^{(\alpha)}} \beta_{sk}^{(\alpha)} b_s^{(\alpha)}(\boldsymbol{x}) \boldsymbol{e}_k^{(\alpha)}, \qquad \tilde{\boldsymbol{w}}^{(\alpha)}\Big|_{\omega_s^{(\alpha)}} = \sum_{l=1}^{\tilde{n}^{(\alpha)}} \eta_{sl}^{(\alpha)} b_s^{(\alpha)}(\boldsymbol{x}) \boldsymbol{e}_l^{(\alpha)} \tag{5}$$

Adopting this representation enables the separation of into a series of local problems associated with segments $\gamma_s \in \mathcal{T}_I$. At this point we introduce three assumptions to further simplify the derivation of the fine-scale model.

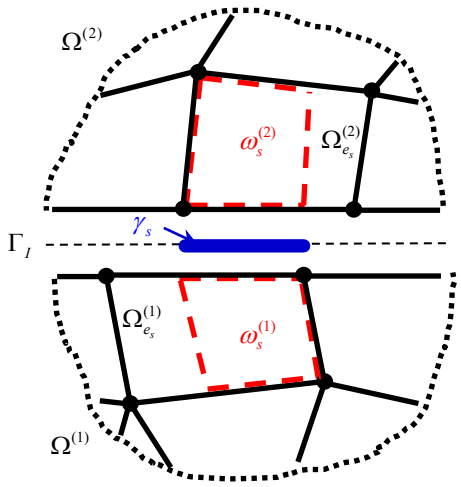


Figure 2: Interface segment γ_s

1. First, the residual of the coarse-scales in the element interior is assumed to be nearly orthogonal to the fine-scale bubble functions $b_s^{(\alpha)}$, which is equivalent to representing the fine scales using residual free bubbles. While we do not strictly enforce the condition $b_s^{(\alpha)} = b_{RFB}$, we argue that the assumption $\left(b_s^{(\alpha)} \boldsymbol{e}_l^{(\alpha)}, \mathcal{A}^{(\alpha)} \hat{\boldsymbol{u}}^{(\alpha)} + \boldsymbol{f}^{(\alpha)}\right)_{\omega_s^{(\alpha)}} \approx 0$ is valid because the residual is expected to be small for stable numerical methods and also to decrease as the mesh is refined.

2. Second, we focus on the trace of the fine scales along the interface and make an approximation which is analogous to employing the average value of the bubble on segment γ_s.

3. Third, the boundary residual $(-1)^{\alpha-1} \boldsymbol{\lambda} - A^{(\alpha)} \hat{\boldsymbol{u}}^{(\alpha)} \in \mathcal{W}$ is taken outside of the bilinear form $b_{\gamma_s}^{(\alpha)}(\bullet, \bullet)$ by applying a projection operator $\Pi_{\mathcal{W}'} : \mathcal{W} \to \mathcal{W}'$, which is equivalent to applying the mean value theorem. Combining these steps leads to an analytical expression for the trace of the fine scales on the interface.

To derive the Stabilized Interface formulation, we embed the fine-scale model into the coarse-scale formulation. Fine scale model is a function of Lagrange multiplier which is an unknown that needs to be derived. Through the incorporation of the stabilizing terms obtained via the fine-scale models, arbitrary combinations for the primary and multiplier fields are admissible in the weak form. To derive Lagrange multiplier, we focus on the continuity equation and employ the space of discontinuous L^2 functions on Γ_I. This approximation allows us to obtain a closed-form expression for the Lagrange multiplier field $\boldsymbol{\lambda}$ on each segment:

$$\boldsymbol{\lambda}\big|_{\gamma_s} = \left[\boldsymbol{\delta}_s^{(1)}\Pi_{\mathcal{W}}\left(A^{(1)}\hat{\boldsymbol{u}}^{(1)}\right) - \boldsymbol{\delta}_s^{(2)}\Pi_{\mathcal{W}}\left(A^{(2)}\hat{\boldsymbol{u}}^{(2)}\right)\right]\Big|_{\gamma_s} - \boldsymbol{\tau}_s\left(\Pi_{\mathcal{W}}\ \ \hat{\boldsymbol{u}}\right)\Big|_{\gamma_s} \tag{6}$$

where the tensorial quantities are derived from the fine-scale models and termed as follows:

$$\boldsymbol{\tau}_s = \left(\boldsymbol{\tau}_s^{(1)} + \boldsymbol{\tau}_s^{(2)}\right)^{-1}, \qquad \boldsymbol{\delta}_s^{(\alpha)} = \boldsymbol{\tau}_s\boldsymbol{\tau}_s^{(\alpha)} \tag{7}$$

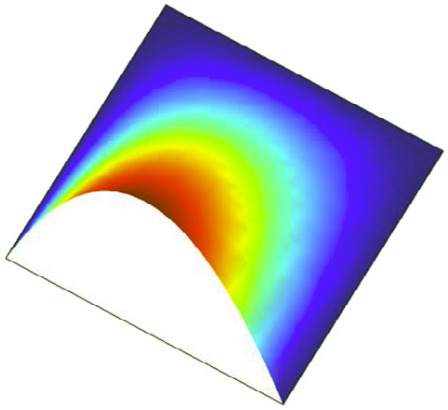


Figure 3: Sample edge bubble function

Embedding the fine model in the coarse-scale variational form leads to interface integrals that are form-equivalent to the classical DG or symmetric interior penalty method. However, the numerical

flux and the penalty parameter are explicitly derived. In addition, it gives rise to interface integrals that involve the product of flux jump terms defined as $[\![A\boldsymbol{u}]\!] = A^{(1)}\boldsymbol{u}^{(1)} + A^{(2)}\boldsymbol{u}^{(2)}$ and an inverse penalty parameter obtained through straightforward algebraic manipulation as $\boldsymbol{\delta}_s = \boldsymbol{\tau}_s^{(1)}\boldsymbol{\delta}_s^{(2)} = \boldsymbol{\tau}_s^{(2)}\boldsymbol{\delta}_s^{(1)} = \left[\left(\boldsymbol{\tau}_s^{(1)}\right)^{-1} + \left(\boldsymbol{\tau}_s^{(2)}\right)^{-1} \right]^{-1}$, which holds for arbitrary tensors $\boldsymbol{\tau}_s^{(1)}$ and $\boldsymbol{\tau}_s^{(2)}$. While this term is not commonly used in single field DG methods, it does appear in DG methods for mixed field problems such as Darcy flow. These four interface terms model the effects of the Lagrange multipliers and fine-scale fields which no longer explicitly appear in Stabilized Form.

CHAPTER 3: STABILIZATION AND ERROR ESTIMATION

3.1 Governing Equations

3.1.1 Strong Form

Let $\Omega \subset \mathbb{R}^{n_{\text{sd}}}$ be an open, bounded domain with a piece-wise smooth boundary Γ, where $n_{\text{sd}} \geq 2$ is the number of spatial dimensions. The boundary Γ is divided into two subsets Γ_g and Γ_h on which Dirichlet and Neumann conditions are applied, respectively, and these subsets satisfy $\Gamma_g \bigcap \Gamma_h = \Gamma, \Gamma_g \bigcap \Gamma_h = \varnothing$. With these definitions, the governing equations of linear elasticity are:

$$\nabla \cdot \boldsymbol{\sigma} + \boldsymbol{b} = \mathbf{0} \qquad \text{in } \Omega \tag{8}$$

$$\boldsymbol{u} = \boldsymbol{g} \qquad \text{on } \Gamma_g \tag{9}$$

$$\boldsymbol{\sigma n} = \boldsymbol{h} \qquad \text{on } \Gamma_h \tag{10}$$

where $\boldsymbol{u} : \Omega \to \mathbb{R}^{n_{\text{sd}}}$ represents the displacement field, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\boldsymbol{b}$ is the body force vector, $\boldsymbol{g}$ is the prescribed displacement, $\boldsymbol{h}$ is the prescribed traction, and $\boldsymbol{n}$ is the unit outward normal to Γ. A mixed form of isotropic elasticity capable of modeling the incompressible limit can be written via the following mixed constitutive law and associated kinematic equations:

$$\boldsymbol{\sigma} = p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) \tag{11}$$

$$\nabla \cdot \boldsymbol{u} = p/\lambda \tag{12}$$

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(\boldsymbol{u}) = \tfrac{1}{2}\left(\nabla \boldsymbol{u} + \left(\nabla \boldsymbol{u}\right)^T\right) \tag{13}$$

where $p : \Omega \to \mathbb{R}$ denotes the pressure field, $\boldsymbol{\varepsilon}$ is the linearized strain tensor, λ and μ are the Lame parameters, and $\boldsymbol{I}$ is the second-order identity tensor. All through the paper, the following conventions are used for vector and tensor operators: $\nabla(\bullet)$ represents the gradient, $\nabla\cdot(\bullet)$ is the divergence, $\Delta(\bullet)$ is the Laplacian, $\mathrm{tr}(\bullet)$ is the trace, and $(\bullet)^T$ is the transpose of the indicated quantity.

Substituting (11) into (8) – (10) gives:

$$\nabla p + \nabla\cdot\left(2\mu\boldsymbol{\varepsilon}(\boldsymbol{u})\right) + \boldsymbol{b} = \boldsymbol{0} \qquad \text{in } \Omega \tag{14}$$

$$\nabla\cdot\boldsymbol{u} = p/\lambda \qquad \text{in } \Omega \tag{15}$$

$$\boldsymbol{u} = \boldsymbol{g} \qquad \text{on } \Gamma_g \tag{16}$$

$$\left[p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u})\right]\boldsymbol{n} = \boldsymbol{h} \qquad \text{on } \Gamma_h \tag{17}$$

3.1.2 Weak Form

The functional spaces appropriate for the displacement and pressure trial solutions and weighting functions are:

$$\mathcal{S} = \left\{\boldsymbol{u} \,|\, \boldsymbol{u} \in \left(H^1(\Omega)\right)^{n_{\mathrm{sd}}}, \boldsymbol{u} = \boldsymbol{g} \text{ on } \Gamma_g\right\} \tag{18}$$

$$\mathcal{V} = \left\{\boldsymbol{w} \,|\, \boldsymbol{w} \in \left(H^1(\Omega)\right)^{n_{\mathrm{sd}}}, \boldsymbol{w} = \boldsymbol{0} \text{ on } \Gamma_g\right\} \tag{19}$$

$$\mathcal{P} = \left\{ p \middle| \, p \in L_2\left(\Omega\right) \right\} \tag{20}$$

where $L_2\left(\Omega\right)$ and $H^1\left(\Omega\right)$ are standard Sobolev spaces. The weak form corresponding to the governing equations can be expressed as: Find $\boldsymbol{u} \in \mathcal{S}, p \in \mathcal{P}$ such that for all $\boldsymbol{w} \in \mathcal{V}, q \in \mathcal{P}$:

$$\int_\Omega \nabla \boldsymbol{w} : \left[p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) \right] \mathrm{d}\Omega = \int_\Omega \boldsymbol{w} \cdot \boldsymbol{b} \, \mathrm{d}\Omega + \int_{\Gamma_h} \boldsymbol{w} \cdot \boldsymbol{h} \, \mathrm{d}\Gamma \tag{21}$$

$$\int_\Omega q\left(\nabla \cdot \boldsymbol{u} - p/\lambda \right) \mathrm{d}\Omega = 0 \tag{22}$$

While the standard weak form given above is well-posed in the continuum setting, the process of discretization gives rise to non-trivial issues that need further consideration. The discretized form is required to satisfy the Babuška–Brezzi (BB) inf-sup condition to ensure uniqueness and stability of the pressure field [7].

3.2 Variational Multiscale Method

In this method, the solution field is decomposed into coarse scales, which are resolved by a given mesh, and unresolved or fine scales that give rise to the discretization error. The motivation for employing the VMS method in the present context is two-fold: (i) the derivation of a stabilized formulation that can accommodate the incompressible limit, and (ii) the development of an error estimation framework. Error has two components: (i) local error, and (ii) global error. Local error involves an element-wise explicit calculation of the fine scales. Since fine scales are embedded in coarse-scales, these are used to get a better coarse-scale field and the difference between the improved and original coarse-scale fields results in global error.

3.2.1 Multiscale Decomposition

Consider a partition of the domain Ω into non-overlapping open subregions Ω^e, $e = 1,...,n_{umel}$, and n_{umel} is the number of elements in the mesh. Let Γ^e denote the boundary of element Ω^e, and let Ω' and Γ' denote the union of element interiors and element boundaries, respectively:

$$\Omega' = \bigcup_{e=1}^{n_{umel}} \Omega^e \tag{23}$$

$$\Gamma' = \bigcup_{e=1}^{n_{umel}} \Gamma^e \tag{24}$$

Finally, let $\Gamma_{\text{int}} = \Gamma' \setminus \Gamma$ denote the set of element boundaries on the interior of domain Ω. The partition Ω' satisfies the following closure property:

$$\bar{\Omega} = \text{closure}\left(\Omega'\right) \tag{25}$$

Now, consider a multiscale overlapping decomposition of the displacement field into coarse and fine scales:

$$\boldsymbol{u} = \underbrace{\bar{\boldsymbol{u}}}_{\text{coarse scale}} + \underbrace{\boldsymbol{u}'}_{\text{fine scale}} \tag{26}$$

The coarse scales represent the component of the solution resolved by a given mesh, while the fine scales can be viewed as the relative component of error between the coarse scale and the exact solution. A similar decomposition is assumed for the weighting functions:

$$\boldsymbol{w} = \underbrace{\bar{\boldsymbol{w}}}_{\text{coarse scale}} + \underbrace{\boldsymbol{w}'}_{\text{fine scale}} \tag{27}$$

We define the appropriate spaces for $\bar{\boldsymbol{u}}$ and $\boldsymbol{u}'$ as:

$$\bar{\mathcal{S}} = \left\{ \bar{\boldsymbol{u}} \middle| \bar{\boldsymbol{u}} \in C^0(\Omega) \cap \mathcal{S},\ \bar{\boldsymbol{u}}|_{\Omega^e} \in \mathcal{P}^k\left(\Omega^e\right) \text{ for } e = 1, ..., n_{umel} \right\} \tag{28}$$

$$\mathcal{S}' = \left\{ \boldsymbol{u}' \middle| \boldsymbol{u}' \in \left(H^1(\Omega)\right)^{n_{sd}},\ \boldsymbol{u}' = \boldsymbol{0} \text{ on } \Gamma_g \right\} \tag{29}$$

where $\mathcal{P}^k\left(\Omega^e\right)$ denotes the set of complete polynomials of order k spanning Ω^e. The functions assumed for $\bar{\boldsymbol{u}}$ are permitted to be non-smooth across element boundaries, which will be an important consideration during the subsequent modification of the weak form (21) and (22). Additionally, for completeness, the functions representing $\boldsymbol{u}'$ are not assumed to vanish on element boundaries. While such an assumption is typical for stabilized variational multiscale methods, this generality will be important for error estimation. The relaxation that $\boldsymbol{u}' \neq \boldsymbol{0}$ on Γ is the major difference between the developments presented thus far and the method presented in [41].

Similarly, the space for the displacement weighting functions is defined as:

$$\bar{\mathcal{V}} = \left\{ \bar{\boldsymbol{w}} \middle| \bar{\boldsymbol{w}} \in C^0(\Omega) \cap \mathcal{V},\ \bar{\boldsymbol{w}}|_{\Omega^e} \in \mathcal{P}^k\left(\Omega^e\right) \text{ for } e = 1, ..., n_{umel} \right\} \tag{30}$$

$$\mathcal{V}' = \mathcal{S}' \tag{31}$$

To ensure a unique decomposition of $\boldsymbol{u}$ given in (26) these spaces are required to be linearly independent, i.e. $\mathcal{S} = \bar{\mathcal{S}} \oplus \mathcal{S}'$ and $\mathcal{V} = \bar{\mathcal{V}} \oplus \mathcal{V}'$.

Furthermore, the strain tensor $\boldsymbol{\varepsilon}$ is assumed to decompose into coarse- and fine-scale components due to its linearity with respect to displacements. To simplify notation, the coarse- and fine-scale components will be represented by $\bar{\boldsymbol{\varepsilon}}$ and $\boldsymbol{\varepsilon}'$, respectively:

$$\bar{\boldsymbol{\varepsilon}} = \boldsymbol{\varepsilon}(\bar{\boldsymbol{u}}) = \tfrac{1}{2}\left(\nabla\bar{\boldsymbol{u}} + \left(\nabla\bar{\boldsymbol{u}}\right)^T\right) \tag{32}$$

$$\boldsymbol{\varepsilon}' = \boldsymbol{\varepsilon}(\boldsymbol{u}') = \tfrac{1}{2}\left(\nabla\boldsymbol{u}' + \left(\nabla\boldsymbol{u}'\right)^T\right) \tag{33}$$

3.2.2 Multiscale Variational Problem

We substitute the decomposed trial solutions and weighting functions into the weak form (21) and (22):

$$\int_{\Omega'} \nabla\left(\bar{\boldsymbol{w}} + \boldsymbol{w}'\right) : \left[p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\bar{\boldsymbol{u}} + \boldsymbol{u}')\right] \mathrm{d}\Omega = \int_{\Omega'} \left(\bar{\boldsymbol{w}} + \boldsymbol{w}'\right) \cdot \boldsymbol{b}\, \mathrm{d}\Omega + \int_{\Gamma_h} \left(\bar{\boldsymbol{w}} + \boldsymbol{w}'\right) \cdot \boldsymbol{h}\, \mathrm{d}\Gamma \tag{34}$$

$$\int_{\Omega'} q\left(\nabla \cdot \left(\bar{\boldsymbol{u}} + \boldsymbol{u}'\right) - p/\lambda\right) \mathrm{d}\Omega = 0 \tag{35}$$

Employing the linearity of the above equations with respect to the displacement weighting function, (34) and (35) can be separated into a coarse-scale problem and fine-scale problem:

Coarse-Scale Problem $\mathcal{C}$

$$\int_{\Omega'} \nabla\bar{\boldsymbol{w}} : \left[p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\bar{\boldsymbol{u}} + \boldsymbol{u}')\right] \mathrm{d}\Omega = \int_{\Omega'} \bar{\boldsymbol{w}} \cdot \boldsymbol{b}\, \mathrm{d}\Omega + \int_{\Gamma_h} \bar{\boldsymbol{w}} \cdot \boldsymbol{h}\, \mathrm{d}\Gamma \tag{36}$$

$$\int_{\Omega'} q\left(\nabla \cdot \left(\bar{\boldsymbol{u}} + \boldsymbol{u}'\right) - p/\lambda\right) \mathrm{d}\Omega = 0 \tag{37}$$

Fine-Scale Problem $\mathcal{F}$

$$\int_{\Omega'} \nabla\boldsymbol{w}' : \left[p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}\left(\bar{\boldsymbol{u}} + \boldsymbol{u}'\right)\right] \mathrm{d}\Omega = \int_{\Omega'} \boldsymbol{w}' \cdot \boldsymbol{b}\, \mathrm{d}\Omega + \int_{\Gamma_h} \boldsymbol{w}' \cdot \boldsymbol{h}\, \mathrm{d}\Gamma \tag{38}$$

We now focus on the fine-scale equation $\mathcal{F}$. This equation is infinite dimensional since no particular form has been assumed for the fine-scale functions. Our goal will be to analyze (38) and

extract a generalized representation for the fine scales, which will serve as a reference point for developing a stabilized formulation and subsequently the error estimators.

3.2.3 Analysis of Fine-Scale Problem

The coarse component of displacement $\bar{\boldsymbol{u}}$ is permitted to be non-smooth across element boundaries; therefore, its derivatives may experience discontinuities. The appropriate operator to express such discontinuities is the jump operator, denoted by $[\![\bullet]\!]$.

Using the strain field as an example, the jump operator acts on a tensor quantity and produces a vector output as follows:

$$\begin{aligned} [\![\varepsilon]\!] &= \varepsilon^+ \boldsymbol{n}^+ + \varepsilon^- \boldsymbol{n}^- \\ &= \varepsilon^+ \boldsymbol{n}^+ - \varepsilon^- \boldsymbol{n}^+ \\ &= \left(\varepsilon^+ - \varepsilon^-\right) \boldsymbol{n} \end{aligned} \tag{39}$$

where $\boldsymbol{n} = \boldsymbol{n}^+ = -\boldsymbol{n}^-$ is an outward unit normal on the element boundary, and the ± superscript designates the element from which the indicated quantity is derived, as shown in Figure 4. Observe that (39) is invariant under a reassignment of the ± designations, and therefore represents a unique quantity.

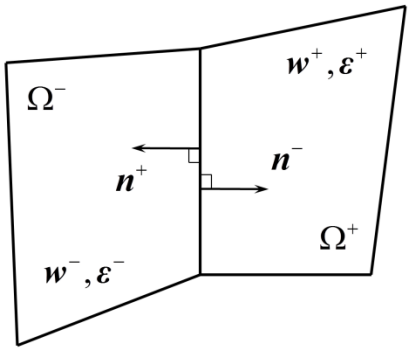


Figure 4: Depiction of unit outward normals on an element interface

Returning to the fine-scale problem $\mathcal{F}$ and using the expressions for the components of strain (32) and (33), those terms depending only upon the fine-scale quantities can be isolated from the coarse-scale quantities:

$$\int_{\Omega'} \nabla \boldsymbol{w}' : \left(2\mu\boldsymbol{\varepsilon}'\right) \mathrm{d}\Omega = -\int_{\Omega'} \nabla \boldsymbol{w}' : \left[p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}}\right] \mathrm{d}\Omega + \int_{\Omega'} \boldsymbol{w}' \cdot \boldsymbol{b} \, \mathrm{d}\Omega + \int_{\Gamma_h} \boldsymbol{w}' \cdot \boldsymbol{h} \, \mathrm{d}\Gamma \tag{40}$$

Integration by parts will now be applied to the first term on the right-hand side of this equation. Because the fine-scale functions do not vanish on Γ', this operation will give rise to boundary integrals; the result is:

$$-\int_{\Omega'} \nabla \boldsymbol{w}' : \left[p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}}\right] \mathrm{d}\Omega = \int_{\Omega'} \boldsymbol{w}' \cdot \left[\nabla p + \nabla \cdot \left(2\mu\bar{\boldsymbol{\varepsilon}}\right)\right] \mathrm{d}\Omega - \int_{\Gamma'} \boldsymbol{w}' \cdot \left[\!\left[p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}} \right]\!\right] \mathrm{d}\Gamma \tag{41}$$

Remark: Although the pressure field p is continuous in the present formulation, we point out that the jump term appearing in (41) arises in a consistent fashion from integration by parts and would accommodate more general formulations involving discontinuous approximations for p.

Recalling from (29) that $\boldsymbol{u}' = \boldsymbol{0}$ on Γ_g, we substitute (41) into $\mathcal{F}$ to obtain the following result:

$$\begin{aligned} \int_{\Omega'} \nabla \boldsymbol{w}' : \left(2\mu\boldsymbol{\varepsilon}'\right) \mathrm{d}\Omega = & \int_{\Omega'} \boldsymbol{w}' \cdot \left[\nabla p + \nabla \cdot \left(2\mu\bar{\boldsymbol{\varepsilon}}\right)\right] \mathrm{d}\Omega + \int_{\Omega'} \boldsymbol{w}' \cdot \boldsymbol{b} \, \mathrm{d}\Omega \\ & + \int_{\Gamma_h} \boldsymbol{w}' \cdot \left[\boldsymbol{h} - \left(p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}}\right)\boldsymbol{n}\right] \mathrm{d}\Gamma \\ & - \int_{\Gamma_{\mathrm{int}}} \boldsymbol{w}' \cdot \left[\!\left[p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}} \right]\!\right] \mathrm{d}\Gamma \end{aligned} \tag{42}$$

We now make the important observation that the right-hand side of (42) is entirely a function of the residual of the Euler-Lagrange equations (14) with respect to the coarse-scale displacement and boundary residuals representing the satisfaction of the traction boundary condition (17) and

point-wise continuity of the stress field across Γ'. To clarify, we introduce the following definitions:

$$\boldsymbol{r}_{\Omega'} = \nabla p + \nabla \cdot \left(2\mu\bar{\boldsymbol{\varepsilon}}\right) + \boldsymbol{b} \tag{43}$$

$$\boldsymbol{r}_{\Gamma_h} = \boldsymbol{h} - \left(p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}}\right)\boldsymbol{n} \tag{44}$$

$$\boldsymbol{r}_{\Gamma_{\text{int}}} = -\left[\!\left[p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}} \right]\!\right] \tag{45}$$

where $\boldsymbol{r}_{\Omega'}$ is the residual of the Euler-Lagrange equations over the sum of element interiors, $\boldsymbol{r}_{\Gamma_h}$ is the residual of the Neumann boundary conditions on Γ_h, and $\boldsymbol{r}_{\Gamma_{\text{int}}}$ is the residual associated with the inter-element continuity of the flux terms. Substituting these definitions into (42) gives:

Fine-Scale Equation $\mathcal{F}_r$

$$\int_{\Omega'} \nabla \boldsymbol{w}' : \left(2\mu\boldsymbol{\varepsilon}'\right) \mathrm{d}\Omega = \int_{\Omega'} \boldsymbol{w}' \cdot \boldsymbol{r}_{\Omega'} \, \mathrm{d}\Omega + \int_{\Gamma_h} \boldsymbol{w}' \cdot \boldsymbol{r}_{\Gamma_h} \, \mathrm{d}\Gamma + \int_{\Gamma_{\text{int}}} \boldsymbol{w}' \cdot \boldsymbol{r}_{\Gamma_{\text{int}}} \, \mathrm{d}\Gamma \tag{46}$$

This compact form represents a paradigm for both the construction of a stabilized formulation and of error estimators for the coarse-scale quantities. Since all terms depending on the fine-scale trial displacement functions have been isolated on the left-hand side of (46), we can see that the fine scales are in essence driven by the residuals of the coarse-scale variables. Thus, the fine scales vanish exactly under the conditions expected: when the coarse scale exactly satisfies the governing equations and the residuals identically vanish everywhere in Ω. This fact is central to the consistency of the resultant stabilized formulation in Section 3.3 and to the validity of the error estimators in Section 3.4.

3.3 Stabilized Mixed Formulation

We apply the Residual-Free Bubble (RFB) method [4,6,8] to the fine-scale equation $\mathcal{F}_r$ to determine an analytical expression for $\boldsymbol{u}'$. This is accomplished by making simplifying assumptions on the functional form of the fine scales. This functional form is then substituted into (46) to derive an expression for $\boldsymbol{u}'$. Substituting this expression into the coarse-scale problem $\mathcal{C}$ removes the explicit appearance of $\boldsymbol{u}'$ as an independent field in $\mathcal{C}$ while implicitly accounting for the fine-scale effects via additional stabilizing terms that result in a modified coarse-scale problem.

3.3.1 Derivation of the Fine-Scale Model

We now make some simplifying assumptions on the fine scales, namely:

$$\boldsymbol{u}' = \boldsymbol{0} \text{ on } \Gamma', \qquad \boldsymbol{w}' = \boldsymbol{0} \text{ on } \Gamma' \tag{47 a,b}$$

One consequence of this assumption is that second and third terms on the right-hand side of (46) vanish identically, which substantially simplifies the equation. Since the remaining terms are integrals over element interiors Ω', (46) can be evaluated as a sum of integrals in an element-by-element fashion:

$$\int_{\Omega^e} \nabla \boldsymbol{w}' : \left(2\mu\boldsymbol{\varepsilon}'\right) \mathrm{d}\Omega = \int_{\Omega^e} \boldsymbol{w}' \cdot \boldsymbol{r}_{\Omega'} \, \mathrm{d}\Omega \qquad \text{for each } e = 1, \ldots, n_{umel} \tag{48}$$

This equation can now be solved independently within each element of the mesh. While a multi-dimensional basis could be used to represent the fine scales, a single basis function typically provides a sufficient approximation for the purpose of stabilization. Therefore, in each element we represent the fine scales by the following expressions:

$$\boldsymbol{u}'\big|_{\Omega^e} = b^e\left(\boldsymbol{\xi}\right)\boldsymbol{\beta} \qquad \rightarrow \qquad u_i'\big|_{\Omega^e} = b^e\left(\boldsymbol{\xi}\right)\beta_i \quad \text{on } \Omega^e \tag{49}$$

$$\boldsymbol{w}'\big|_{\Omega^e} = b^e\left(\boldsymbol{\xi}\right)\boldsymbol{\gamma} \qquad \rightarrow \qquad w_i'\big|_{\Omega^e} = b^e\left(\boldsymbol{\xi}\right)\gamma_i \quad \text{on } \Omega^e \tag{50}$$

where $b^e\left(\boldsymbol{\xi}\right)$ denotes the bubble shape function over element domain Ω^e, $i = 1,...,n_{\text{sd}}$, and $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ represent the scaling coefficients for the fine-scale trial solutions and weighting functions, respectively.

Remark: In general, a bubble function is any interpolation or basis function that is zero on the entire boundary of an element. These functions are chosen to satisfy the characteristics of $\boldsymbol{u}'$ as specified by (29) and (47 a).

Substituting these forms of $\boldsymbol{u}'$ and $\boldsymbol{w}'$ into the integral on the left-hand side of (48), we can derive an expression that is valid over element interiors:

$$\begin{aligned} \nabla\boldsymbol{w}' : \left(2\mu\boldsymbol{\varepsilon}'\right) &= \boldsymbol{w}' : \left(\mu\nabla\boldsymbol{u}' + \mu\left(\nabla\boldsymbol{u}'\right)^T\right) \\ &= \left(\boldsymbol{\gamma}\otimes\nabla b^e\right) : \left(\mu\boldsymbol{\beta}\otimes\nabla b^e + \mu\nabla b^e\otimes\boldsymbol{\beta}\right) \\ &= \boldsymbol{\gamma}\cdot\left[\left(\nabla b^e\cdot\nabla b^e\right)\mu\boldsymbol{I} + \mu\left(\nabla b^e\otimes\nabla b^e\right)\right]\boldsymbol{\beta} \end{aligned} \tag{51}$$

Using this expression with (49) and (50), we may rewrite (48) as:

$$\boldsymbol{\gamma}\cdot\left[\int_{\Omega^e}\left(\nabla b^e\cdot\nabla b^e\right)\mu\boldsymbol{I} + \mu\left(\nabla b^e\otimes\nabla b^e\right)\mathrm{d}\Omega\right]\boldsymbol{\beta} = \boldsymbol{\gamma}\cdot\int_{\Omega^e} b^e\boldsymbol{r}_{\Omega'}\,\mathrm{d}\Omega \tag{52}$$

where the vectors of constant coefficients have been factored out of the integrals. For (52) to hold for any arbitrary weighting function, we have the following equation written in matrix form:

$$\boldsymbol{\beta} = \boldsymbol{K}^{-1}\boldsymbol{R} \tag{53}$$

where $\boldsymbol{K}$ and $\boldsymbol{R}$ are defined as follows:

$$\boldsymbol{K} = \int_{\Omega^e} \left(\nabla b^e \cdot \nabla b^e\right)\mu\boldsymbol{I} + \mu\left(\nabla b^e \otimes \nabla b^e\right) \mathrm{d}\Omega \tag{54}$$

$$\boldsymbol{R} = \int_{\Omega^e} b^e \boldsymbol{r}_{\Omega'} \, \mathrm{d}\Omega \tag{55}$$

Substituting (53) into (49) gives an analytical expression for $\boldsymbol{u}'$:

$$\boldsymbol{u}'\big|_{\Omega^e} = b^e(\xi)\boldsymbol{K}^{-1}\boldsymbol{R} \tag{56}$$

Based on our assumptions, this expression is valid over element interiors and presents a relation between the fine scales $\boldsymbol{u}'$ and the coarse-scale residual $\boldsymbol{r}_{\Omega'}$.

Employing the mean-value theorem, we can simplify (56) by taking the residual $\boldsymbol{r}_{\Omega'}$ out of the integral expression in (55):

$$\boldsymbol{R} \approx \left(\int_{\Omega^e} b^e \, \mathrm{d}\Omega\right)\boldsymbol{r}_{\Omega'} \tag{57}$$

This leads to a succinct expression for the fine-scale displacement $\boldsymbol{u}'$ over the sum of element interiors:

$$\boldsymbol{u}'\big|_{\Omega^e} = \boldsymbol{\tau}\boldsymbol{r}_{\Omega'} \tag{58}$$

where $\boldsymbol{\tau}$ is a second-order stabilization tensor with the following form:

$$\boldsymbol{\tau} = \left(b^e \int_{\Omega^e} b^e \, \mathrm{d}\Omega\right)\left[\left(\int_{\Omega^e} \mu \nabla b^e \cdot \nabla b^e \, \mathrm{d}\Omega\right)\boldsymbol{I} + \int_{\Omega^e} \mu \nabla b^e \otimes \nabla b^e \, \mathrm{d}\Omega\right]^{-1} \tag{59}$$

Therefore, under our assumptions, we conclude that $\boldsymbol{u}'$ is a function of the mechanical material parameters, the bubble functions b^e, and the residual of the equilibrium equation $\boldsymbol{r}_{\Omega'}$.

Remark: From a practical standpoint, on a sufficiently refined mesh, $\boldsymbol{r}_{\Omega'}$ converges to a constant value on the interior of each element.

3.3.2 Embedding Fine-Scale Model in the Coarse-Scale Formulation

We reconsider the terms involving $\boldsymbol{u}'$ in (36) and (37) and apply integration by parts wherever necessary to remove derivatives. Beginning with (36), we employ the decomposition of strain given by (32) and (33) to separate the term on the left-hand side and obtain:

$$\int_{\Omega'} \nabla \bar{\boldsymbol{w}} : \left[p\boldsymbol{I} + 2\mu\bar{\boldsymbol{\varepsilon}}\right] \mathrm{d}\Omega + \int_{\Omega'} 2\mu\left(\nabla \bar{\boldsymbol{w}} : \boldsymbol{\varepsilon}'\right) \mathrm{d}\Omega = \int_{\Omega'} \bar{\boldsymbol{w}} \cdot \boldsymbol{b} \, \mathrm{d}\Omega + \int_{\Gamma_h} \bar{\boldsymbol{w}} \cdot \boldsymbol{h} \, \mathrm{d}\Gamma \tag{60}$$

Focusing on the second term in (60), recalling that $\boldsymbol{u}' = \boldsymbol{0}$ on Γ', employing the identity $\nabla \boldsymbol{a} : \boldsymbol{\varepsilon}(\boldsymbol{b}) = \boldsymbol{\varepsilon}(\boldsymbol{a}) : \nabla \boldsymbol{b}$ which holds for all vector fields $\boldsymbol{a}$ and $\boldsymbol{b}$, and integrating by parts we obtain:

$$\int_{\Omega'} 2\mu\left(\nabla \bar{\boldsymbol{w}} : \boldsymbol{\varepsilon}'\right) \mathrm{d}\Omega = -\int_{\Omega'} 2\mu\left[\nabla \cdot \boldsymbol{\varepsilon}(\bar{\boldsymbol{w}})\right] \cdot \boldsymbol{u}' \, \mathrm{d}\Omega \tag{61}$$

Substituting the expression for $\boldsymbol{u}'$ (58) into (61) gives:

$$-\int_{\Omega'} 2\mu\left[\nabla \cdot \boldsymbol{\varepsilon}(\bar{\boldsymbol{w}})\right] \cdot \boldsymbol{u}' \, \mathrm{d}\Omega = -\int_{\Omega'} 2\mu\left[\nabla \cdot \boldsymbol{\varepsilon}(\bar{\boldsymbol{w}})\right] \cdot \boldsymbol{\tau}\boldsymbol{r}_{\Omega'} \, \mathrm{d}\Omega \tag{62}$$

Now returning to (37), employ the decomposition of $\boldsymbol{u}$ given by (26) to rewrite this equation as:

$$\int_{\Omega'} q\left(\nabla\cdot\bar{\boldsymbol{u}}-p/\lambda\right)\mathrm{d}\Omega+\int_{\Omega'} q\nabla\cdot\boldsymbol{u}'\ \mathrm{d}\Omega=0 \tag{63}$$

Considering the second term, we may integrate by parts and substitute (58) to obtain:

$$\int_{\Omega'} q\nabla\cdot\boldsymbol{u}'\,\mathrm{d}\Omega=-\int_{\Omega'}\nabla q\cdot\boldsymbol{u}'\,\mathrm{d}\Omega=-\int_{\Omega'}\nabla q\cdot\boldsymbol{\tau}\boldsymbol{r}_{\Omega'}\,\mathrm{d}\Omega \tag{64}$$

Inserting (62) into (60) and (64) into (63), the coarse-scale problem $\mathcal{C}$ can be written in the following modified form:

Modified Coarse-Scale Problem $\mathcal{M}$

$$\int_{\Omega'}\nabla\bar{\boldsymbol{w}}:\left[p\boldsymbol{I}+2\mu\bar{\boldsymbol{\varepsilon}}\right]\mathrm{d}\Omega-\int_{\Omega'}2\mu\left[\nabla\cdot\boldsymbol{\varepsilon}(\bar{\boldsymbol{w}})\right]\cdot\boldsymbol{\tau}\boldsymbol{r}_{\Omega'}\,\mathrm{d}\Omega=\int_{\Omega'}\bar{\boldsymbol{w}}\cdot\boldsymbol{b}\,\mathrm{d}\Omega+\int_{\Gamma_h}\bar{\boldsymbol{w}}\cdot\boldsymbol{h}\,\mathrm{d}\Gamma \tag{65}$$

$$\int_{\Omega'} q\left(\nabla\cdot\bar{\boldsymbol{u}}-p/\lambda\right)\mathrm{d}\Omega-\int_{\Omega'}\nabla q\cdot\boldsymbol{\tau}\boldsymbol{r}_{\Omega'}\,\mathrm{d}\Omega=0 \tag{66}$$

3.3.3 The Stabilized Form

We combine (65) and (66) to obtain a single expression that represents the stabilized form for incompressible elasticity. Since all fine-scale terms have been explicitly eliminated from the equations, the superimposed bars on the coarse-scale terms will be dropped for simplicity. To accommodate the additional stabilization terms, the appropriate space of functions for the pressure field now becomes:

$$\mathcal{Q}=\left\{p\middle|\,p\in\mathcal{P},\ p\big|_{\Omega^e}\in H^1\left(\Omega^e\right)\ \text{for}\ e=1,2,...,n_{umel}\right\} \tag{67}$$

We rearrange the terms and introduce the expression for $\boldsymbol{r}_{\Omega'}$ from (43) to obtain the stabilized form that can be expressed as: Find $\boldsymbol{u}\in\bar{\mathcal{V}},p\in\mathcal{Q}$ such that for all $\boldsymbol{w}\in\bar{\mathcal{V}},q\in\mathcal{Q}$:

$$\int_{\Omega'} \nabla \boldsymbol{w} : \left[p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) \right] \mathrm{d}\Omega + \int_{\Omega'} q\left(\nabla \cdot \boldsymbol{u} - p/\lambda \right) \mathrm{d}\Omega \tag{68}$$
$$- \int_{\Omega'} \left[\nabla q + 2\mu \nabla \cdot \boldsymbol{\varepsilon}(\boldsymbol{w}) \right] \cdot \boldsymbol{\tau} \left[\nabla p + 2\mu \nabla \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}) \right] \mathrm{d}\Omega$$
$$= \int_{\Omega'} \boldsymbol{w} \cdot \boldsymbol{b} \, \mathrm{d}\Omega + \int_{\Gamma_h} \boldsymbol{w} \cdot \boldsymbol{h} \, \mathrm{d}\Gamma + \int_{\Omega'} \left[\nabla q + 2\mu \nabla \cdot \boldsymbol{\varepsilon}(\boldsymbol{w}) \right] \cdot \boldsymbol{\tau} \boldsymbol{b} \, \mathrm{d}\Omega$$

Remark: The last terms on the left-hand and right-hand side have appeared due to the assumption of fine scales in the problem. These terms account for the subgrid scales that are unaccounted for by the standard Galerkin methods on a given discretization. These terms provide improved stability to the formulation.

Remark: Since the stabilization terms are residual-based, this method is consistent; when the coarse scales represent the total solution, the residual of the Euler-Lagrange equations vanishes identically, and we recover the standard Galerkin form (21) and (22).

Remark: We emphasize that the structure of the stabilization tensor $\boldsymbol{\tau}$ was derived based on a variational principle and therefore is not an explicit function of the characteristic mesh parameter h or any other user-defined parameter, except for the choice of the element bubble function.

3.4 Residual based Error Estimation

In this section we describe a procedure for error estimation that emanates from the philosophy of the Variational Multiscale Method. As presented in (26), within the standard Galerkin framework the fine-scale displacement $\boldsymbol{u}'$ represents the component of the exact solution that is unaccounted for in the coarse scale $\bar{\boldsymbol{u}}$ on a given discretization, where $\bar{\boldsymbol{u}}(\boldsymbol{u}') \cong \boldsymbol{u}^h$ is the typical finite element solution. If we rearrange this expression and recall the definition of standard error, we see that the fine scales can represent the error between the coarse solution and the exact solution:

$$\boldsymbol{e} = \boldsymbol{u} - \underbrace{\bar{\boldsymbol{u}}(\boldsymbol{u}')}_{\boldsymbol{u}^h} \tag{69}$$

The coupling between the coarse- and fine-scale problems (36) – (38) suggests a strategy for computing a robust error estimate. A simple expression for $\boldsymbol{u}'$ was derived in Section 3.1. Because of the assumptions behind this expression, this approximation may not provide a sharp estimate of error. By relaxing these assumptions, however, we can return to the fine-scale problem to obtain a better representation of $\boldsymbol{u}'$. Specifically, once the finite element solutions $\boldsymbol{u}^h$ and p^h are obtained from the modified coarse-scale problem $\mathcal{M}$, everything in the fine-scale problem $\mathcal{F}_r$ is computable. Thus, a better approximation of $\boldsymbol{u}'$ can be obtained. Additionally, because the computed coarse scales are a function of the embedded fine scales, i.e. $\bar{\boldsymbol{u}} = \bar{\boldsymbol{u}}(\boldsymbol{u}')$, this improved value of $\boldsymbol{u}'$ can be used to obtain a better coarse-scale field. Thus, an estimate for the total error in the discrete solution is the sum of the improved fine-scale field and the difference between the improved and original coarse-scale fields.

Following this logic, we assume an additive decomposition of the error $\boldsymbol{e}$ into two components:

$$\boldsymbol{e} = \boldsymbol{e}_L + \boldsymbol{e}_G \tag{70}$$

The first component $\boldsymbol{e}_L$ is an element of the fine-scale space that can be obtained by solving (46) in some manner. Because solving this equation on a finer discretization over the entire domain is intractable, we will employ a technique to localize the problem over a series of subdomains; hence, this component is referred to as the local error. The second component $\boldsymbol{e}_G$ represents the global pollution error, which we approximate by solving a single global problem analogous to the

modified coarse-scale problem $\mathcal{M}$. Thus, the total estimated error can be viewed as the sum of a fine-scale local part and a coarse-scale global part.

Remark: Typical error estimation methods involving localization have been known to miss the pollution error, which represents the effect on error at one location by residuals further away in the domain [3,22,44]. Accounting for this global source of error provides a mechanism for computing reliable total estimates.

For the mixed formulation presented here, we have elected to use the L_2 norm and H^1 seminorm, which are defined for an arbitrary vector field $\boldsymbol{v}$ (e.g., $\boldsymbol{u}$ or p) as:

$$\|\boldsymbol{v}\|_{L_2(\omega)} = \sqrt{\int_\omega \boldsymbol{v}\cdot\boldsymbol{v}\,\mathrm{d}\Omega} \tag{71}$$

$$|\boldsymbol{v}|_{H^1(\omega)} = \sqrt{\int_\omega \nabla\boldsymbol{v}:\nabla\boldsymbol{v}\,\mathrm{d}\Omega} \tag{72}$$

where ω is the domain of integration, typically either a single element Ω^e or the entire domain Ω. In the latter case, we will abbreviate notation as follows: $\|\boldsymbol{v}\|_{L_2(\Omega)} = \|\boldsymbol{v}\|_{L_2}$ and $|\boldsymbol{v}|_{H^1(\Omega)} = |\boldsymbol{v}|_{H^1}$.

We are especially concerned with the following local error indicators η_L^e computed from the fine scale in each element and η_G^e that accounts for the pollution error in each element:

$$\eta_L^e = |\boldsymbol{e}_L|_{H^1(\Omega^e)}, \qquad \eta_G^e = |\boldsymbol{e}_G|_{H^1(\Omega^e)} \tag{73 a,b}$$

These elemental quantities can be aggregated into a single error indicator η over the entire domain:

$$\eta = \left\{ \sum_{e=1}^{n_{umel}} \left(\eta_L^e \right)^2 + \sum_{e=1}^{n_{umel}} \left(\eta_G^e \right)^2 \right\}^{\frac{1}{2}} \tag{74}$$

3.4.1 Local-Explicit a Posteriori Error Indicator

Recall the analytical form derived for $\boldsymbol{u}'$ given by (58) and (59) that is a function of the residual of the governing equations $\boldsymbol{r}_{\Omega'}$(43). Once the modified coarse-scale problem $\mathcal{M}$ has been solved numerically for $\boldsymbol{u}^h$ and p^h, this expression can be directly evaluated element-wise via a simple post-processing step and therefore can be considered as a local-explicit error indicator:

$$\boldsymbol{u}' = \boldsymbol{e}_L = \boldsymbol{\tau} \boldsymbol{r}_{\Omega'} \tag{75}$$

This formula has an analogy with traditional explicit residual-based error estimates [19,20,47]. These explicit residual-based estimates typically contain unknown constants that can be found through solving dual problems [14]. In the present case, the stabilization tensor $\boldsymbol{\tau}$ serves as an approximation of this constant, an approximation that was consistently derived from the governing equations. Thus, the VMS formulation comes equipped with an error indicator that does not require any additional mechanisms to evaluate beyond those already utilized in the solution process. The formula for the error indicator using the H^1 seminorm is:

$$\eta_L^e = \left| \boldsymbol{e}_L \right|_{H^1\left(\Omega^e\right)} \tag{76}$$

where,

$$\nabla \boldsymbol{e}_L = \left(\nabla \boldsymbol{\tau}\right) \boldsymbol{r}_{\Omega'} \tag{77}$$
$$= \nabla b^e \left(\int_{\Omega^e} b^e \, \mathrm{d}\Omega \right) \left[\left(\int_{\Omega^e} \mu \nabla b^e \cdot \nabla b^e \, \mathrm{d}\Omega \right) \boldsymbol{I} + \int_{\Omega^e} \mu \nabla b^e \otimes \nabla b^e \, \mathrm{d}\Omega \right]^{-1} \boldsymbol{r}_{\Omega'}$$

The local-explicit fine-scale error quantifies the local error in the finite element solution.

Remark: This method provides a simple procedure that is merely a post-processing step after the main solution phase.

Remark: The accuracy of this method depends upon the validity of the assumption that $\boldsymbol{u}' = \boldsymbol{0}$ on Γ', use of the mean value of $\boldsymbol{r}_{\Omega'}$ over element interiors, and the ability of the bubble function b^e to represent $\boldsymbol{u}'$.

3.4.2 Evaluation of Global Error

We now turn toward computing an approximation of the global error $\boldsymbol{e}_G$. As stated previously, the coarse-scale fields are a function of the embedded fine-scale fields. The total solution is represented as follows:

$$\boldsymbol{u}_{tot} = \boldsymbol{u}^h + \boldsymbol{e}_{G,\boldsymbol{u}} \tag{78}$$

$$p_{tot} = p^h + e_{G,p} \tag{79}$$

Thus, we return to the coarse-scale problem $\mathcal{C}$ in Section 3.2 and, using (78), substitute $\bar{\boldsymbol{u}} \cong \boldsymbol{u}_{tot}$ along with expressions (78) and (79) for $\boldsymbol{u}'$ and p, respectively. Moving the terms involving $\boldsymbol{e}_L$ to the right-hand side, we obtain:

$$\int_{\Omega'} \nabla \bar{\boldsymbol{w}} : \left[\left(e_{G,p} + p^h \right) \boldsymbol{I} + 2\mu \boldsymbol{\varepsilon} \left(\boldsymbol{e}_{G,\boldsymbol{u}} + \boldsymbol{u}^h + \boldsymbol{e}_L \right) \right] \mathrm{d}\Omega = \int_{\Omega'} \bar{\boldsymbol{w}} \cdot \boldsymbol{b} \, \mathrm{d}\Omega + \int_{\Gamma_h} \bar{\boldsymbol{w}} \cdot \boldsymbol{h} \, \mathrm{d}\Gamma - \int_{\Omega'} \nabla \bar{\boldsymbol{w}} : 2\mu \boldsymbol{\varepsilon} \left(\boldsymbol{e}_L \right) \mathrm{d}\Omega \tag{80}$$

$$\int_{\Omega'} q \left(\nabla \cdot \left(\boldsymbol{e}_{G,\boldsymbol{u}} + \boldsymbol{u}^h + \boldsymbol{e}_L \right) - \left(e_{G,p} + p^h \right) / \lambda \right) \mathrm{d}\Omega = - \int_{\Omega'} q \nabla \cdot \boldsymbol{e}_L \, \mathrm{d}\Omega \tag{81}$$

Proceeding as in Section 3.3, a stabilized form analogous to (68) can be derived. However, a key ingredient in simplifying the expression is the following: the system $\left(\boldsymbol{u}^h, \boldsymbol{e}_L, p^h \right)$ already solves (68) for all $\boldsymbol{w}^h \in \bar{\mathcal{V}}, q^h \in \mathcal{Q}$, which is derived from $\mathcal{C}$. Therefore, those terms drop out, and the final stabilized form for computing $\boldsymbol{e}_G$ is given as:

$$\begin{aligned} \int_{\Omega'} \nabla \boldsymbol{w} : \left[e_{G,p} \boldsymbol{I} + 2\mu \boldsymbol{\varepsilon} \left(\boldsymbol{e}_{G,\boldsymbol{u}} \right) \right] \mathrm{d}\Omega + \int_{\Omega'} q \left(\nabla \cdot \boldsymbol{e}_{G,\boldsymbol{u}} - e_{G,p} / \lambda \right) \mathrm{d}\Omega \\ - \int_{\Omega'} \left[\nabla q + 2\mu \nabla \cdot \boldsymbol{\varepsilon} \left(\boldsymbol{w} \right) \right] \cdot \boldsymbol{\tau} \left[\nabla e_{G,p} + 2\mu \nabla \cdot \boldsymbol{\varepsilon} \left(\boldsymbol{e}_{G,\boldsymbol{u}} \right) \right] \mathrm{d}\Omega \\ = - \int_{\Omega'} \nabla \boldsymbol{w} : 2\mu \boldsymbol{\varepsilon} \left(\boldsymbol{e}_L \right) \mathrm{d}\Omega - \int_{\Omega'} q \nabla \cdot \boldsymbol{e}_L \, \mathrm{d}\Omega \end{aligned} \tag{82}$$

Remark: Equation (82) is in fact a residual from of the coarse-scale problem $\mathcal{C}$ where certain terms in the out-of-balance force vector get dropped out. Consequently, $\boldsymbol{e}_{G,\boldsymbol{u}}$ and $e_{G,p}$ can be viewed as the second consistent iterates of the course solution. As a result, $\boldsymbol{u}^h$ and p^h are the dominant terms in (78) and (79).

3.4.3 Computation of Total Error

The representation of the fine-scale error combined with the derived coarse-scale components $\boldsymbol{e}_{G,\boldsymbol{u}}$ and $e_{G,p}$ will be termed as the implicit error estimator:

$$\boldsymbol{e}_I = \boldsymbol{e}_L + \boldsymbol{e}_{G,\boldsymbol{u}} \tag{83}$$

$$e_p = e_{G,p} \tag{84}$$

The error indicator within each element associated with this quantity is:

$$\eta_I^e = \left|\boldsymbol{e}_I\right|_{H^1\left(\Omega^e\right)} \tag{85}$$

Remark: The form of the left-hand side of (82) is exactly the same as from (68). Thus, the same stiffness matrix $\boldsymbol{K}$ used in the modified coarse-scale problem can be used to compute the global error. In particular, if $\boldsymbol{K}$ is factorized by a direct solver and retained, then computing $\left(\boldsymbol{e}_{G,\boldsymbol{u}}, e_{G,p}\right)$ involves only a backward-substitution with an updated right-hand side.

Remark: Because of the assumption that fine scales are embedded into the stabilized stiffness matrix $\boldsymbol{K}$, part of the direct fine scales are accounted for on the left-hand side of (82). However, this overlap is neglected, and we observe from numerical tests that this method provides a reasonable predictor of the size and distribution of error in the finite element solution.

The following observations summarize the key features of the various proposed error estimators:

1) The local-explicit error indicator $\boldsymbol{e}_L$ only contains local information about the error in the solution as a function of the residual on element interiors. However, these fine-scale errors do predict the relative distribution of error in strains and stresses quite well.
2) A level of sophistication is added to the explicit method by solving a global problem to obtain an estimate of the pollution error. This estimate predicts the coarse-scale trends in the error quite well as seen in numerical tests; however, it is not as sharp as the implicit error estimate in term of predicting the magnitude.

CHAPTER 4: COUPLED PDEs WITH NON-MATCHING PRIMAL FIELDS IN SOLID MECHANICS

4.1 Interface Operator for Displacement Form of Elasticity

We consider the problem of coupling linear elastic domains with nonmatching meshes and possibly dissimilar material properties, for which the bilinear form and boundary operators take the following form:

$$a_{\Omega^{(\alpha)}}^{(\alpha)}\left(\boldsymbol{w}^{(\alpha)},\boldsymbol{u}^{(\alpha)}\right)=\int_{\Omega^{(\alpha)}}\boldsymbol{\varepsilon}\left(\boldsymbol{w}^{(\alpha)}\right):\left[\mathbf{C}^{(\alpha)}\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(\alpha)}\right)\right]\mathrm{d}\Omega \tag{86}$$

$$A^{(\alpha)}\left(\boldsymbol{u}^{(\alpha)}\right)=A^{*(\alpha)}\left(\boldsymbol{u}^{(\alpha)}\right)=\boldsymbol{n}^{(\alpha)}\cdot\gamma_o\left[\mathbf{C}^{(\alpha)}\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(\alpha)}\right)\right] \tag{87}$$

$$B^{(\alpha)}\left(\boldsymbol{u}^{(\alpha)}\right)=\gamma_o\boldsymbol{u}^{(\alpha)} \tag{88}$$

where $\boldsymbol{u}^{(\alpha)}\in\left(H^1\left(\Omega^{(\alpha)}\right)\right)^{n_{sd}}$ is the displacement field restricted to region $\Omega^{(\alpha)}$, $\boldsymbol{\varepsilon}(\bullet)=\frac{1}{2}\left[\nabla(\bullet)+\left(\nabla(\bullet)\right)^T\right]$ is the symmetric gradient operator, and $\mathbf{C}^{(\alpha)}$ is a fourth-order symmetric positive definite tensor of material moduli. From (86), we identify the stress tensor as $\boldsymbol{\sigma}^{(\alpha)}=\mathbf{C}^{(\alpha)}\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(\alpha)}\right)$ and the strain tensor as $\boldsymbol{\varepsilon}^{(\alpha)}=\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(\alpha)}\right)$. Following along the lines of Section 2.2 in [53] and substituting these definitions into the general equations leads to the following expression for the fine scales:

$$B^{(\alpha)}\tilde{\boldsymbol{u}}^{(\alpha)}\big|_{\gamma_s}=\boldsymbol{\tau}_s^{(\alpha)}\left[(-1)^{\alpha-1}\boldsymbol{\lambda}-\boldsymbol{n}^{(\alpha)}\cdot\boldsymbol{\sigma}^{(\alpha)}\left(\hat{\boldsymbol{u}}^{(\alpha)}\right)\right] \tag{89}$$

where $\boldsymbol{\lambda} \in \left(L^2 \left(\Gamma_I \right) \right)^{n_{sd}}$ has the physical connotation of the interface traction derived from domain Ω_1, and the expression for $\boldsymbol{\tau}_\alpha^j$ can be simplified to give:

$$\boldsymbol{\tau}_s^{(\alpha)} = \sum_{k,l=1}^{n_{sd}} \left[\text{meas}\left(\gamma_s \right) \right]^{-1} \left(\int_{\gamma_s} b_s^{(\alpha)} \, \mathrm{d}\Gamma \right)^2 \left[a_{\omega_s^{(\alpha)}}^{(\alpha)} \left(b_s^{(\alpha)} \boldsymbol{e}_l^{(\alpha)}, b_s^{(\alpha)} \boldsymbol{e}_k^{(\alpha)} \right) \right]^{-1} \left(\boldsymbol{e}_k^j \otimes \boldsymbol{e}_l^j \right) \tag{90}$$

Proceeding along the lines of the derivation in Section 2.3, the displacement jump follows simply as $\left[\!\left[\boldsymbol{u} \right]\!\right] = \boldsymbol{u}^{(1)} - \boldsymbol{u}^{(2)}$, and we define the numerical flux according to (6) as follows:

$$\left\{ \boldsymbol{n} \cdot \left(\mathbf{C} \boldsymbol{\varepsilon} \left(\boldsymbol{u} \right) \right) \right\} = \boldsymbol{\delta}_s^{(1)} \left(\boldsymbol{n}^{(1)} \cdot \left(\mathbf{C}^{(1)} \boldsymbol{\varepsilon} \left(\boldsymbol{u}^{(1)} \right) \right) \right) - \boldsymbol{\delta}_s^{(2)} \left(\boldsymbol{n}^{(2)} \cdot \left(\mathbf{C}^{(2)} \boldsymbol{\varepsilon} \left(\boldsymbol{u}^{(2)} \right) \right) \right) \tag{91}$$

where the weighting tensors $\boldsymbol{\delta}_s^{(\alpha)}$ are defined as:

$$\boldsymbol{\delta}_s^{(\alpha)} = \boldsymbol{\tau}_s \boldsymbol{\tau}_s^{(\alpha)}, \quad \boldsymbol{\tau}_s = \left(\boldsymbol{\tau}_s^{(1)} + \boldsymbol{\tau}_s^{(2)} \right)^{-1} \tag{92}$$

Remark: While the stabilization tensors $\boldsymbol{\tau}_s^{(\alpha)}$ are usually diagonally-dominant for elements with acceptable aspect ratios, they are not diagonal for triangular or distorted quadrilateral meshes [6]. Thus, the numerical flux $\left\{ \bullet \right\}$ involves a general linear combination of the traction components from the adjoining regions. If desired, the off-diagonal terms of the matrices $\left[\boldsymbol{\tau}_s^{(\alpha)} \right]$ could be dropped in order to simplify the calculations.

Substituting these results into (36) and neglecting the traction jump term, we obtain the stabilized interface formulation for linear elasticity:

$$a^{(1)}_{\tilde{\Omega}^{(1)}}\left(\hat{\boldsymbol{w}}^{(1)},\hat{\boldsymbol{u}}^{(1)}\right)+a^{(2)}_{\tilde{\Omega}^{(2)}}\left(\hat{\boldsymbol{w}}^{(2)},\hat{\boldsymbol{u}}^{(2)}\right)-\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right],\left\{\boldsymbol{n}\cdot\left(\mathbf{C}\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{u}}\right)\right)\right\}\right\rangle_{\tilde{\Gamma}_I}-\left\langle\left\{\boldsymbol{n}\cdot\left(\mathbf{C}\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}\right)\right)\right\},\left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I} \tag{93}$$
$$+\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right],\boldsymbol{\tau}_s\left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I}=\left(\hat{\boldsymbol{w}}^{(1)},\boldsymbol{f}^{(1)}\right)_{\tilde{\Omega}^{(1)}}+\left(\hat{\boldsymbol{w}}^{(2)},\boldsymbol{f}^{(2)}\right)_{\tilde{\Omega}^{(2)}}$$

Up to the definition of the penalty term and the weighted numerical flux, this formulation is form-identical to the DG method proposed in [37].

4.1.1 Accommodation of Residual Stresses and Strains

Next, our aim is to extend to the interface formulation (93) to account for residual stresses of the following form:

$$\bar{\boldsymbol{\sigma}}^{(\alpha)}\left(\boldsymbol{u}^{(\alpha)}\right)=\mathbf{C}^{(\alpha)}\left[\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(\alpha)}\right)-T^{(\alpha)}\boldsymbol{c}^{(\alpha)}\right]+\boldsymbol{\sigma}_i^{(\alpha)}=\boldsymbol{\sigma}^{(\alpha)}\left(\boldsymbol{u}^{(\alpha)}\right)+\boldsymbol{\sigma}'^{(\alpha)} \tag{94}$$

where $T^{(\alpha)}$ is a relative temperature field, $\boldsymbol{c}^{(\alpha)}$ is a thermal strain tensor often taken to be $\boldsymbol{c}^{(\alpha)}=\eta^{(\alpha)}\boldsymbol{I}$ with $\eta^{(\alpha)}$ the coefficient of thermal expansion and $\boldsymbol{I}$ the second-order identity tensor, and $\boldsymbol{\sigma}_i^{(\alpha)}$ is an initial stress tensor. In order to include the effects of this modified stress tensor in (93), we introduce the following affine functional:

$$\bar{a}^{(\alpha)}_{\Omega^{(\alpha)}}\left(\boldsymbol{w}^{(\alpha)},\boldsymbol{u}^{(\alpha)}\right)=a^{(\alpha)}_{\Omega^{(\alpha)}}\left(\boldsymbol{w}^{(\alpha)},\boldsymbol{u}^{(\alpha)}\right)+l^{(\alpha)}_{\Omega^{(\alpha)}}\left(\boldsymbol{w}^{(\alpha)}\right) \tag{95}$$

$$l^{(\alpha)}_{\Omega^{(\alpha)}}\left(\boldsymbol{w}^{(\alpha)}\right)=\int_{\Omega^{(\alpha)}}\boldsymbol{\varepsilon}\left(\boldsymbol{w}^{(\alpha)}\right):\boldsymbol{\sigma}'^{(\alpha)}\,\mathrm{d}\Omega \tag{96}$$

The equations for $\tilde{\boldsymbol{u}}^{(\alpha)}$ and $\boldsymbol{\tau}_s^{(\alpha)}$ in Section 4.1 remain valid except that the stress tensor $\boldsymbol{\sigma}^{(\alpha)}$ in the expression for the fine scales (89) is replaced by $\bar{\boldsymbol{\sigma}}^{(\alpha)}$ from (94). A modification is also required in the numerical flux term to account for the boundary term resulting from integration by parts of the thermal term:

$$\left\{\boldsymbol{n}\cdot\bar{\boldsymbol{\sigma}}\left(\boldsymbol{u}\right)\right\}=\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}\left(\boldsymbol{u}\right)\right\}+\boldsymbol{\delta}_s^{(1)}\left(\boldsymbol{n}^{(1)}\cdot\boldsymbol{\sigma}'^{(1)}\right)-\boldsymbol{\delta}_s^{(1)}\left(\boldsymbol{n}^{(2)}\cdot\boldsymbol{\sigma}'^{(2)}\right)=\left\{\boldsymbol{n}\cdot\left(\mathbf{C}\boldsymbol{\varepsilon}\left(\boldsymbol{u}\right)\right)\right\}+\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}'\right\} \tag{97}$$

Substituting (94) and (97) into (93), we arrive at the stabilized interface formulation for linear elasticity with residual stresses:

$$\begin{aligned}&a_{\tilde{\Omega}^{(1)}}^{(1)}\left(\hat{\boldsymbol{w}}^{(1)},\hat{\boldsymbol{u}}^{(1)}\right)+a_{\tilde{\Omega}^{(2)}}^{(2)}\left(\hat{\boldsymbol{w}}^{(2)},\hat{\boldsymbol{u}}^{(2)}\right)-\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right],\left\{\boldsymbol{n}\cdot\left(\mathbf{C}\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{u}}\right)\right)\right\}\right\rangle_{\tilde{\Gamma}_I}-\left\langle\left\{\boldsymbol{n}\cdot\left(\mathbf{C}\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}\right)\right)\right\},\left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I}\\&+\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right],\varepsilon_s\left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I}=\left(\hat{\boldsymbol{w}}^{(1)},\boldsymbol{f}^{(1)}\right)_{\tilde{\Omega}^{(1)}}+\left(\hat{\boldsymbol{w}}^{(2)},\boldsymbol{f}^{(2)}\right)_{\tilde{\Omega}^{(2)}}-l_{\tilde{\Omega}^{(1)}}^{(1)}\left(\hat{\boldsymbol{w}}^{(1)}\right)-l_{\tilde{\Omega}^{(2)}}^{(2)}\left(\hat{\boldsymbol{w}}^{(2)}\right)+\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right],\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}'\right\}\right\rangle_{\tilde{\Gamma}_I}\end{aligned} \tag{98}$$

Remark: The inclusion of the nonstandard term $\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}'\right\}$ on the right-hand side could be easily overlooked in the implementation of the classical DG method for elasticity with thermal strains. However, this term arises consistently during the present derivations, and the importance of retaining this term is demonstrated in Chapter 7.

4.2 Interface Operator for Mixed Form of Elasticity

As an example involving a mixed field problem, we consider a displacement-pressure formulation for elasticity that is capable of modeling incompressible materials. The main enhancement beyond the formulation in Section 4.1 is that the stress tensor becomes a function of the kinematic pressure $p^{(\alpha)}:\Omega^{(\alpha)}\to\mathbb{R}$. Assuming isotropic behavior, the stress-strain constitutive equation and associated compatibility condition are stated as:

$$\boldsymbol{\sigma}^{(\alpha)}\left(\boldsymbol{u}^{(\alpha)},p^{(\alpha)}\right)=p^{(\alpha)}\boldsymbol{I}+2\mu^{(\alpha)}\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(\alpha)}\right)\qquad\text{in }\Omega^{(\alpha)} \tag{99}$$

$$\nabla\cdot\boldsymbol{u}^{(\alpha)}=p^{(\alpha)}/\lambda^{(\alpha)}\qquad\text{in }\Omega^{(\alpha)} \tag{100}$$

where $\lambda^{(\alpha)}$, $\mu^{(\alpha)}$ are the Lame parameters characterizing the material in each region (not to be confused with the Lagrange multipliers). Substituting (99) into (86) leads to a three-field formulation in terms of $\{\boldsymbol{u}, p, \boldsymbol{\lambda}\}$. In order to eliminate the interfacial Lagrange multipliers, we again follow the following steps. A multiscale decomposition is applied only to the displacement field; this simplifies subsequent derivations and has been demonstrated to be sufficient for stabilizing mixed-field problems [35,38,39]. Adopting the fine-scale modeling assumptions from Section 3.3.1, we arrive at the following expression for $\tilde{\boldsymbol{u}}^{(\alpha)}$ at the interface:

$$B^{(\alpha)}\tilde{\boldsymbol{u}}^{(\alpha)}\big|_{\gamma_s} = \boldsymbol{\tau}_s^{(\alpha)}\left[(-1)^{\alpha-1}\boldsymbol{\lambda} - \boldsymbol{n}^{(\alpha)}\cdot\boldsymbol{\sigma}^{(\alpha)}\left(\hat{\boldsymbol{u}}^{(\alpha)}, p^{(\alpha)}\right)\right] \tag{101}$$

where $\boldsymbol{\sigma}^{(\alpha)}\left(\boldsymbol{u}^{(\alpha)}, p^{(\alpha)}\right)$ is evaluated using (99) and the stabilization tensors $\boldsymbol{\tau}_s^{(\alpha)}$ are given by:

$$\boldsymbol{\tau}_s^{(\alpha)} = \left[\text{meas}\left(\gamma_s\right)\right]^{-1}\left(\int_{\gamma_s} b_s^{(\alpha)}\,\mathrm{d}\Gamma\right)^2\left[\left(\int_{\omega_s^{(\alpha)}}\mu\nabla b_s^{(\alpha)}\cdot\nabla b_s^{(\alpha)}\,\mathrm{d}\Omega\right)\boldsymbol{I} + \int_{\omega_s^{(\alpha)}}\mu\nabla b_s^{(\alpha)}\otimes\nabla b_s^{(\alpha)}\,\mathrm{d}\Omega\right]^{-} \tag{102}$$

Embedding the representation of the fine scales (101) into the associated coarse-scale problem and proceeding along the lines of Section 2 to solve for the multiplier field $\boldsymbol{\lambda}$, we arrive at the stabilized interface formulation for mixed elasticity:

$$\begin{aligned}
&\sum_{\alpha=1,2}\left[\left(\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}^{(\alpha)}\right), 2\mu^{(\alpha)}\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{u}}^{(\alpha)}\right)\right)_{\tilde{\Omega}^{(\alpha)}} + \left(\nabla\cdot\hat{\boldsymbol{w}}^{(\alpha)}, p^{(\alpha)}\right)_{\tilde{\Omega}^{(\alpha)}}\right] \\
&+\sum_{\alpha=1,2}\left[\left(q^{(\alpha)}, \nabla\cdot\hat{\boldsymbol{u}}^{(\alpha)}\right)_{\tilde{\Omega}^{(\alpha)}} - \left(q^{(\alpha)}, p^{(\alpha)}/\lambda\right)_{\tilde{\Omega}^{(\alpha)}}\right] \\
&-\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right], \left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}\left(\hat{\boldsymbol{u}}, p\right)\right\}\right\rangle_{\tilde{\Gamma}_I} - \left\langle\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}\left(\hat{\boldsymbol{w}}, q\right)\right\}, \left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I} + \left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right], \boldsymbol{\tau}_s\left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I} = \sum_{\alpha=1,2}\left(\hat{\boldsymbol{w}}^{(\alpha)}, \boldsymbol{f}^{(\alpha)}\right)_{\tilde{\Omega}^{(\alpha)}}
\end{aligned} \tag{103}$$

where the numerical flux terms are computed using (91) with the stress tensor $\boldsymbol{\sigma}^{(\alpha)} = \mathbf{C}^{(\alpha)}\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(\alpha)}\right)$ replaced by definition (99).

Although the interface formulation (103) accommodates nonconforming discretizations for the displacement and pressure fields along Γ_I, the discrete function spaces for these two fields must satisfy the Babuška–Brezzi condition associated with the domain interior terms for each region $\Omega^{(\alpha)}$ in order for (103) to be globally stable. To admit equal-order interpolations of the displacement and pressure fields, additional domain-based stabilization terms are incorporated into the weak form that can be derived by employing a Variational Multiscale approach to the displacement field on element interiors [38,39]. Although the fine-scale models employed for stabilizing the mixed displacement-pressure formulation are assumed to vanish on element boundaries and therefore do not contribute directly to the interface integrals, overlapping contributions from the edge and interior bubble functions would be expected on the interior of the elements adjoining the interface. However, we choose to neglect these coupling effects in our implementation for simplicity, which does not upset the consistency of the formulation. Thus, the final form of the proposed interface formulation for mixed elasticity that admits arbitrary interpolation combinations across the interface is as follows:

$$
\begin{aligned}
&\sum_{\alpha=1,2}\left[\left(\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}^{(\alpha)}\right),2\mu^{(\alpha)}\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{u}}^{(\alpha)}\right)\right)_{\tilde{\Omega}^{(\alpha)}}+\left(\nabla\cdot\hat{\boldsymbol{w}}^{(\alpha)},p^{(\alpha)}\right)_{\tilde{\Omega}^{(\alpha)}}\right] \\
&+\sum_{\alpha=1,2}\left[\left(q^{(\alpha)},\nabla\cdot\hat{\boldsymbol{u}}^{(\alpha)}\right)_{\tilde{\Omega}^{(\alpha)}}-\left(q^{(\alpha)},p^{(\alpha)}/\lambda\right)_{\tilde{\Omega}^{(\alpha)}}\right] \\
&-\sum_{\alpha=1,2}\left(\nabla q^{(\alpha)}+2\mu^{(\alpha)}\nabla\cdot\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}^{(\alpha)}\right),\boldsymbol{\tau}_{\Omega}\cdot\left[\nabla p^{(\alpha)}+2\mu^{(\alpha)}\nabla\cdot\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{u}}^{(\alpha)}\right)\right]\right)_{\tilde{\Omega}^{(\alpha)}} \\
&-\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right],\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}\left(\hat{\boldsymbol{u}},p\right)\right\}\right\rangle_{\tilde{\Gamma}_I}-\left\langle\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}\left(\hat{\boldsymbol{w}},q\right)\right\},\left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I}+\left\langle\left[\!\left[\hat{\boldsymbol{w}}\right]\!\right],\boldsymbol{\tau}_s\left[\!\left[\hat{\boldsymbol{u}}\right]\!\right]\right\rangle_{\tilde{\Gamma}_I} \\
&=\sum_{\alpha=1,2}\left(\hat{\boldsymbol{w}}^{(\alpha)},\boldsymbol{f}^{(\alpha)}\right)_{\tilde{\Omega}^{(\alpha)}}-\left(\nabla q^{(\alpha)}+2\mu^{(\alpha)}\nabla\cdot\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}^{(\alpha)}\right),\boldsymbol{\tau}_{\Omega}\cdot\boldsymbol{f}^{(\alpha)}\right)_{\tilde{\Omega}^{(\alpha)}}
\end{aligned}
\tag{104}
$$

The form of the interior stabilization tensor $\boldsymbol{\tau}_{\Omega}$ is adopted from [39]:

$$\boldsymbol{\tau}_{\Omega} = b_e \int_{\Omega_e} b_e \, \mathrm{d}\Gamma \left[\left(\int_{\Omega_e} \mu \nabla b_e \cdot \nabla b_e \, \mathrm{d}\Omega \right) \boldsymbol{I} + \int_{\Omega_e} \mu \nabla b_e \otimes \nabla b_e \, \mathrm{d}\Omega \right]^{-1} \tag{105}$$

where the bubble function b_e is supported on the interior of element Ω_e. We remark that the stabilized weak form (104) is form-equivalent to the DG method proposed in Section 4.7 of [38] except for the crucial distinction that the penalty parameter and weighting coefficients have been consistently derived according to (92) and (102).

4.3 Multi-PDE Model Problem: Combining Pure-Displacement and Mixed Elasticity

As discussed in the beginning of Section 2, the stabilized interface formulation is capable of coupling different physical governing equations across nonmatching interfaces. Therefore, consider as a motivating example a linear elastic domain Ω that is partitioned into two regions $\Omega^{(\alpha)}$, the first of which is modeled using the pure-displacement constitutive equation (86) and the second of which is modeled using mixed elasticity via the relations (99) and (100). Possible reasons for considering different constitutive models include the ability to simulate (i) composite materials in which one constituent is incompressible as well as (ii) localized incompressible plastic flow within an otherwise elastic domain. The obvious computational economy is that the calculation of the pressure field is avoided in the larger region. Combining the results of Sections 4.1 and 4.2, the composite interface formulation is as follows:

$$\begin{aligned}
&\left(\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}^{(1)}\right), \mathbf{C}^{(1)} \boldsymbol{\varepsilon}\left(\hat{\boldsymbol{u}}^{(1)}\right) \right)_{\tilde{\Omega}^{(1)}} + a^{(2)}_{\tilde{\Omega}^{(2)}}\left(\hat{\boldsymbol{w}}^{(2)}, q^{(2)}; \hat{\boldsymbol{u}}^{(2)}, p^{(2)} \right) \\
&- \left\langle \left[\!\left[\hat{\boldsymbol{w}} \right]\!\right], \left\{ \boldsymbol{n} \cdot \boldsymbol{\sigma}\left(\hat{\boldsymbol{u}}, p\right) \right\} \right\rangle_{\tilde{\Gamma}_I} - \left\langle \left\{ \boldsymbol{n} \cdot \boldsymbol{\sigma}\left(\hat{\boldsymbol{w}}, q\right) \right\}, \left[\!\left[\hat{\boldsymbol{u}} \right]\!\right] \right\rangle_{\tilde{\Gamma}_I} + \left\langle \left[\!\left[\hat{\boldsymbol{w}} \right]\!\right], \boldsymbol{\tau}_s \left[\!\left[\hat{\boldsymbol{u}} \right]\!\right] \right\rangle_{\tilde{\Gamma}_I} \\
&= \left(\hat{\boldsymbol{w}}^{(1)}, \boldsymbol{f}^{(1)} \right)_{\tilde{\Omega}^{(1)}} + \left(\hat{\boldsymbol{w}}^{(2)}, \boldsymbol{f}^{(2)} \right)_{\tilde{\Omega}^{(2)}} - \left(\nabla q^{(2)} + 2\mu^{(2)} \nabla \cdot \boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}^{(2)}\right), \boldsymbol{\tau}_{\Omega} \cdot \boldsymbol{f}^{(2)} \right)_{\tilde{\Omega}^{(2)}}
\end{aligned} \tag{106}$$

where the bilinear form $a^{(2)}_{\tilde{\Omega}^{(2)}}(\bullet,\bullet)$ contains all of the domain integral contributions from the left-hand side of the mixed elasticity formulation (104), and the numerical flux terms are evaluated through the following composite definition:

$$\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}(\boldsymbol{u},p)\right\} = \boldsymbol{\delta}_s^{(1)}\left(\boldsymbol{n}^{(1)}\cdot\left(\mathbf{C}^{(1)}\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{(1)}\right)\right)\right) - \boldsymbol{\delta}_s^{(2)}\left(\boldsymbol{n}^{(2)}\cdot\boldsymbol{\sigma}^{(2)}\left(\boldsymbol{u}^{(2)},p^{(2)}\right)\right) \tag{107}$$

Both the stress tensors $\boldsymbol{\sigma}^{(\alpha)}$ and the stabilization tensors $\boldsymbol{\tau}_s^{(\alpha)}$ required to obtain the values for $\boldsymbol{\delta}_s^{(\alpha)}$ and $\boldsymbol{\tau}_s$ are evaluated according to the appropriate expressions in Sections 4.1 and 4.2. Thus, the physics of both governing differential operators is consistently embedded in the definition of the numerical interface parameters $\boldsymbol{\delta}_s^{(\alpha)}$ and $\boldsymbol{\tau}_s$ to provide a robust coupling mechanism at the discrete interface. These definitions remove the ambiguity in postulating a Discontinuous Galerkin method for a coupled system of PDEs. The performance of this multi-PDE elasticity formulation is assessed through a benchmark study in Section 7.2

Remark: In the preceding developments, the fine scales are viewed as arising due to the nonconforming discrete representation of the interface as well as the numerical instabilities in the classical mixed primal-multiplier method. Thus, $\tilde{\boldsymbol{u}}$ is treated as variationally embedded within the coarse scales and governed by the same PDE. However, the fine scales could also be viewed as a vehicle for accommodating multiscale physical features at the interface. One example is the frictional contact of rough surfaces through asperity interactions, where the micro-features on the interface are orders of magnitude smaller than the macro-structure. A related approach employing multiscale constitutive models for asperity-interaction within the DG method is presented in [52].

CHAPTER 5: STABILIZATION AND INTERFACE COUPLING IN FLUID MECHANICS

5.1 Darcy Equation

Darcy equation models flow through porous medium wherein permeability is low and frictional force between the fluid and the porous medium is the dominant factor, i.e., flow through clays and other dense geologic materials.

5.1.1 Mixed Velocity-Pressure Formulation

Let $\Omega_G \subseteq \mathbb{R}^{n_{sd}}$ be an open bounded region with piecewise smooth boundary Γ. The number of space dimensions, n_{sd}, is equal to 2 or 3. The Darcy law and conservation of mass are given by the following equations:

$$\boldsymbol{v} = -\frac{\kappa}{\mu}(\nabla p + \frac{\rho}{g_c}\boldsymbol{g}) \qquad \text{in } \Omega \qquad \left(\text{Darcy Law}\right) \tag{108}$$

$$\operatorname{div} \boldsymbol{v} = \varphi \qquad \text{in } \Omega \qquad \left(\text{Conservation of Mass}\right) \tag{109}$$

$$\boldsymbol{v} \cdot \boldsymbol{n} = \psi \qquad \text{on } \Gamma \tag{110}$$

where $\boldsymbol{v}$ is the Darcy velocity vector, p is the pressure, $\boldsymbol{g}$ is the gravity vector, φ is the volumetric flow rate source or sink, ψ is the normal component of the velocity field on the boundary, $\mu > 0$ is the viscosity, $\kappa > 0$ is the permeability, $\rho > 0$ is the density, g_c is a conversion constant, and $\boldsymbol{n}$ is the unit outward normal vector to Γ. It is apparent from (109) and (110) that the prescribed data φ and ψ must satisfy the constraint $\int_\Omega \varphi \, \mathrm{d}\Omega = \int_\Gamma \psi \, \mathrm{d}\Gamma$.

5.1.2 The Standard Weak Form

Let,

$$\mathcal{V} = \boldsymbol{H}\left(\text{div}, \Omega\right) \stackrel{def}{=} \left\{ \boldsymbol{v} \mid \boldsymbol{v} \in \left(L_2\left(\Omega\right)\right)^{n_{sd}}, \text{div}\, \boldsymbol{v} \in L_2\left(\Omega\right), \text{trace}\left(\boldsymbol{v} \cdot \boldsymbol{n}\right) = \psi \in H^{-\frac{1}{2}}\left(\Gamma\right) \right\} \tag{111}$$

$$\mathcal{P} = L_2\left(\Omega\right) \backslash \mathbb{R} \stackrel{def}{=} \left\{ p \mid p \in L_2\left(\Omega\right), \int_\Omega p \, d\Omega = 0 \right\} \tag{112}$$

$$\mathcal{V}_0 = \boldsymbol{H}_0\left(\text{div}, \Omega\right) \stackrel{def}{=} \left\{ \boldsymbol{v} \mid \boldsymbol{v} \in \left(L_2\left(\Omega\right)\right)^{n_{sd}}, \text{div}\, \boldsymbol{v} \in L_2\left(\Omega\right), \text{trace}\left(\boldsymbol{v} \cdot \boldsymbol{n}\right) = 0 \right\} \tag{113}$$

Further elaboration on these spaces is available in [7]. It is assumed that $\kappa, \mu, \rho, g_c, \boldsymbol{g}, \varphi$ and ψ are given. Thus, the standard weak form of (108-110) is: Find $\boldsymbol{v} \in \mathcal{V}, p \in \mathcal{P}$, such that, for all $\boldsymbol{w} \in \mathcal{V}_0, q \in \mathcal{P}$,

$$\left(\boldsymbol{w}, \frac{\mu}{\kappa} \boldsymbol{v} \right) - \left(\text{div}\, \boldsymbol{w}, p \right) + \left(q, \text{div}\, \boldsymbol{v} \right) = -\left(\boldsymbol{w} \frac{\rho}{g_c} \boldsymbol{g} \right) + \left(q, \varphi \right) \tag{114}$$

where $\left(\cdot,\cdot\right)$ is the $L_2\left(\Omega\right)$ inner product. A unique solution to the weak form exists for sufficiently regular data. For simplicity, it is convenient to rewrite (114) as: Let $\mathcal{Y} = \mathcal{V} \times \mathcal{P}$, $\mathcal{Y}_0 = \mathcal{V}_0 \times \mathcal{P}$, $\mathbf{V} = \left\{ \boldsymbol{v}, p \right\}$ and $\mathbf{W} = \left\{ \boldsymbol{w}, q \right\}$. Find $\mathbf{V} \in \mathcal{Y}$, such that for all $\mathbf{W} \in \mathcal{Y}_0$,

$$B_D\left(\mathbf{W}, \mathbf{V}\right) = L_D\left(\mathbf{W}\right) \tag{115}$$

where,

$$B_D(\mathbf{W},\mathbf{V}) = \left(\boldsymbol{w},\ \frac{\mu}{\kappa}\boldsymbol{v} \right) - \left(\text{div}\ \boldsymbol{w}, p\right) + \left(q,\ \text{div}\ \boldsymbol{v}\right) \tag{116}$$

$$L_D(\mathbf{W}) = -\left(\boldsymbol{w}, \frac{\rho}{g_c}\boldsymbol{g} \right) + \left(q,\varphi\right) \tag{117}$$

Remark: The Galerkin finite element method is based on (115). Stability is achieved for only certain combinations of velocity and pressure interpolations [7,12,46].

5.1.3 The Stabilized Weak Form

Employing the Variational Multiscale framework proposed in [33], a stabilized form that is summarized as follows: Let

$$\mathcal{Q} = H^1(\Omega)\backslash\mathbb{R} \overset{def}{=} \left\{ q \mid q \in H^1(\Omega), \int_\Omega q\, d\Omega = 0 \right\} \tag{118}$$

$$\mathcal{Z} = \mathcal{V} \times \mathcal{Q}, \quad \mathcal{Z}_0 = \mathcal{V}_0 \times \mathcal{Q} \tag{119}$$

The stabilized weak form is: Find $\mathbf{V} \in \mathcal{Z}$, such that for all $\mathbf{W} \in \mathcal{Z}_0$,

$$B_D^{stab}(\mathbf{W},\mathbf{V}) = L_D^{stab}(\mathbf{W}) \tag{120}$$

where,

$$B_D^{stab}(\mathbf{W},\mathbf{V}) = B_D(\mathbf{W},\mathbf{V}) + \frac{1}{2}\left(\left(-\frac{\mu}{\kappa}\boldsymbol{w} + \nabla q \right), \frac{\kappa}{\mu}\left(\frac{\mu}{\kappa}\boldsymbol{v} + \nabla p \right) \right) \tag{121}$$

$$L_D^{stab}\left(\mathbf{W}\right)=L_D\left(\mathbf{W}\right)-\frac{1}{2}\left(\left(-\frac{\mu}{\kappa}\boldsymbol{w}+\nabla q\right),\frac{\kappa}{\mu}\left(\frac{\rho}{g_c}\boldsymbol{g}\right)\right) \tag{122}$$

5.1.4 Interface Operator for Darcy Equation

The launching point for the derivation in the context of Darcy flow is the standard Lagrange multiplier method for weakly enforcing continuity of the normal component of velocity at the interface, which is stated using the functionals from (115) and (116) as follows:

$$\sum_{\alpha=1,2} B_D^{(\alpha)}\left(\mathbf{W}^{(\alpha)},\mathbf{V}^{(\alpha)}\right)+\left\langle\left\{\boldsymbol{w}\cdot\boldsymbol{n}\right\},\lambda\right\rangle_{\Gamma_I}=\sum_{\alpha=1,2} L_D^{(\alpha)}\left(\mathbf{W}^{(\alpha)}\right) \tag{123}$$

$$\left\langle\mu,\left\{\boldsymbol{v}\cdot\boldsymbol{n}\right\}\right\rangle_{\Gamma_I}=0 \tag{124}$$

where $\lambda,\ \mu\in H^{-\frac{1}{2}}\left(\Gamma_I\right)$ are the Lagrange multipliers, $\left\{\boldsymbol{v}\cdot\boldsymbol{n}\right\}=\boldsymbol{v}^{(1)}\cdot\boldsymbol{n}-\boldsymbol{v}^{(2)}\cdot\boldsymbol{n}$ is the numerical flux, and the subscripts indicate the restriction of the associated fields and functional forms to the respective sub-domains $\Omega^{(\alpha)}$, and $\boldsymbol{n}$ is the unit outward normal vector to $\Omega^{(1)}$.

Now, the key idea is to derive an expression for the Lagrange multiplier field λ in terms of the primary fields $\boldsymbol{v}^{(1)}$ and $\boldsymbol{v}^{(2)}$ at the interface in order to remove the explicit appearance of λ in (123). This is accomplished by following the general framework described in [9] whereby a multiscale decomposition is applied to the velocity field locally at the interface. By applying modeling assumptions to the fine-scales $\boldsymbol{v}'^{(1)}$ and $\boldsymbol{v}'^{(2)}$ on either side of the interface, we arrive at analytical expressions for these scales in terms of the coarse-scales $\tilde{\boldsymbol{v}}^{(1)}$ and $\tilde{\boldsymbol{v}}^{(2)}$ and the multiplier λ. The reader is referred to [9] for a complete discussion of the modeling procedures. The major assumption is that the fine-scales are considered to be localized to the elements from regions $\Omega^{(\alpha)}$

that border the interface Γ_I. Specifically, let $\{\gamma_s\}_{s=1}^{n_{seg}}$ be a partition of Γ_I into a series of segments γ_s defined by adjacent pairs of elements $\Omega_{e_s}^{(\alpha)}$ such that $\partial\Omega_{e_s}^{(1)} \bigcap \partial\Omega_{e_s}^{(2)} = \gamma_s$ for each j. Throughout, we use the subscript α to denote the model region $\Omega^{(1)}$ and $\Omega^{(2)}$. In the vicinity of each segment γ_s, we represent the fine scales by edge bubble functions $b_s^{(\alpha)}$ that are supported over sectors $\omega_s^{(\alpha)} \subseteq \Omega_s^{(\alpha)}$ as follows:

$$\boldsymbol{v}'^{(\alpha)}\Big|_{\omega_s^{(\alpha)}} = \sum_{k=1}^{n_{sd}} \beta_k^{(\alpha)} b_s^{(\alpha)}(\boldsymbol{x})\boldsymbol{e}_k^{(\alpha)}, \qquad \boldsymbol{w}'^{(\alpha)}\Big|_{\omega_s^{(\alpha)}} = \sum_{l=1}^{n_{sd}} \eta_l^{(\alpha)} b_s^{(\alpha)}(\boldsymbol{x})\boldsymbol{e}_l^{(\alpha)} \tag{125}$$

where $\{\boldsymbol{e}_{\alpha k}\}_{k=1}^{n_{sd}}$ is a set of linearly-independent unit vectors spanning $\mathbb{R}^{n_{sd}}$, and $\beta_k^{(\alpha)}, \eta_l^{(\alpha)}$ are undetermined coefficients. The concept of segments and sectors associated with a discretized interface is illustrated in Figure 2, which is adapted from [53].

Carrying through the derivation along the lines of [53] by employing the representation (125) within the fine-scale problem associated with (123) leads to the following analytical expression for $\boldsymbol{v}'^{(\alpha)}$ at the interface:

$$\boldsymbol{v}'^{(\alpha)} \cdot \boldsymbol{n}\Big|_{\gamma_s} = (-1)^{(\alpha-1)} \tau_s^{(\alpha)} \left(-\lambda + p^{(\alpha)}\right) \tag{126}$$

$$\tau_s^{(\alpha)} = \left[\text{meas}(\gamma_s)\right]^{-1} \left(\int_{\omega_s^{(\alpha)}} \left(b_s^{(\alpha)}\right)^2 \mu^{(\alpha)}/\kappa^{(\alpha)}\, \mathrm{d}\Omega\right)^{-1} \left(\int_{\gamma_s} b_s^{(\alpha)}\, \mathrm{d}\Gamma\right)^2 \tag{127}$$

where $\tau_s^{(\alpha)}$ are stabilization parameters accounting for the element geometry and material parameters. Note that the fine scales are driven by the boundary residual, since for the exact solution of (123) – (124) we have $\lambda = p^{(\alpha)}$.

Embedding the representation of the fine scales (126) into the coarse-scale problem associated with (123) – (124) results in the following stabilized mixed weak form:

$$\sum_{\alpha=1,2} B_D^{(\alpha)}\left(\tilde{\mathbf{W}}^{(\alpha)},\tilde{\mathbf{V}}^{(\alpha)}\right)+\left\langle\left\{\tilde{\boldsymbol{w}}\cdot\boldsymbol{n}\right\},\lambda\right\rangle_{\Gamma_I}+\sum_{\alpha=1,2}\left\langle q^{(\alpha)},\tau_s^{(\alpha)}\left(-\lambda+p^{(\alpha)}\right)\right\rangle_{\Gamma_I}=\sum_{\alpha=1,2} L_D^{(\alpha)}\left(\tilde{\mathbf{W}}^{(\alpha)}\right) \quad (128)$$

$$\left\langle\mu,\left\{\boldsymbol{v}\cdot\boldsymbol{n}\right\}\right\rangle_{\Gamma_I}+\sum_{\alpha=1,2}\left\langle\mu,\tau_s^{(\alpha)}\left(-\lambda+p^{(\alpha)}\right)\right\rangle_{\Gamma_I}=0 \quad (129)$$

Since the formulation is stabilized by the terms arising consistently from the fine-scales, we are free to select the interpolation space of the multipliers without recourse to the Babuska-Brezzi condition. By adopting a discontinuous approximation of the multipliers as L_2 functions along each segment, we can solve the continuity equation (129) to obtain an analytical expression for λ

$$\lambda\big|_{\gamma_s}=\left\{p\right\}\big|_{\gamma_s}+\tau_s\left[\!\left[\tilde{\boldsymbol{v}}\right]\!\right]_{\gamma_s} \quad (130)$$

in which the numerical flux is defined as $\left\{p\right\}=\delta_s^{(1)}p^{(1)}+\delta_s^{(2)}p^{(2)}$ and the velocity jump is denoted by $\left[\!\left[\boldsymbol{v}\right]\!\right]=\boldsymbol{v}^{(1)}\cdot\boldsymbol{n}^{(1)}+\boldsymbol{v}^{(2)}\cdot\boldsymbol{n}^{(2)}$. The weighting coefficients $\delta_s^{(\alpha)}$ and the velocity penalty parameter τ_s are evaluated from the fine-scale stabilization parameters $\tau_s^{(\alpha)}$ along each interface segment as follows: $\delta_s^{(\alpha)}=\tau_s\tau_s^{(\alpha)}$, $\tau_s=\left(\tau_s^{(1)}+\tau_s^{(2)}\right)^{-1}$. In this manner, possible heterogeneity in the element geometry or material properties is accounted for within the expression for the numerical flux. Observe that by definition we have that $\delta_s^{(1)}+\delta_s^{(2)}=1$.

Substituting (130) into (128) and regrouping terms, we arrive at the stabilized interface formulation for Darcy flow:

$$\sum_{\alpha=1,2} B_D^{(\alpha)}\left(\tilde{\mathbf{W}}^{(\alpha)},\tilde{\mathbf{V}}^{(\alpha)}\right)-\left\langle\{q\},[\![\tilde{\boldsymbol{v}}]\!]\right\rangle_{\Gamma_I}+\left\langle[\![\tilde{\boldsymbol{w}}]\!],\{p\}\right\rangle_{\Gamma_I}+\left\langle[\![q]\!],\delta_s[\![p]\!]\right\rangle_{\Gamma_I} + \left\langle[\![\tilde{\boldsymbol{w}}]\!],\tau_s[\![\tilde{\boldsymbol{v}}]\!]\right\rangle_{\Gamma_I}=\sum_{\alpha=1,2} L_D^{(\alpha)}\left(\tilde{\mathbf{W}}^{(\alpha)}\right) \tag{131}$$

where the pressure (flux) jump and penalty parameter are defined as $[\![p]\!]=p^{(1)}\boldsymbol{n}^{(1)}+p^{(2)}\boldsymbol{n}^{(2)}$ and $\delta_s=\left[\left(\tau_s^{(1)}\right)^{-1}+\left(\tau_s^{(2)}\right)^{-1}\right]^{-1}$, respectively. This formulation is form-equivalent to standard DG methods for the Darcy equation (see e.g. [27]) except for the enhanced definitions of the numerical flux and penalty parameters. Comparing (131) for Darcy with (104) for mixed form of elasticity, we conclude that the flux and jump terms play the role of the interface operator $\boldsymbol{\mathcal{L}}_I\left(\boldsymbol{u}^{(1)};\boldsymbol{u}^{(2)}\right)$.

As remarked in Section 5.1.2, only certain combinations of velocity-pressure interpolations yield stable results for the standard Galerkin form, and this observation is also relevant for the present interface problem. Therefore, we elect to replace the functional forms on the domain interiors $\Omega^{(\alpha)}$ with their stabilized counterparts from Section 5.1.3, resulting in the final form of the method for non-overlapping solution decomposition: Find $\left\{\mathbf{V}^{(1)},\mathbf{V}^{(2)}\right\}\in\mathcal{Z}_1\times\mathcal{Z}_2$ such that for all $\left\{\mathbf{W}^{(1)},\mathbf{W}^{(2)}\right\}\in\mathcal{Z}_{0,1}\times\mathcal{Z}_{0,2}$:

$$\sum_{\alpha=1,2} B_{stab,D}^{(\alpha)}\left(\mathbf{W}^{(\alpha)},\mathbf{V}^{(\alpha)}\right)-\left\langle\{q\},[\![\boldsymbol{v}]\!]\right\rangle_{\Gamma_I}+\left\langle[\![\boldsymbol{w}]\!],\{p\}\right\rangle_{\Gamma_I}+\left\langle[\![q]\!],\delta_s[\![p]\!]\right\rangle_{\Gamma_I} + \left\langle[\![\boldsymbol{w}]\!],\tau_s[\![\boldsymbol{v}]\!]\right\rangle_{\Gamma_I}=\sum_{\alpha=1,2} L_{stab,D}^{(\alpha)}\left(\mathbf{W}^{(\alpha)}\right) \tag{132}$$

While the modeling approach for overlapping domains leads naturally to a staggered solution strategy, the present approach is more amenable to solving for $\mathbf{V}^{(\alpha)}$ in a concurrent fashion.

5.2 Stokes Equation

The preceding developments can be easily extended to treat the situation where both domains are governed by the Stokes equation. Let $\Omega \subset \mathbb{R}^{n_{sd}}$ be an open bounded region with piecewise smooth boundary Γ. The number of space dimensions, n_{sd}, is equal to 2 or 3. The governing equations for Stokes flow are given by the following equations:

$$\nabla p - \operatorname{div}\left(2\mu\boldsymbol{\varepsilon}\left(\boldsymbol{v}\right)\right) = \boldsymbol{f} \qquad \text{in } \Omega \qquad \text{(Stokes Equation)} \tag{133}$$

$$\operatorname{div}\boldsymbol{v} = 0 \qquad \text{in } \Omega \qquad \text{(Conservation of Mass)} \tag{134}$$

$$\boldsymbol{v} = \boldsymbol{g} \qquad \text{on } \Gamma \tag{135}$$

where $\boldsymbol{v}$ is the velocity vector, p is the pressure, $\boldsymbol{\varepsilon}\left(\boldsymbol{v}\right) = \frac{1}{2}\left[\nabla\boldsymbol{v} + \left(\nabla\boldsymbol{v}\right)^T\right]$ is the deformation rate tensor, μ is the viscosity, $\boldsymbol{f}$ is a source term, and $\boldsymbol{g}$ is the prescribed velocity on the boundary. We also define the stress tensor combining volumetric and viscous effects as $\boldsymbol{\sigma}\left(\boldsymbol{u}, p\right) = -p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}\left(\boldsymbol{u}\right)$.

Since our main focus is on developing the interfacial coupling operator, we suppress the discussion of the weak form for the domain interior and simply adopt the stabilized formulation of [36] to accommodate equal-order interpolation spaces. To reduce the formulation from the Navier-Stokes equations to the present Stokes equations, we drop the effects of the time-dependent and nonlinear advection terms. Another relevant note is that the proper function spaces for the velocity field is $\boldsymbol{H}^1\left(\Omega\right)$ rather than $\boldsymbol{H}\left(\operatorname{div},\Omega\right)$ in the case of Darcy flow. In regard to the interface, the major differences compared to Section 6.1.4 are that the viscous effects are incorporated into the

numerical flux and full continuity of the velocity is imposed across Γ_I. Sparing the details, for which the reader is referred to [53] for an analogous discussion of mixed elasticity, the resulting interfacial weak form is stated as follows:

$$\begin{aligned}&\sum_{\alpha=1,2} B^{(\alpha)}_{stab,ST}\left(\mathbf{W}^{(\alpha)},\mathbf{V}^{(\alpha)}\right)-\left\langle\left\{\boldsymbol{\sigma}^{w}\left(\boldsymbol{w},q\right)\boldsymbol{n}\right\},\left[\!\left[\boldsymbol{v}\right]\!\right]\right\rangle_{\Gamma_I}-\left\langle\left[\!\left[\boldsymbol{w}\right]\!\right],\left\{\boldsymbol{\sigma}\left(\boldsymbol{v},p\right)\boldsymbol{n}\right\}\right\rangle_{\Gamma_I}\\&-\left\langle\left[\!\left[\boldsymbol{\sigma}^{w}\left(\boldsymbol{w},q\right)\boldsymbol{n}\right]\!\right],\boldsymbol{\delta}_s\left[\!\left[\boldsymbol{\sigma}\left(\boldsymbol{v},p\right)\boldsymbol{n}\right]\!\right]\right\rangle_{\Gamma_I}+\left\langle\left[\!\left[\boldsymbol{w}\right]\!\right],\boldsymbol{\tau}_s\left[\!\left[\boldsymbol{v}\right]\!\right]\right\rangle_{\Gamma_I}=\sum_{\alpha=1,2} L^{(\alpha)}_{stab,ST}\left(\mathbf{W}^{(\alpha)}\right)\end{aligned} \tag{136}$$

Starting with the interfacial integrals, the jump and flux terms are defined as follows: $\left[\!\left[\boldsymbol{v}\right]\!\right]=\boldsymbol{v}^{(1)}-\boldsymbol{v}^{(2)}$, $\left\{\boldsymbol{\sigma n}\right\}=\boldsymbol{\delta}^{(1)}_j\left(\boldsymbol{\sigma}^{(1)}\boldsymbol{n}^{(1)}\right)-\boldsymbol{\delta}^{(2)}_j\left(\boldsymbol{\sigma}^{(2)}\boldsymbol{n}^{(2)}\right)$, and $\left[\!\left[\boldsymbol{\sigma n}\right]\!\right]=\boldsymbol{\sigma}^{(1)}\boldsymbol{n}^{(1)}+\boldsymbol{\sigma}^{(2)}\boldsymbol{n}^{(2)}$, where the arguments to the stress tensor (either $\left(\boldsymbol{v},p\right)$ or $\left(\boldsymbol{w},q\right)$) have been suppressed to make the notation compact. Also, in the modified stress tensor $\boldsymbol{\sigma}_w$ for the weighting function, the sign of the pressure variable q is reversed: $\boldsymbol{\sigma}_w\left(\boldsymbol{w},q\right)=q\boldsymbol{I}+2\mu\boldsymbol{\varepsilon}\left(\boldsymbol{w}\right)$. The stabilization tensors are evaluated as $\boldsymbol{\delta}^{(\alpha)}_s=\boldsymbol{\tau}_s\boldsymbol{\tau}^{(\alpha)}_s$, $\boldsymbol{\tau}_s=\left(\boldsymbol{\tau}^{(1)}_s+\boldsymbol{\tau}^{(2)}_s\right)^{-1}$, and $\boldsymbol{\delta}_s=\left[\left(\boldsymbol{\tau}^{(1)}_s\right)^{-1}+\left(\boldsymbol{\tau}^{(2)}_s\right)^{-1}\right]^{-1}$, where the individual fine-scale tensors $\boldsymbol{\tau}^{(\alpha)}_s$ are obtained as:

$$\boldsymbol{\tau}^{(\alpha)}_s=\left[\text{meas}\left(\gamma_s\right)\right]^{-1}\left(\int_{\gamma_s} b^{(\alpha)}_s\,\mathrm{d}\Gamma\right)^2\left[\left(\int_{\omega^{(\alpha)}_s}\mu\nabla b^{(\alpha)}_s\cdot\nabla b^{(\alpha)}_s\,\mathrm{d}\Omega\right)\boldsymbol{I}+\int_{\omega^{(\alpha)}_s}\mu\nabla b^{(\alpha)}_s\otimes\nabla b^{(\alpha)}_s\,\mathrm{d}\Omega\right]^{-} \tag{137}$$

The domain interior contributions from the bilinear form and linear form $B_{stab,ST}\left(\mathbf{W},\mathbf{V}\right)$ and $L_{stab,ST}\left(\mathbf{W}\right)$, respectively, are determined as follows:

$$B_{stab,ST}\left(\mathbf{W},\mathbf{V}\right)=B_{ST}\left(\mathbf{W},\mathbf{V}\right)+\left(\left(\nabla q+2\mu\,\mathrm{div}\,\boldsymbol{\varepsilon}\left(\boldsymbol{w}\right)\right),\boldsymbol{\tau}\left(\nabla p-2\mu\,\mathrm{div}\,\boldsymbol{\varepsilon}\left(\boldsymbol{v}\right)\right)\right) \tag{138}$$

$$L_{stab,ST}\left(\mathbf{W}\right)=L_{ST}\left(\mathbf{W}\right)+\left(\left(\nabla q+2\mu\,\mathrm{div}\,\boldsymbol{\varepsilon}\left(\boldsymbol{w}\right)\right),\boldsymbol{f}\right) \tag{139}$$

in which the standard Galerkin terms are as follows:

$$B_{ST}\left(\mathbf{W},\mathbf{V}\right)=\left(\boldsymbol{\varepsilon}\left(\boldsymbol{w}\right),2\mu\,\boldsymbol{\varepsilon}\left(\boldsymbol{v}\right)\right)-\left(\mathrm{div}\,\boldsymbol{w},p\right)+\left(q,\mathrm{div}\,\boldsymbol{v}\right) \tag{140}$$

$$L_{ST}\left(\mathbf{W}\right)=\left(\boldsymbol{w},\boldsymbol{f}\right) \tag{141}$$

and the Stokes stabilization tensor $\boldsymbol{\tau}$ is defined in terms of an element interior bubble function b^e according to the formula presented in Section 4.1.1 of [36]:

$$\boldsymbol{\tau}=b^e\int_{\Omega^e}b^e\,\mathrm{d}\Omega\left[\left(\int_{\Omega^e}\mu\nabla b^e\cdot\nabla b^e\,\mathrm{d}\Omega\right)\boldsymbol{I}+\int_{\Omega^e}\mu\nabla b^e\otimes\nabla b^e\,\mathrm{d}\Omega\right]^{-1} \tag{142}$$

5.3 Coupling of Different PDEs: Stokes-Darcy System

In this section, we consider the coupling of Stokes and Darcy flow as a model problem to develop the interface coupling operator in the case of non-overlapping solution decomposition. While similar developments could be performed for combining Darcy and Darcy-Stokes flow regimes as presented in the preceding sections, we instead focus on a related topic which has recently been an active area of research. A sampling of applications for combined Stokes and Darcy models include the modeling of contaminant transfer between rivers and groundwater, coupled models of nutrient transfer between biological tissue and the bloodstream, and design of industrial filtration systems. A common approach to couple the flow regions is through domain decomposition; see for example the work by Vassilev and Yotov [54] in which the transport equation is also incorporated. A

summary of numerical techniques for coupling flow regimes is contained in [18]. Herein, we adopt the procedure proposed by Truster and Masud [51,53] to derive a primal interface operator with the character of a Discontinuous Galerkin (DG) method by starting from a Lagrange multiplier interface formulation. This approach relies crucially on applying concepts from the VMS method [24] locally at the interface between the local and global models to derive the numerical flux terms for the DG method, which allows for different element types and jumps in material properties between the two models. First, we discuss the procedure for the case when both models are represented by the same governing equations, and then we present the generalization to the Stokes-Darcy system in this section.

In the following developments, the Darcy regime is indicated as $\Omega^{(1)}$ and the Stokes regime as $\Omega^{(2)}$. When distinct Stokes and Darcy flow regimes are combined within the same modeling domain, the interfacial conditions between the two regions play a key role in properly modeling the physics. Additionally, the conditions must be mathematically consistent with the different regularity requirements for functions in $\boldsymbol{H}\left(\text{div}, \Omega^{(1)}\right)$ compared to $\boldsymbol{H}^1\left(\Omega^{(2)}\right)$ that are associated with the Darcy equation and Stokes equation, respectively. Herein, we impose the following conditions for the velocity and flux fields:

$$\boldsymbol{v}^{(1)} \cdot \boldsymbol{n}^{(1)} + \boldsymbol{v}^{(2)} \cdot \boldsymbol{n}^{(2)} = 0 \qquad \text{on } \Gamma_I \tag{143}$$

$$\boldsymbol{n}^{(2)} \cdot \boldsymbol{\sigma}^{(2)}\left(\boldsymbol{v}^{(2)}, p^{(2)}\right)\boldsymbol{n}^{(2)} = p^{(1)} \qquad \text{on } \Gamma_I \tag{144}$$

$$-\left[\boldsymbol{I} - \boldsymbol{n}^{(2)} \otimes \boldsymbol{n}^{(2)}\right]\boldsymbol{\sigma}^{(2)}\left(\boldsymbol{v}^{(2)}, p^{(2)}\right)\boldsymbol{n}^{(2)} = \frac{2\mu\alpha_o}{\sqrt{\kappa}}\left[\boldsymbol{I} - \boldsymbol{n}^{(2)} \otimes \boldsymbol{n}^{(2)}\right]\boldsymbol{v}^{(2)} \qquad \text{on } \Gamma_I \tag{145}$$

Equations (145) and (146) represent the continuity of the normal component of the velocity and flux field, respectively. The third condition (145) imposes the so-called Beavers-Joseph-Saffman law [48] to account for the experimentally observed slip at the surface of the porous medium, where α_o is an experimentally determined material parameter. The combination of equations (143) – (145) implies that both normal and tangential boundary conditions are assigned to the Stokes region while only the normal-direction conditions are assigned to the Darcy region.

The standard Lagrange multiplier formulation obtained by combining the governing equations for Darcy (108) – (110) and Stokes (133) – (135) with weak imposition of the interface conditions (143) – (145) is as follows:

$$\sum_{\alpha=1,2} B_D^{(\alpha)}\left(\mathbf{W}^{(\alpha)},\mathbf{V}^{(\alpha)}\right)+\left\langle\left\{\boldsymbol{w}\cdot\boldsymbol{n}\right\},\lambda\right\rangle_{\Gamma_I}+\left\langle \boldsymbol{w}^{(2)},\frac{2\alpha_o\mu^{(2)}}{\sqrt{K_I}}\left[\boldsymbol{I}-\boldsymbol{n}\otimes\boldsymbol{n}\right]\boldsymbol{v}^{(2)}\right\rangle_{\Gamma_I} = \sum_{\alpha=1,2} L_D^{(\alpha)}\left(\mathbf{W}^{(\alpha)}\right) \tag{146}$$

$$\left\langle\mu,\left\{\boldsymbol{v}\cdot\boldsymbol{n}\right\}\right\rangle_{\Gamma_I}=0 \tag{147}$$

where $\left\{\boldsymbol{v}\cdot\boldsymbol{n}\right\}=\boldsymbol{v}^{(1)}\cdot\boldsymbol{n}-\boldsymbol{v}^{(2)}\cdot\boldsymbol{n}$ is the numerical flux. To derive the interfacial operator for this model system, we combine the developments in the preceding sections and utilize the local fine-scale modeling procedure on either side of the interface. Presently, the enforcement of normal-direction continuity alone implies that the Stokes stabilization tensor from (137) must be converted into a scalar quantity. Modifying the corresponding developments in Section 5.2 leads to the scalar stabilization parameter: $\tau_s^{(2)}=\boldsymbol{n}\cdot\boldsymbol{\tau}_s^{(2)}\boldsymbol{n}$

Carrying out the steps for the interface modeling and incorporating the domain interior stabilization for both models, we arrive at the composite interface formulation for Stokes and Darcy flow:

$$\begin{aligned}\sum_{\alpha=1,2} B^{(\alpha)}_{stab,D}\left(\mathbf{W}^{(\alpha)},\mathbf{V}^{(\alpha)}\right) &- \left\langle \left\{ \boldsymbol{n}\cdot\boldsymbol{\sigma}_w\left(\boldsymbol{w},q\right)\boldsymbol{n}\right\},\left[\!\left[\boldsymbol{v}\right]\!\right]\right\rangle_{\Gamma_I} - \left\langle \left[\!\left[\boldsymbol{w}\right]\!\right],\left\{ \boldsymbol{n}\cdot\boldsymbol{\sigma}\left(\boldsymbol{v},p\right)\boldsymbol{n}\right\}\right\rangle_{\Gamma_I} \\ &- \left\langle \left[\!\left[\boldsymbol{\sigma}_w\left(\boldsymbol{w},q\right)\boldsymbol{n}\right]\!\right],\delta_s\left[\!\left[\boldsymbol{\sigma}\left(\boldsymbol{v},p\right)\boldsymbol{n}\right]\!\right]\right\rangle_{\Gamma_I} + \left\langle \left[\!\left[\boldsymbol{w}\right]\!\right],\tau_s\left[\!\left[\boldsymbol{v}\right]\!\right]\right\rangle_{\Gamma_I} \\ &+ \left\langle \boldsymbol{w}^{(2)},\frac{2\alpha_o\mu^{(2)}}{\sqrt{K_I}}\left[\boldsymbol{I}-\boldsymbol{n}\otimes\boldsymbol{n}\right]\boldsymbol{v}^{(2)}\right\rangle_{\Gamma_I} = \sum_{\alpha=1,2} L^{(\alpha)}_{stab,D}\left(\mathbf{W}^{(\alpha)}\right)\end{aligned} \tag{148}$$

The interface terms in (148) are defined using composite formulas including both the Darcy and Stokes contributions:

$$\left[\!\left[\boldsymbol{v}\right]\!\right] = \boldsymbol{v}^{(1)}\cdot\boldsymbol{n}^{(1)} + \boldsymbol{v}^{(2)}\cdot\boldsymbol{n}^{(2)} \tag{149}$$

$$\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}\boldsymbol{n}\right\} = \delta_s^{(1)}\left(-p^{(1)}\right) + \delta_s^{(2)}\boldsymbol{n}^{(2)}\cdot\boldsymbol{\sigma}^{(2)}\boldsymbol{n}^{(2)}, \qquad \left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}_w\boldsymbol{n}\right\} = \delta_s^{(1)}\left(q^{(1)}\right) + \delta_s^{(2)}\boldsymbol{n}^{(2)}\cdot\boldsymbol{\sigma}_w^{(2)}\boldsymbol{n}^{(2)} \tag{150}$$

$$\left[\!\left[\boldsymbol{\sigma}\boldsymbol{n}\right]\!\right] = \left(-p^{(1)}\right)\boldsymbol{n}^{(1)} + \boldsymbol{\sigma}^{(2)}\left(\boldsymbol{v}^{(2)},p^{(2)}\right)\boldsymbol{n}^{(2)}, \qquad \left[\!\left[\boldsymbol{\sigma}_w\boldsymbol{n}\right]\!\right] = \left(q^{(1)}\right)\boldsymbol{n}^{(1)} + \boldsymbol{\sigma}_w^{(2)}\left(\boldsymbol{w}^{(2)},q^{(2)}\right)\boldsymbol{n}^{(2)} \tag{151}$$

and recall that the modified stress tensor $\boldsymbol{\sigma}_w\left(\boldsymbol{w},q\right)$ is defined in Section 5.2.

An algorithm for solving the discrete counterpart of the coupled system (148) is provided in Table 1.

Table 1: Algorithm for formation and solution of non-overlapping Stokes-Darcy system

- Assemble contributions from elements $\Omega_e^{(1)}$ in Darcy subdomain $\Omega^{(1)}$:

$$B_{stab,D}^{(1)}\left(\mathbf{W}^{(1)},\mathbf{V}^{(1)}\right),\ L_{stab,D}^{(1)}\left(\mathbf{W}^{(1)}\right) \quad \text{from (121), (122)} \tag{a}$$

- Assemble contributions from elements $\Omega_e^{(2)}$ in Stokes sub-domain $\Omega^{(2)}$:

$$B_{stab,ST}^{(2)}\left(\mathbf{W}^{(2)},\mathbf{V}^{(2)}\right),\ L_{stab,ST}^{(2)}\left(\mathbf{W}^{(2)}\right) \quad \text{from (138), (139)} \tag{b}$$

- Assemble contributions from interface segments γ_s corresponding to element pairs $\left\{\Omega_{e_s}^{(1)},\Omega_{e_s}^{(2)}\right\}$ such that $\partial\Omega_{e_s}^{(1)}\bigcap\partial\Omega_{e_s}^{(2)}\neq\varnothing$:

$$\begin{aligned}&-\left\langle\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}_w\boldsymbol{n}\right\},\left[\!\left[\boldsymbol{v}\right]\!\right]\right\rangle_{\Gamma_I}-\left\langle\left[\!\left[\boldsymbol{w}\right]\!\right],\left\{\boldsymbol{n}\cdot\boldsymbol{\sigma}\boldsymbol{n}\right\}\right\rangle_{\Gamma_I}-\left\langle\left[\!\left[\boldsymbol{\sigma}_w\boldsymbol{n}\right]\!\right],\delta_s\left[\!\left[\boldsymbol{\sigma}\boldsymbol{n}\right]\!\right]\right\rangle_{\Gamma_I}\\&+\left\langle\left[\!\left[\boldsymbol{w}\right]\!\right],\tau_s\left[\!\left[\boldsymbol{v}\right]\!\right]\right\rangle_{\Gamma_I}+\left\langle\boldsymbol{w}^{(2)},\frac{2\alpha_o\mu^{(2)}}{\sqrt{K_I}}\left[\boldsymbol{I}-\boldsymbol{n}\otimes\boldsymbol{n}\right]\boldsymbol{v}^{(2)}\right\rangle_{\Gamma_I}\end{aligned} \tag{c}$$

- Solve the fully coupled system (148) for $\mathbf{V}^{(1)}$ and $\mathbf{V}^{(2)}$

Remark: The stabilization parameters τ_s, δ_s and the weighting coefficients $\delta_s^{(\alpha)}$ encapsulate the information concerning the material properties, element geometry, and differential operators through the fine-scale models (127) and (137). Additionally, the flux and jump terms are such that the functions from $\boldsymbol{H}\left(\text{div},\Omega^{(1)}\right)$ and $\boldsymbol{H}^1\left(\Omega^{(2)}\right)$ mathematically commute across the interface. Thus, these DG terms provide a rich underlying mathematical structure to the coupled problem (148).

CHAPTER 6: VARIATIONAL EMBEDDING OF PHYSICS BASED DATA

In this chapter, we present a framework for variationally embedding physics-based data into the interface formulation given in Chapters 4 and 5. We also briefly discuss the machine learning concepts and algorithms relevant to our framework.

6.1 Overview of Machine Learning

Machine Learning (ML) is the field of study that gives computers the ability to *learn* without being explicitly programmed. The rules for learning are based on optimization of some useful data. One of the most basic machine learning models that exist is the *linear regression*. There are three basic components of ML: (1) A (parametrized) model, (2) Data, and (3) Optimization strategy (a way to minimize error between data and predicted values). This is also called *model training* as it produces optimized parameters to *predict* the output corresponding to any input value. *Parameters* in a model can also be treated as degrees of freedom.

There is a large variety of machine learning methods including ridge regression, lasso regression, k-nearest neighbors, support vector machines, Gaussian processes, neural networks, decision trees, and ensemble methods (which combine different models). Each particular machine learning method contains its own error function and optimization algorithm. Most methods also allow the user to choose a number of values called *hyperparameters*, which give control over some aspects of the method's behavior. The book in references [49] provide a good introduction to the topic.

In our framework, we employ a stabilized numerical method for solving boundary-value problems. We optimize the solution method by performing a least-squares best fit between the model predicted values and the data, using two approaches: (i) ordinary least squares regression, and (ii)

kernel supported regression. The optimized results are also subject to the satisfaction of essential constraints and conservation laws. We provide below a basic description of the above-mentioned methods that have been employed in this work.

Linear regression is a common statistical tool for modeling the relationship between some 'explanatory' input variables and some real valued outputs. Cast as a machine learning problem, the domain set $\boldsymbol{X}$ is a subset of $\mathbb{R}^d$, for some d data samples, and the *target* set $\boldsymbol{Y}$ is the set of real numbers. We would like to learn a function $h:\mathbb{R}^d \to \mathbb{R}$ that best approximates the relationship between our variables. The class of linear regression predictors is simply the set of linear functions:

$$L_d = \left\{ x \mapsto h_{\boldsymbol{w},b}\left(\boldsymbol{x}\right) : \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R} \right\} \tag{152}$$

i-e L_d is a set of functions, where each function is parameterized by a weight vector $\boldsymbol{w} \in \mathbb{R}^d$, a bias $b \in \mathbb{R}$, and each such function takes as input a vector $\boldsymbol{x}$ and returns the scalar output $h_{\boldsymbol{w},b} = \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b$.

Next, we define a *loss function* $\ell\left(h,\left(\boldsymbol{x},y\right)\right)$ for regression. This simply indicates whether $h\left(\boldsymbol{x}\right)$ correctly predicts a *target value* y or not. One common way is to use a squared-loss function in regression, defines as:

$$\ell\left(h,\left(\boldsymbol{x},y\right)\right) = \left(h\left(\boldsymbol{x}\right) - y\right)^2 \tag{153}$$

For this loss function, the empirical risk function is called the Mean Squared Error, which is defined on a given training S with m values as:

$$L_S(h) = \frac{1}{m}\sum_{i=1}^{m}\left(h(\boldsymbol{x}_i) - y_i\right)^2 \tag{154}$$

Least squares is the algorithm that solves the empirical risk minimization (ERM) problem for linear regression predictors with respect to the squared loss. The ERM problem for least squares is given as:

$$\arg\min_{\boldsymbol{w}} L_S(h_{\boldsymbol{w}}) = \arg\min_{\boldsymbol{w}} \frac{1}{m}\sum_{i=1}^{m}\left(h(x_i) - y_i\right)^2 \tag{155}$$

And the solution to the above problem gives parameters that minimizes the error:

$$\boldsymbol{w} = \boldsymbol{A}^{-1}\boldsymbol{b} \tag{156}$$

$$\text{where, } \boldsymbol{A} = \left(\sum_{i=1}^{m} x_i x_i^T\right) \quad \text{and} \quad \boldsymbol{b} = \left(\sum_{i=1}^{m} y_i x_i\right) \tag{157}$$

Linear regression can also fit nonlinear functions (e.g. higher order polynomials) if a mapping is applied to the vector $\boldsymbol{x}$. The mapping $\varphi : \mathbb{R}^n \to \mathbb{R}^d$ is done through a *kernel function* which provides a way to manipulate data as though it were projected into a higher dimensional space, by operating on it in its original space. So the approximation function is given by $h(\boldsymbol{x}) = \left(\langle \boldsymbol{w}, \varphi(\boldsymbol{x})\rangle - b\right)$ where $\boldsymbol{w} \in \mathbb{R}^d$ and $\varphi(\boldsymbol{x})$ is any mapping from $\mathbb{R}^n \to \mathbb{R}^d$. Predictions made with this method have a computational cost of $O(d)$.

In addition, since the mappings only appear inside inner products, for some particular kernel function $K(\boldsymbol{x}, \boldsymbol{x}') = \langle \varphi(\boldsymbol{x}), \varphi(\boldsymbol{x}')\rangle$ we can obtain the inner product directly and without explicitly

applying the mapping to input vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$. Thus, the method allows us to map our data/ features into infinitely high-dimensional spaces without any significant computational cost. This procedure is called a *kernel trick* in the machine learning literature. Interested reader is referred to [49] for further details on its implementation. Furthermore, to save the cost of evaluating kernels during the learning process, it is pre-computed for all the pairs of training examples in the dataset and the resulting matrix of inner products is called the Gram matrix i-e $G_{i,j} = K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$. Key properties of a kernel and a Gram matrix is that it must be symmetric and semi-positive definite.

A feature mapping, φ, may also be viewed as expanding the lower dimensional space to a richer or higher dimensional space. One can therefore think of a mapping φ as a way to express and utilize a priori knowledge about the problem at hand. For example, in solving a boundary-value problem, if we know that there exists a component of the solution in a richer space, we can design the mapping to incorporate that in our solution space.

Most of the regression algorithms are based on vectors as input, but in many real cases input samples can be in a high dimensional space and are represented in the form of tensors. A kernel support tensor regression algorithm is proposed in [17] which takes a multi-dimensional input, maps each row of the tensor into a higher dimensional feature space using a kernel, and then compute regression functions of the form given in (152). In this case, the kernel function is a matrix as opposed to the case of support vector regression where it is a scalar.

6.2 Data Embedding in DG Interface Formulation

We consider that some data (e.g. sensor displacements) is available in a small patch of a larger domain. We treat this patch as a discontinuity in that domain. The idea is to couple the two domains

together using the earlier derived DG framework so that the data in one small patch improve the overall accuracy of the solution in the larger domain. Figure 5 illustrates the idea in which Ω_2 represents the data patch in the domain Ω_1 that are coupled together through an interface Γ_I.

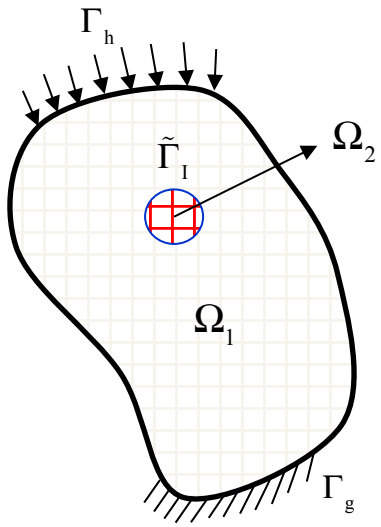


Figure 5: Data patch embedded in a domain with a DG interface

The data-embedded problem is run in two steps. In the first step, the standard interface formulation for a particular PDE is solved to get the numerical solution in domain Ω_1. Then we introduce data in the second step through the interface term $\left\langle [\![\boldsymbol{w}]\!], \boldsymbol{\tau}_s [\![\boldsymbol{u}]\!] \right\rangle_{\tilde{\Gamma}_I}$ as it enforces continuity of the primal fields, a residual of the numerical solution in the first step and the data is computed, and the problem is solved again with that residual on the right hand side. The residual $\boldsymbol{r}\left(\boldsymbol{u}_{(1)}, \boldsymbol{u}^{(data)}\right) = \left(\boldsymbol{u}^{(1)}_{(1)} - \boldsymbol{u}^{(data)}\right)$ determines how much our computed solution is off from the *target values*. In this way, second step acts as a corrector to the numerical solution computed in the first step. Thus, the modified interface term for the second step is:

$$\left\langle \left[\!\left[\boldsymbol{w} \right]\!\right], \boldsymbol{\tau}_s \left[\!\left[\boldsymbol{u} \right]\!\right] \right\rangle_{\tilde{\Gamma}_I} = \left\langle \left[\!\left[\boldsymbol{w} \right]\!\right], \tilde{\boldsymbol{\tau}}_s \left[\!\left[\boldsymbol{u} - \boldsymbol{r}\left(\boldsymbol{u}_{(1)}, \boldsymbol{u}^{(data)} \right) \right]\!\right] \right\rangle_{\tilde{\Gamma}_I} \tag{158}$$

where $\left(\bullet \right)_{(step)}^{(domain)}$, and $\tilde{\boldsymbol{\tau}}_s = 10\ \boldsymbol{\tau}_s$ i-e the variationally derived stabilization parameter is amplified by a constant factor to have a stricter enforcement of penalty across the interface. Here we want to make a crucial observation about (158) that it behaves like a least-squares type linear regression function with the empirical risk minimization (ERM) problem similar to (157) is given as:

$$\arg\min_{\boldsymbol{u}} L_S\left(l\left(\boldsymbol{u}, \boldsymbol{r} \right), \tilde{\boldsymbol{\tau}}_s \right) = \arg\min_{\boldsymbol{u}} \frac{1}{2} \left\langle \tilde{\boldsymbol{\tau}}_s \left[\!\left[\boldsymbol{u} - \boldsymbol{r}\left(\boldsymbol{u}_{(1)}, \boldsymbol{u}^{(\text{data})} \right) \right]\!\right]^2 \right\rangle_{\tilde{\Gamma}_I} \tag{159}$$

Where $l\left(\boldsymbol{u}, \boldsymbol{r} \right) = \left[\!\left[\boldsymbol{u} - \boldsymbol{r}\left(\boldsymbol{u}_{(1)}, \boldsymbol{u}^{(\text{data})} \right) \right]\!\right]^2$ is the squared loss function.

Now, (158) can be inserted in the interface formulation of a particular PDE to get a data-embedded system. For example, for the mixed form of elasticity, (103) becomes:

$$\begin{aligned} &\sum_{\alpha=1,2} \left[\left(\boldsymbol{\varepsilon}\left(\hat{\boldsymbol{w}}^{(\alpha)} \right), 2\mu^{(\alpha)} \boldsymbol{\varepsilon}\left(\hat{\boldsymbol{u}}^{(\alpha)} \right) \right)_{\tilde{\Omega}^{(\alpha)}} + \left(\nabla \cdot \hat{\boldsymbol{w}}^{(\alpha)}, p^{(\alpha)} \right)_{\tilde{\Omega}^{(\alpha)}} \right] \\ &+ \sum_{\alpha=1,2} \left[\left(q^{(\alpha)}, \nabla \cdot \hat{\boldsymbol{u}}^{(\alpha)} \right)_{\tilde{\Omega}^{(\alpha)}} - \left(q^{(\alpha)}, p^{(\alpha)} / \lambda \right)_{\tilde{\Omega}^{(\alpha)}} \right] - \left\langle \left[\!\left[\hat{\boldsymbol{w}} \right]\!\right], \left\{ \boldsymbol{n} \cdot \boldsymbol{\sigma}\left(\hat{\boldsymbol{u}}, p \right) \right\} \right\rangle_{\tilde{\Gamma}_I} \\ &\qquad - \left\langle \left\{ \boldsymbol{n} \cdot \boldsymbol{\sigma}\left(\hat{\boldsymbol{w}}, q \right) \right\}, \left[\!\left[\hat{\boldsymbol{u}} \right]\!\right] \right\rangle_{\tilde{\Gamma}_I} + \left\langle \left[\!\left[\hat{\boldsymbol{w}} \right]\!\right], \tilde{\boldsymbol{\tau}}_s \left[\!\left[\hat{\boldsymbol{u}} \right]\!\right] \right\rangle_{\tilde{\Gamma}_I} \\ &\qquad = \sum_{\alpha=1,2} \left(\hat{\boldsymbol{w}}^{(\alpha)}, \boldsymbol{f}^{(\alpha)} \right)_{\tilde{\Omega}^{(\alpha)}} + \left\langle \left[\!\left[\hat{\boldsymbol{w}} \right]\!\right], \tilde{\boldsymbol{\tau}}_s\ \boldsymbol{r}\left(\hat{\boldsymbol{u}}_{(1)}, \boldsymbol{u}^{(data)} \right) \right\rangle_{\tilde{\Gamma}_I} \end{aligned} \tag{160}$$

Remark: The advantage of embedding data through the residual is that it retains the variational consistency of the formulation while improving the accuracy of the solution.

Remark: Equation (160) represents the data-embedded system for mixed form of elasticity. For other PDEs discussed in Chapters 5 and 6, a similar system can be obtained by substituting (158) in the interface formulation.

Remark: To validate the method, we have used the exact solution as data at the patch nodal points in the test cases presented in Chapter 7 since we do not have any observational or sensor data available.

6.3 Interpretation of Stabilization Parameter τ_s as a Kernel Function

In VMS method, a solution field $\boldsymbol{u}$ is decomposed into coarse scales $\bar{\boldsymbol{u}}$, which are resolved by a given mesh, and unresolved or fine scales $\boldsymbol{u}'$. In other words, fine scales $\boldsymbol{u}'$ represent the orthogonal component of the solution of a PDE in a higher dimensional space that is not resolved by our numerical solution $\bar{\boldsymbol{u}}$. These fine scales give rise to the discretization error, which otherwise if we add to our numerical solution would give us the exact solution (Figure 6).

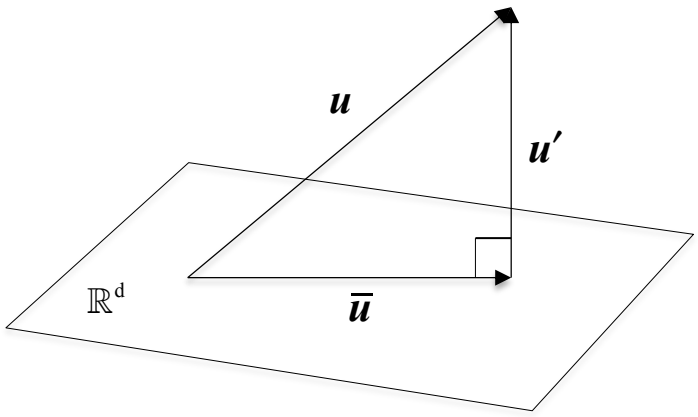


Figure 6: Decomposition of a solution field into coarse and fine scale components

As shown in Section 4, fine scales are solved in terms of a stabilization parameter $\boldsymbol{\tau}_s$. The structure of the stabilization matrix $\boldsymbol{\tau}_s$ is derived by employing higher order edge bubble functions to represent the fine scales. $\boldsymbol{\tau}_s$ stabilizes and enriches our numerical solution by projecting a higher dimensional component of the solution onto the lower dimensional coarse scales.

We have stated in Section 6.1 that a kernel function, through a mapping, expands a lower dimesnional space to a richer or higher dimesnional space and it can be used to incorporate priori knwledge about the problem at hand. In view of this, we can interpret $\boldsymbol{\tau}_s$ as a kernel function that enriches our solution field by projecting a higher dimesnional solution component onto it. From the structure of $\boldsymbol{\tau}_s$ given in (102), we see that it is symmetric and positive definite, which are the essential properties of a kernel or a Gram matrix. Thus, in (159) when $\boldsymbol{\tau}_s$ acts on the loss function, it magnifies its impact by enriching its space and we see a significant improvement in the accuracy of the solution field with little amount of embedded data.

CHAPTER 7: NUMERICAL RESULTS

Figure 7 shows the equal order elements employed in the numerical studies. In all the test cases, velocity and pressure fields are assumed continuous across elements within individual modeling regions. The following quadrature rules were used throughout: linear quadrilaterals, 2×2 Gauss quadrature; linear triangles, 4-point quadrature; higher-order elements, appropriate extensions of the linear rules [23].

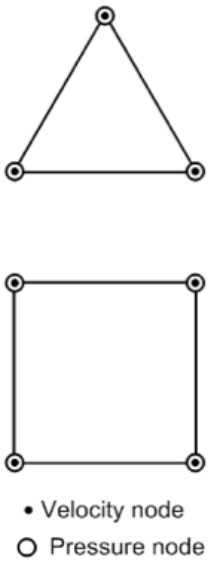


Figure 7: Equal order triangular and quadrilateral elements

The bubble functions employed for the domain-based and interface-based stabilization terms are given in terms of element natural coordinates (ξ,η) by the expressions shown in Table 2 and Table 3, respectively. The element type abbreviations designate the shape of the element, either triangular (T) or quadrilateral (Q), and the number of nodes per element, varying between 3 and 4.

Table 2: Interior bubble functions employed for fine-scale fields

Element	Bubble Function
T3	$\xi\eta(1-\xi-\eta)$
Q4	$(1-\xi^2)(1-\eta^2)$

Table 3: Edge bubble functions employed for fine-scale fields

Element	Bubble Function
T3	$4\xi(1-\xi-\eta)$
Q4	$\frac{1}{2}(1-\xi^2)(1-\eta)$

7.1 Mixed Form of Elasticity: L-shaped Domain

We start with the case of an L-shaped domain loaded in a manner to produce the deformation corresponding to Mode 1 fracture. This problem exhibits a singularity in the stress and pressure fields at the reentrant corner, and therefore serves as a mathematically hard problem for the convergence rate study and for evaluation of the error estimators. Plane strain conditions are assumed, tractions derived from the exact solution are applied on all edges of the domain, and the mesh is constrained to exclude rigid body modes. The description of the problem is shown in Figure 8; the exact solution derived from elasticity theory is given in (161) – (163) [50]. To make the problem harder and test the robustness of the DG interface method, the numerical simulation is run on a non-conforming meshes with an interface inserted in between. $\Omega^{(1)}$ represents part of the domain with a coarser mesh while $\Omega^{(2)}$ is the region having a finer mesh with Γ_{12} as DG interface.

$$u_x^{\text{exact}} = \frac{r^{\lambda}}{2G}\left[\left\{k - Q(\lambda+1)\right\}\cos(\lambda\theta) - \lambda\cos\left((\lambda-2)\theta\right)\right] \tag{161}$$

$$u_y^{\text{exact}} = \frac{r^{\lambda}}{2G}\left[\left\{k + Q(\lambda+1)\right\}\sin(\lambda\theta) + \lambda\sin\left((\lambda-2)\theta\right)\right] \tag{162}$$

$$p^{\text{exact}} = \frac{2\nu\lambda r^{\lambda-1}}{1-2\nu}\left[(k-1)\cos\left((\lambda-1)\theta\right)\right] \tag{163}$$

Where following parameters are selected for this test case:

$$\lambda = 0.54448373678246398, \qquad Q = 0.54307557883673652 \tag{164}$$

$$E = 7.5 \text{ x } 10^7, \quad \nu = 0.4999, \quad G = \frac{E}{2(1+\nu)}, \quad k = 3 - 4\nu \tag{165}$$

Four convergence rate studies employing 96, 384, 1536, and 6144 linear quadrilateral elements are presented. The first case involves no data while the other three involves data embedded in the domain in different forms. In cases 2 and 3, data is embedded in a small patch of the domain as the exact solution at nodal points. Case 4 has data in the discrete form at the nodal points along an element edge. Figure 9(a) shows domain partitioning with boundary conditions while Figure 9(b) depicts coarsest discretization with embedded data types and locations.

The convergence rates for the standard and explicit error in the displacement field measured in the L_2 norm and H^1 seminorm are presented in Figure 10. We observe a reduced rate of convergence in both the norms due to the singularity of the solution field at the reentrant corner. From finite element theory, the convergence rate for this type of problem is governed by the regularity of the

solution [2]. Specifically, the rate for the H^1 seminorm should match the value of $\lambda = 0.545$ in (164), which is in fact the case for all the cases.

From Figure 10, it is observed that data embedding considerably improves the accuracy of the solution while still retaining the variational consistency of the DG interface method. The absolute value of both standard and residual based (explicit) error significantly decreases in a consistent manner. In the case of H^1 seminorm, convergence rates also improve slightly. Regarding the study involving the effect of the location and the type of data embedded in the domain, we see from cases 2 and 3 that the method does not depend on the location of a data patch in the domain. However, if data is introduced in a discrete fashion along an edge, the results are slightly worse as compared to the case of data in a patch.

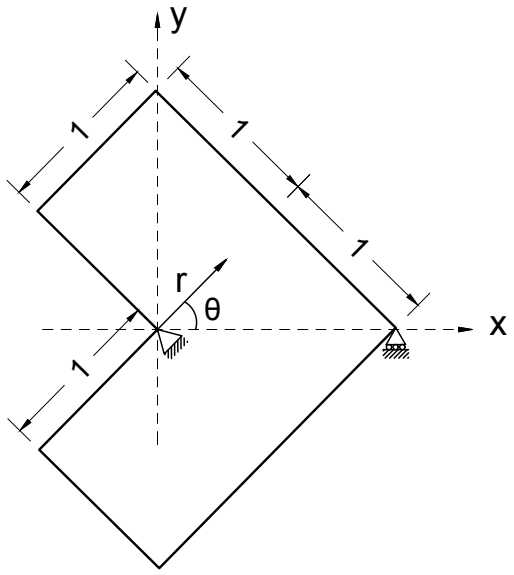
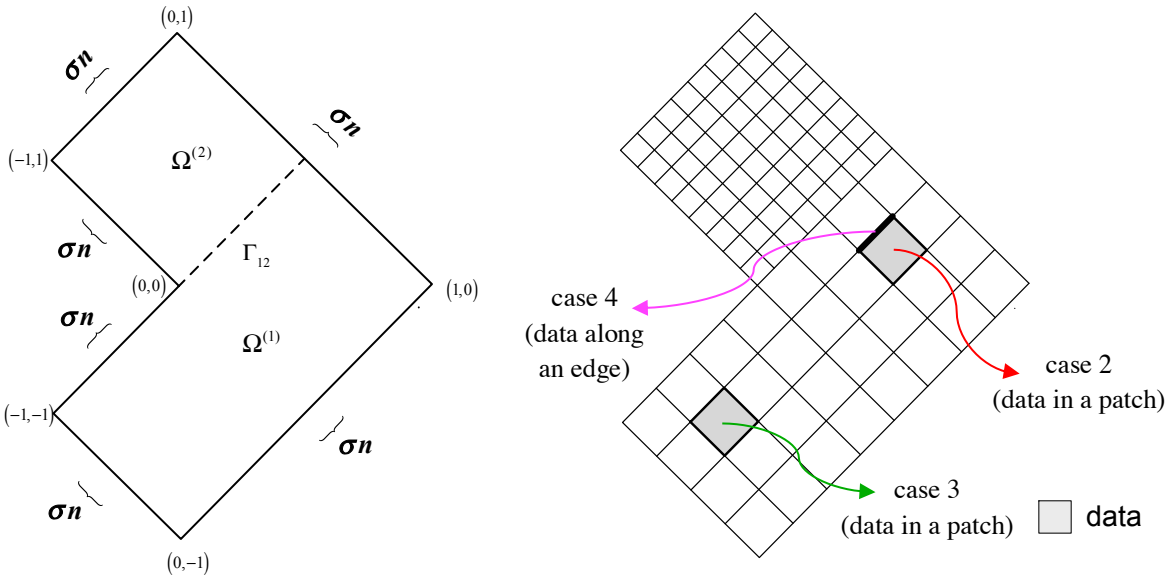
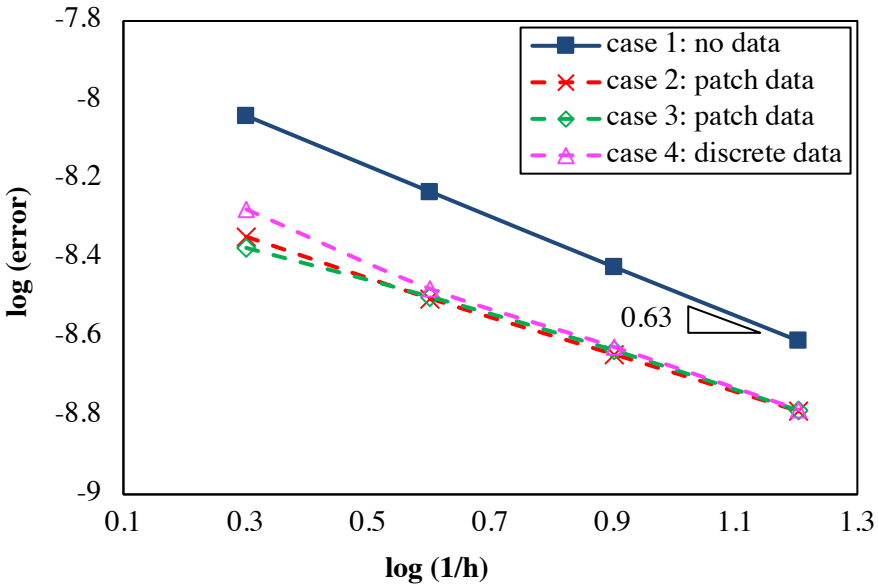


Figure 8: L-shaped domain problem description.



(a) (b)

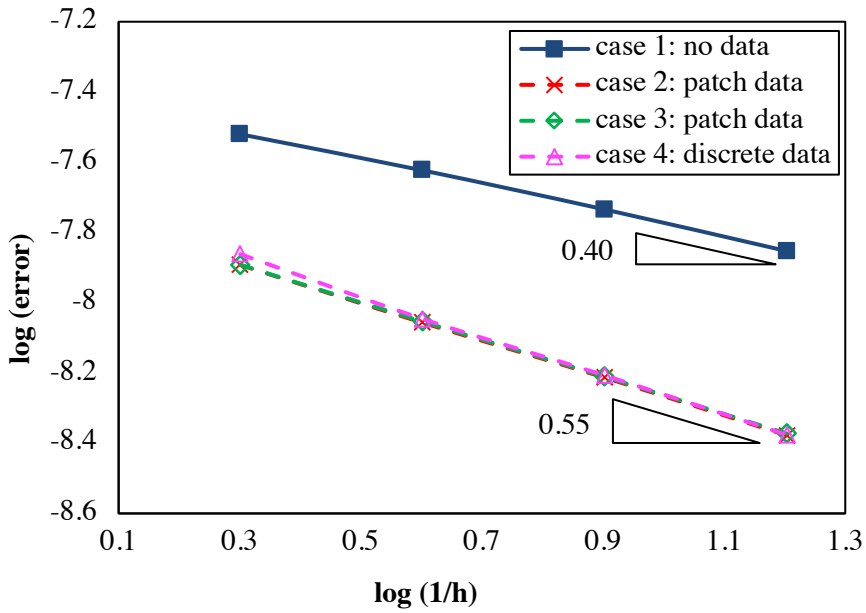
Figure 9: L-shaped domain problem: (a) domain description (b) non-conforming mesh with data



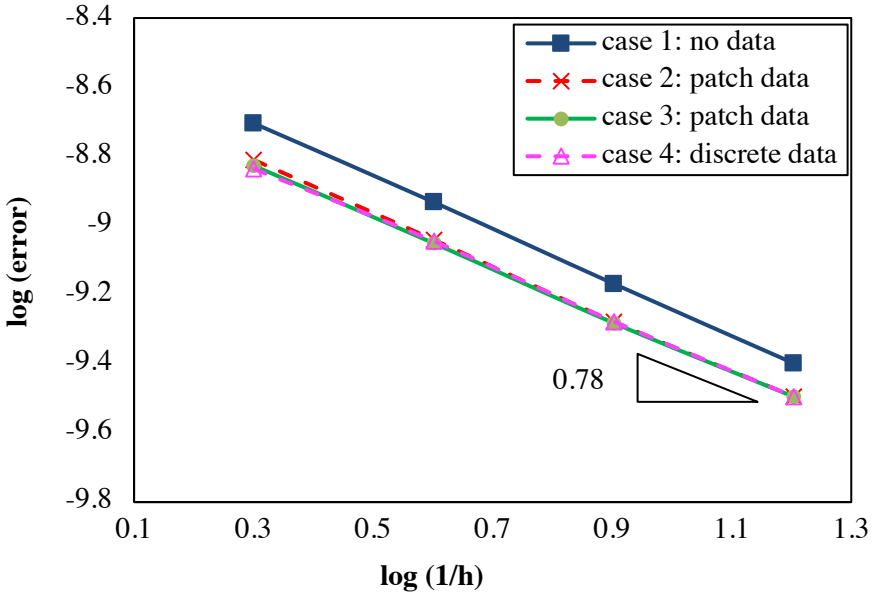
(a)

Figure 10: Convergence rates of error estimates: (a) L_2 norm of standard displacement error (b) H^1 seminorm of standard displacement error (c) L_2 norm of (residual based) explicit error (d)) H^1 seminorm of (residual based) explicit error

Figure 10: (cont.)

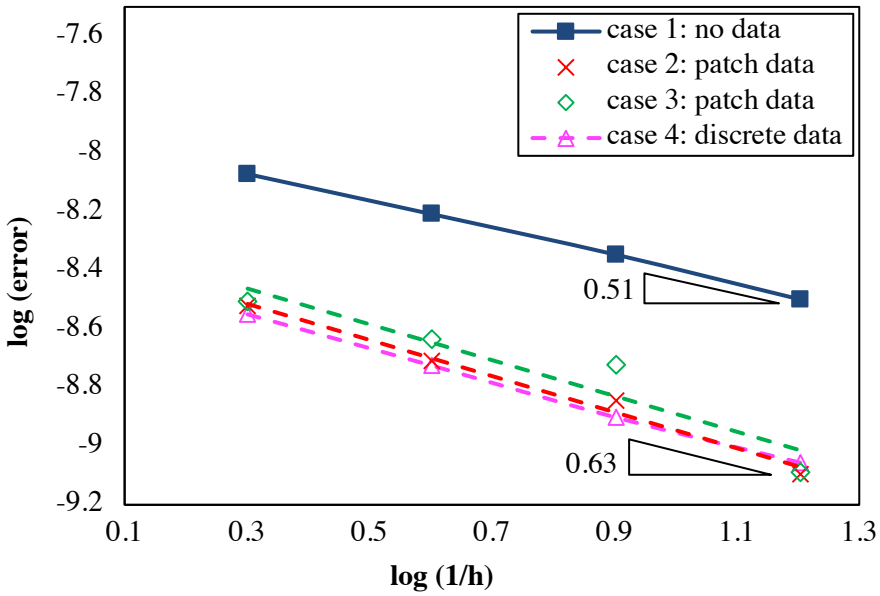


(b)



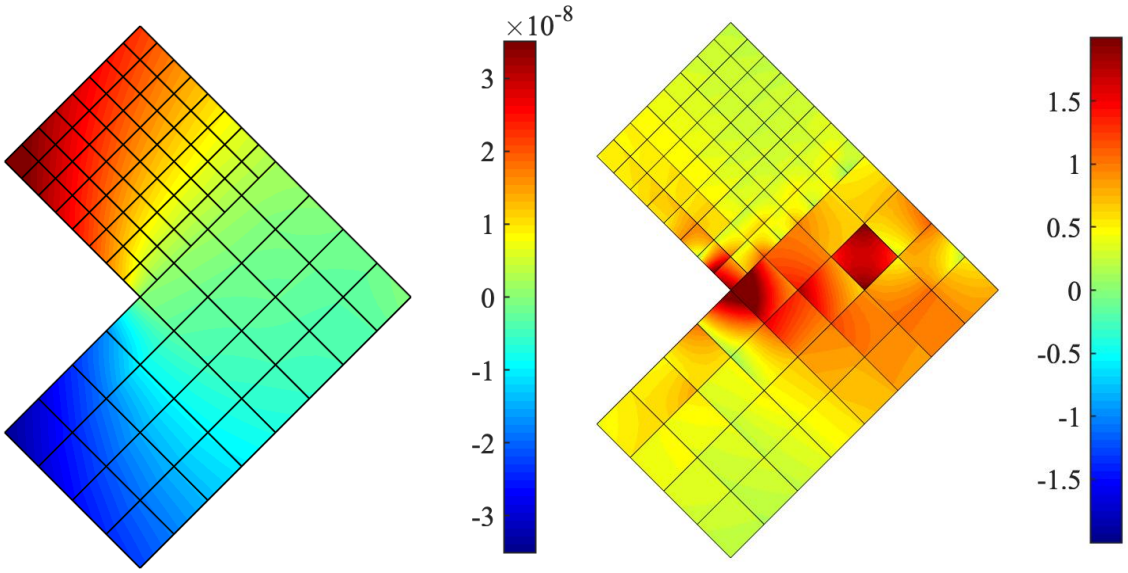
(c)

Figure 10: (cont.)



(d)

The contour plots for vertical displacement and unsmoothed axial stress in the non-conforming mesh are shown in Figure 11. The singularity of the stress field at the reentrant corner can be seen in Figure 11(b) due to which there is a minimal disturbance of the stress field across the DG interface. However, the displacement field shows perfect continuity across the non-conforming mesh confirming the robustness of the interface formulation in Section 4.2 employed here.



(a) (b)

Figure 11: Contour plots of (a) vertical displacement u_y (b) unsmoothed axial stress σ_{xx}

7.2 Coupled PDEs in Elasticity

As a benchmark problem for the multi-PDE case, the problem of an L-shaped domain shown in Figure 8 and with the exact solution given in (161) – (164) is again considered. Problem description is given in Figure 11(a). We model the region $\Omega^{(1)}$ using mixed elasticity formulation given in Section 4.2 and the region $\Omega^{(2)}$ using pure displacment formulation of Section 4.1. The two regions are are coupled together through a DG interface Γ_{12}. Thus, the composite formulas from Section 4.3 are utilized for the numerical flux and stability parameters. The material parameters are same as in (165) except the Poisson's ratio which in this case is 0.3 to accommodate pure displacement formulation. We consider two subcases of the multi-PDE problem: (i) case 1 has a non-conforming mesh and linear quadrilateral elements in both regions (ii) case 2 has a non-conforming mesh but with triangular elements in the pure displacement region and quadrilateral elements in the mixed-elasticity region. Coarsest discretization for both the cases is shown in Figures 12(b) – (c), respectively.

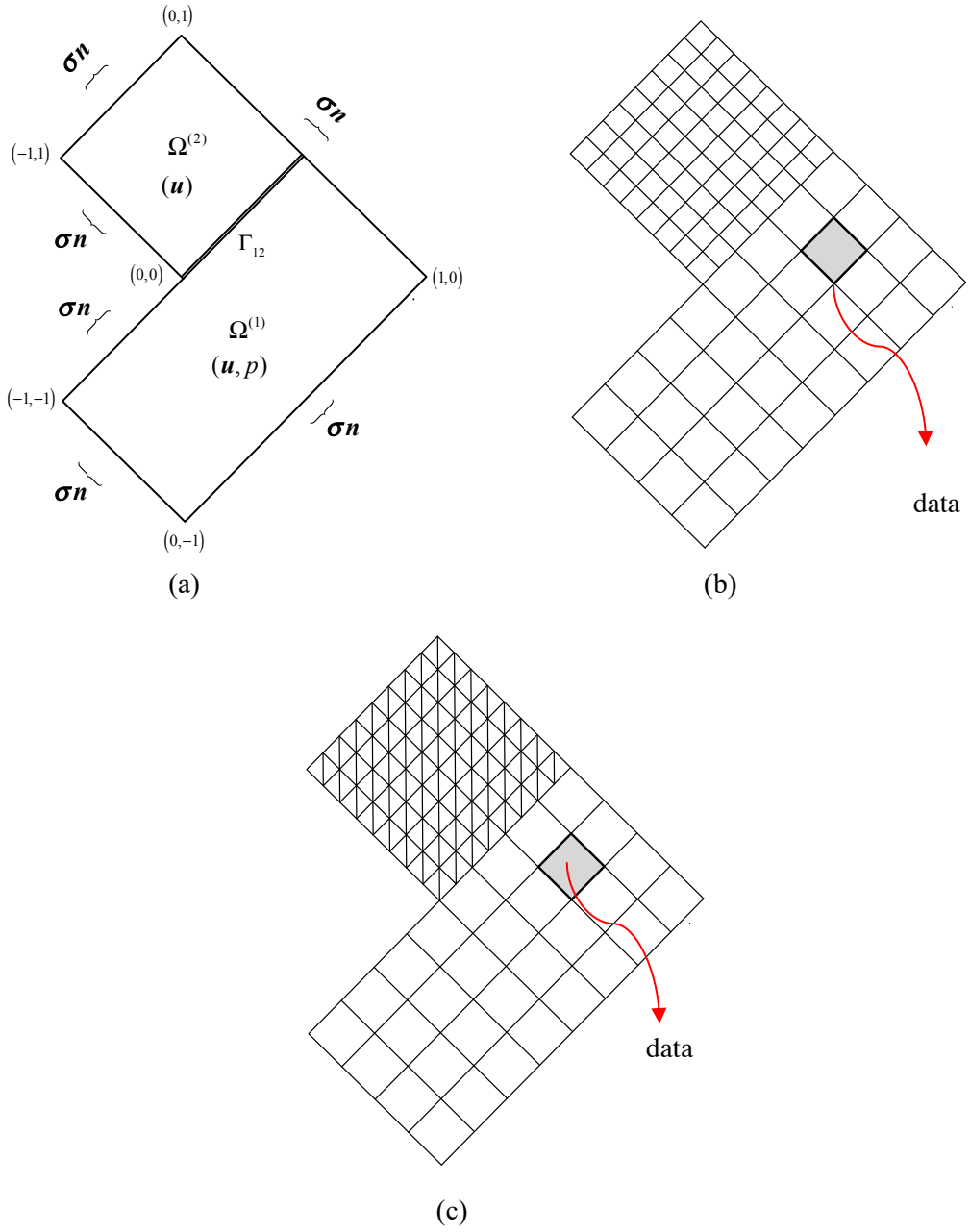
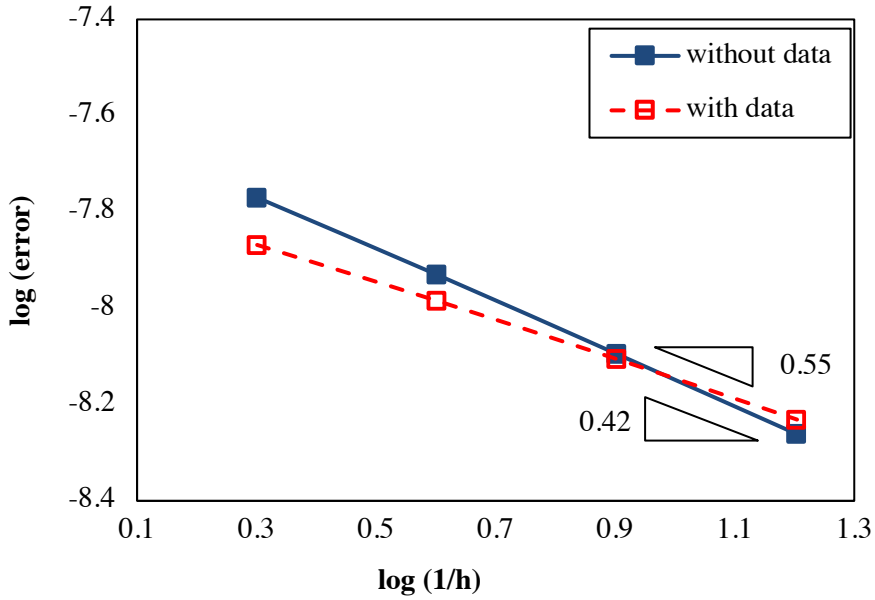


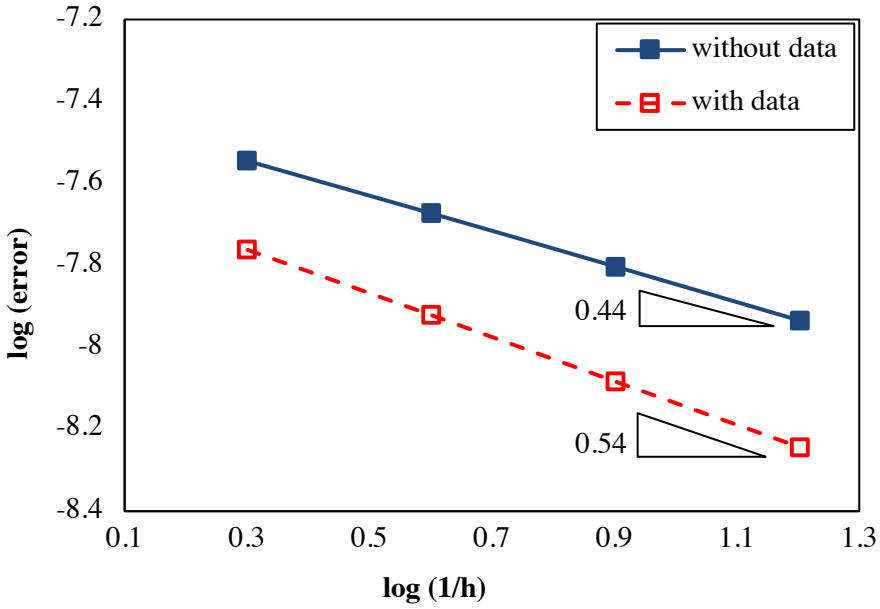
Figure 12: Schematics of multi-PDE L-shaped domain problem (a) problem description (b) case 1: non-conforming mesh and same element type (c) non-conforming mesh with different element types

The convergence of the displacement error measured in the L_2 norm and H^1 seminorm for cases 1 and 2 is presented in Figures 12 and 13, respectively. For the convergence study, case 1 was run with a hierarchal mesh of 96, 384, 1536, and 6144 linear quadrilateral elements while 160, 640, 2560, and 10240 linear triangular and quadrilateral elements were employed in case 2. Like the

previous example, slopes of the error norms are sub-optimal due to singularity in the solution field. However, convergence rates of H^1 seminorm conform to the regularity of the solution which is given by $\lambda = 0.545$.

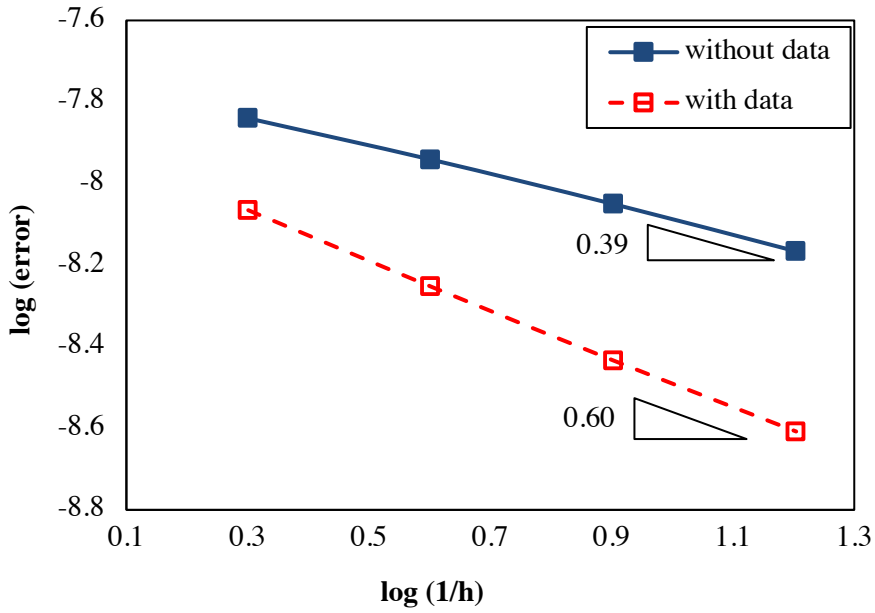


(a)

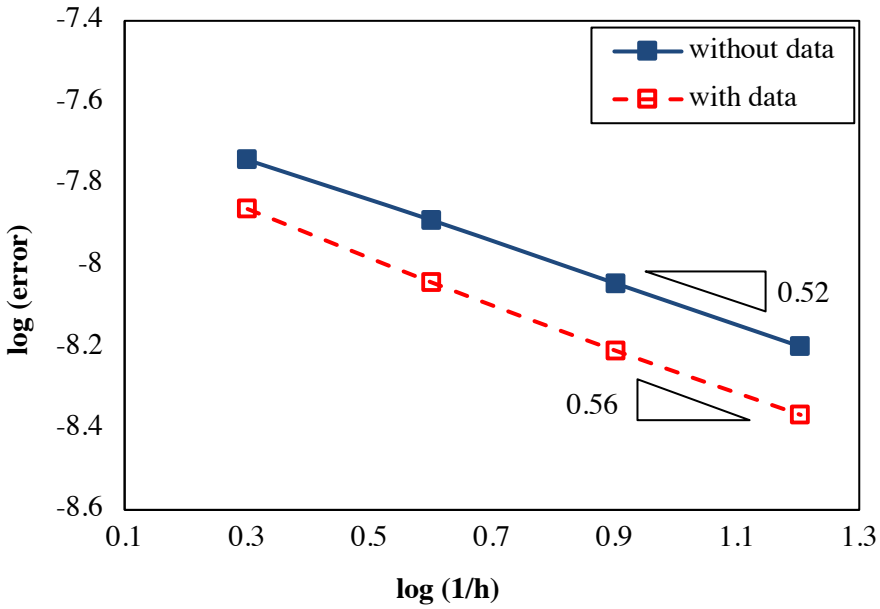


(b)

Figure 13: Convergence rates of standard displacement error for case 1 (a) L_2 norm (b) H^1 seminorm



(a)



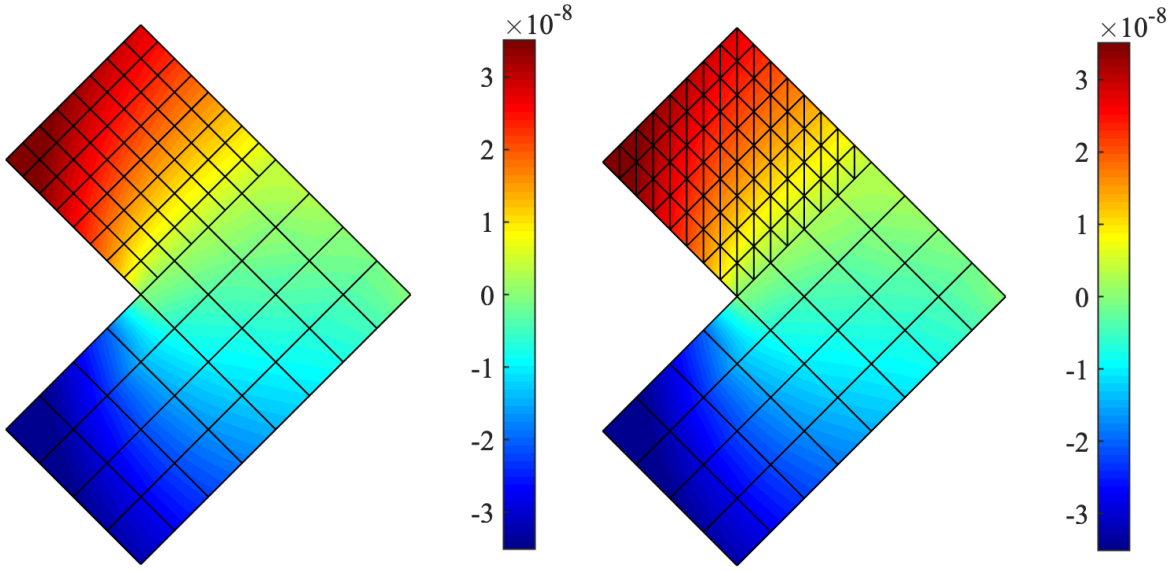
(b)

Figure 14: Convergence rates of standard displacement error for case 2 (a) L_2 norm (b) H^1 seminorm

It is observed that the data-embedded method significantly improves the accuracy of the solution. With the inclusion of data as the exact solution at nodal points in a small patch of the problem

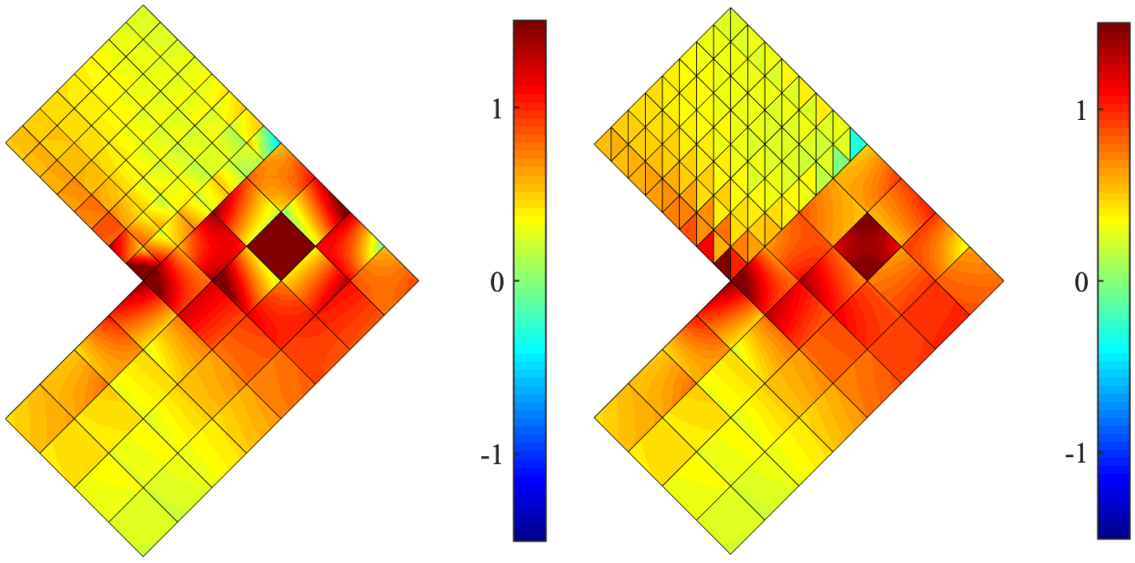
domain, both the error norms of the standard displacement error decrease and the convergence rate improves. However, with the introduction of data in case 1, L_2 norm of the displacement error decrease for the coarser meshes and the effect diminishes as we refine the mesh.

As a qualitative assessment of the solution accuracy provided by the interface formulation, we present the contour plots for vertical displacement and unsmoothed axial stress for both the cases in Figures 15 and 16, respectively. No smoothing techniques have been applied to the field; the strains and stresses are directly evaluated through differentiating the finite element displacements and evaluating the constitutive relations (86) and (99), respectively. The stress singularity can be seen at the reentrant corner due to which we see some oscillations at the DG interface.



(a) (b)

Figure 15: Contour plots for vertical displacement u_y for (a) case 1 (b) case 2



(a) (b)

Figure 16: Contour plots for unprocessed axial stress σ_{xx} for (a) case 1 (b) case 2

7.3 Darcy Flow

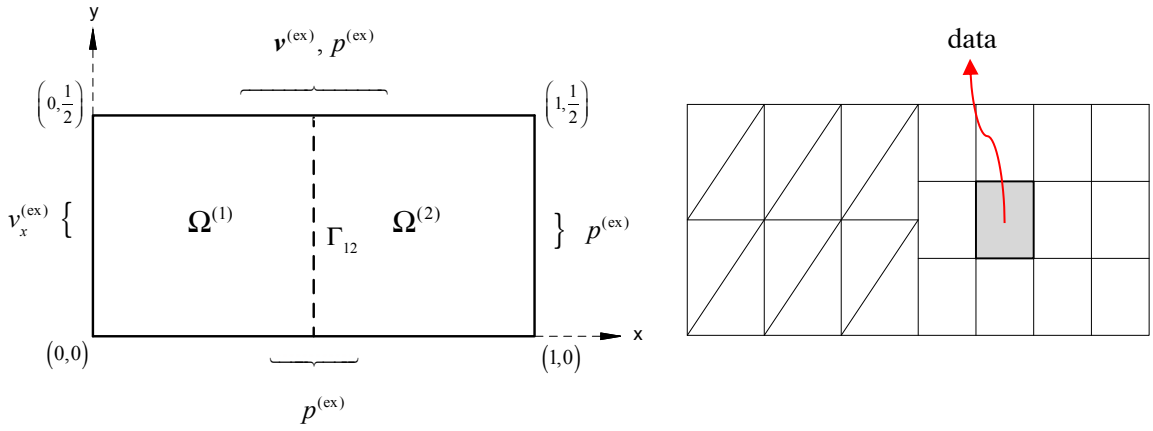
Now we turn towards problems in fluid mechanics. The first test case is of a Darcy flow in a domain $\left(0,1\right)\times\left(0,\frac{1}{2}\right)$. The exact solution for this problem is taken from [54] and is given in (166) – (168). The domain is composed of subregions in which $\Omega^{(1)}$ comprises of triangular elements while $\Omega^{(2)}$ contains quadrilateral elements. The non-matching meshes are patched together via the interface operator presented in Section 5.1. The problem description and the coarsest non-matching mesh with the embedded data are shown in Figure 17.

$$v_x^{\text{exact}} = \sin\left(\frac{x}{G}+\omega\right)e^{y/G}+\omega\sin\left(\omega x\right) \tag{166}$$

$$v_y^{\text{exact}} = -\cos\left(\frac{x}{G}+\omega\right)e^{y/G} \tag{167}$$

$$p^{\text{exact}} = \frac{\mu G}{\kappa}\cos\left(\frac{x}{G}+\omega\right)e^{y/G} + \frac{\mu}{\kappa}\cos(\omega x) \tag{168}$$

where the parameter $G = \sqrt{\kappa}/\alpha_o$. The material properties are taken as $\kappa = 1$, $\mu = 1$, $\alpha_o = 0.5$, and $\omega = 1.05$.

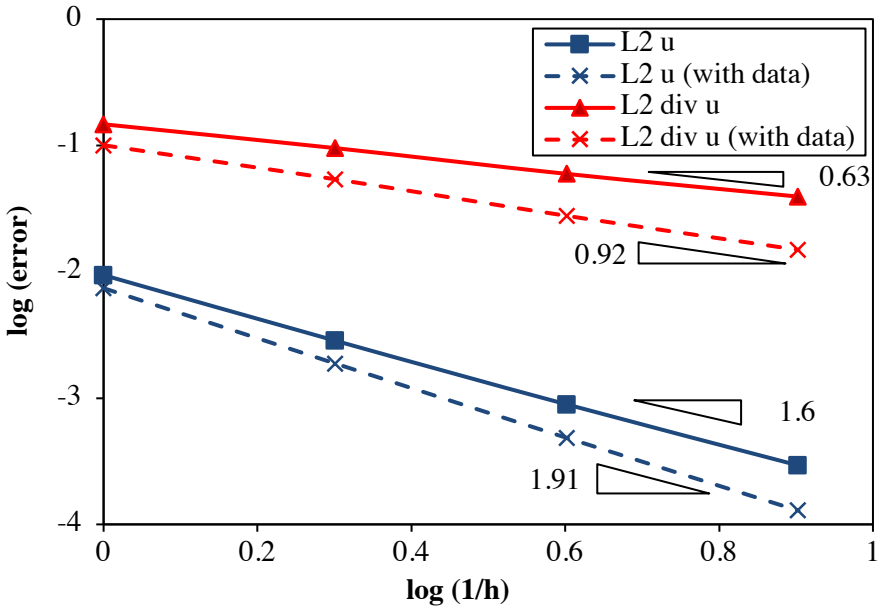


(a) (b)

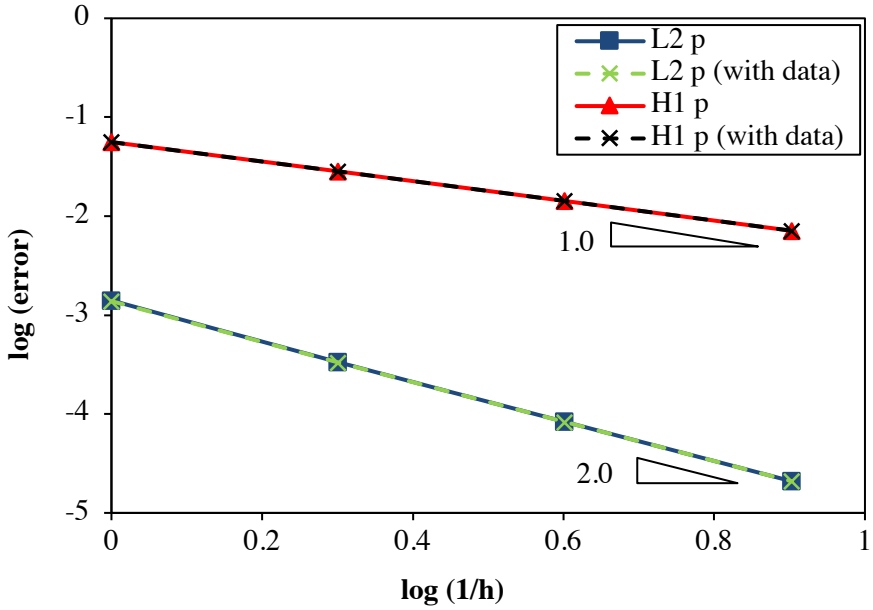
Figure 17: Schematic of Darcy flow case (a) problem description (b) coarsest non-conforming mesh

A convergence study was employed to see the performance of the interface operator and the impact of data on the accuracy of the numerical solution. The meshes employed have 24, 96, 384, and 1536 elements. Without the embedded data, velocity error norms show the convergence rates of 1.6 and 0.63 (Figure 18(a)) which is less than the theoretically predicted values of 2.0 and 1.0. However, with the introduction of data, we not only see a decrease in the velocity error norms but also the rates of convergence improve to 1.91 and 0.92 which are closer to theoretical values. The pressure error norms do not show any improvement in terms of reduction in absolute error or the convergence rate with the introduction of data (Figure 18(b)). The reason might be that the pressure norms achieve theoretical convergence rates of 2.0 and 1.0 without the introduction of data, and

when the data is embedded in the form of the exact solution it does not have any significant impact as the numerical solution is already closer to the exact solution.



(a)



(b)

Figure 18: Convergence rates of L_2 norm and H^1 seminorm for (a) velocity field (b) pressure field

The contour plots for the velocity and pressure fields are given in Figure 19. We remark that the interface between the triangular and quadrilateral elements is very smooth and nearly invisible. This shows the effectiveness of the interface formulation employed here across different element types.

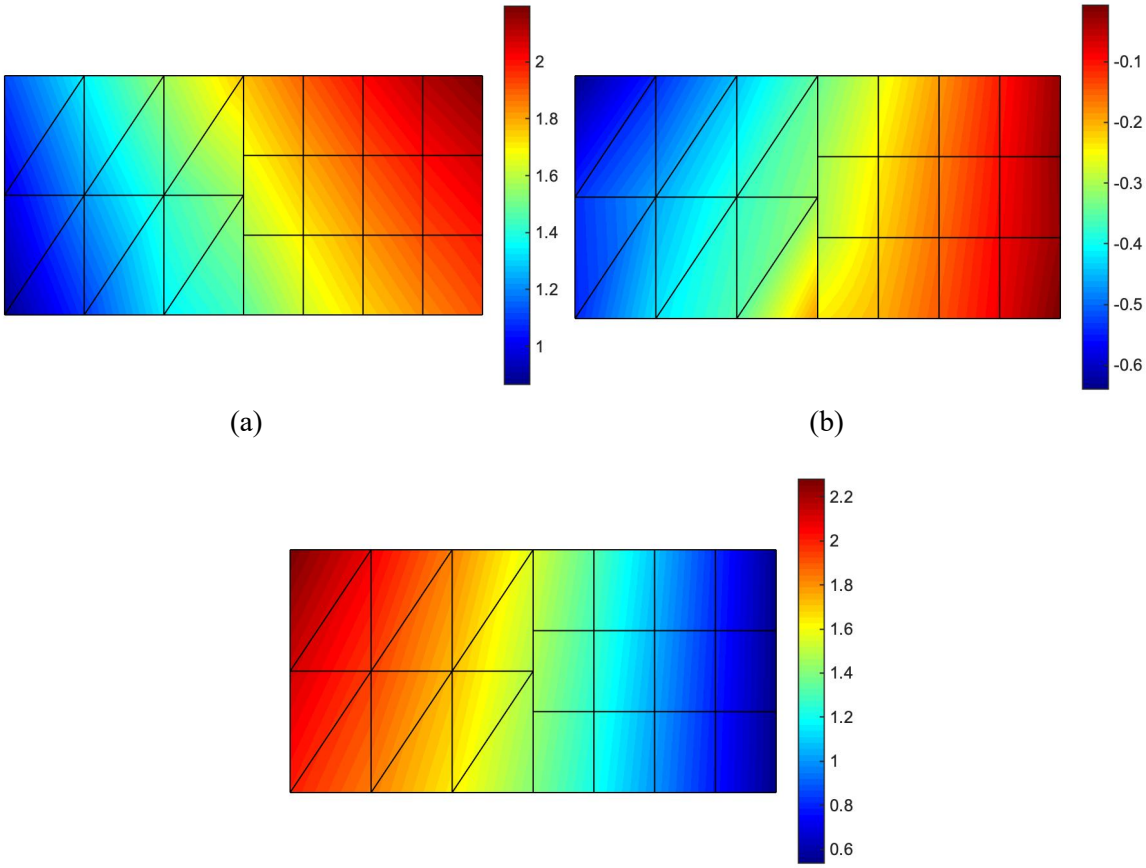


Figure 19: Contour plots for (a) velocity v_x field (b) velocity v_y field (c) pressure field

7.4 Stokes Flow in L-shaped Domain

To study the limitations imposed by singularities of the geometry for a fluid mechanics problem and the role of data in improving the solution, we consider the L-shaped domain $\Omega = \Omega_0 \setminus \Omega_1$, where $\Omega_0 = (-1,1)\times(-1,1)$ and $\Omega_1 = (0,1)\times(-1,0)$ are the square domains. Since Ω has a

reentrant corner at the point $(0,0)$, the exact solution is singular at the origin. This example proposed in [55] has the exact solution:

$$v_x^{\text{exact}} = r^{\lambda}\left[(\lambda+1)\sin(\theta)\Phi(\theta)+\cos(\theta)\Phi'(\theta)\right] \tag{169}$$

$$v_y^{\text{exact}} = r^{\lambda}\left[-(\lambda+1)\cos(\theta)\Phi(\theta)+\sin(\theta)\Phi'(\theta)\right] \tag{170}$$

$$p^{\text{exact}} = -r^{(\lambda-1)}(1-\lambda)^{-1}\left[(\lambda+1)^2\Phi'(\theta)+\Phi'''(\theta)\right] \tag{171}$$

where

$$\begin{aligned}\Phi(\theta) = {} & \sin((1+\lambda)\theta)\cos(\omega\lambda)/(1+\lambda)-\cos((1+\lambda)\theta) \\ & -\sin((1-\lambda)\theta)\cos(\omega\lambda)/(1-\lambda)+\cos((1-\lambda)\theta)\end{aligned} \tag{172}$$

$$\lambda = 0.54448373678246398, \qquad \omega = 3\pi/2 \tag{173}$$

The problem domain consists of two subdomains: $\Omega^{(1)}$ has a coarser mesh of quadrilateral elements while $\Omega^{(2)}$ has a finer mesh of same element type. The two non-conforming mesh are coupled together through an interface formulation presented in Section 5.2. Dirichlet boundary conditions are applied all over the domain edges. A schematic of the problem description and the coarsest mesh with a patch of data is shown in Figure 20.

Then, a convergence rate study with hierarchal meshes of 96, 348, 1536, and 6144 elements is presented in Figure 21. Due to singularity in the geometry, the convergence rates of the L_2 norm and H^1 seminorm of the velocity field error are sub-optimal. And same is the case with the L_2

norm of the pressure field error. However, the convergence rates are not less than the regularity of the solution ($\lambda = 0.545$) in all the cases. When a patch of data is embedded in the domain, we see a marginal decrease in the L_2 error norm of velocity and pressure fields. However, H^1 seminorm of velocity field error does not show any improvement.

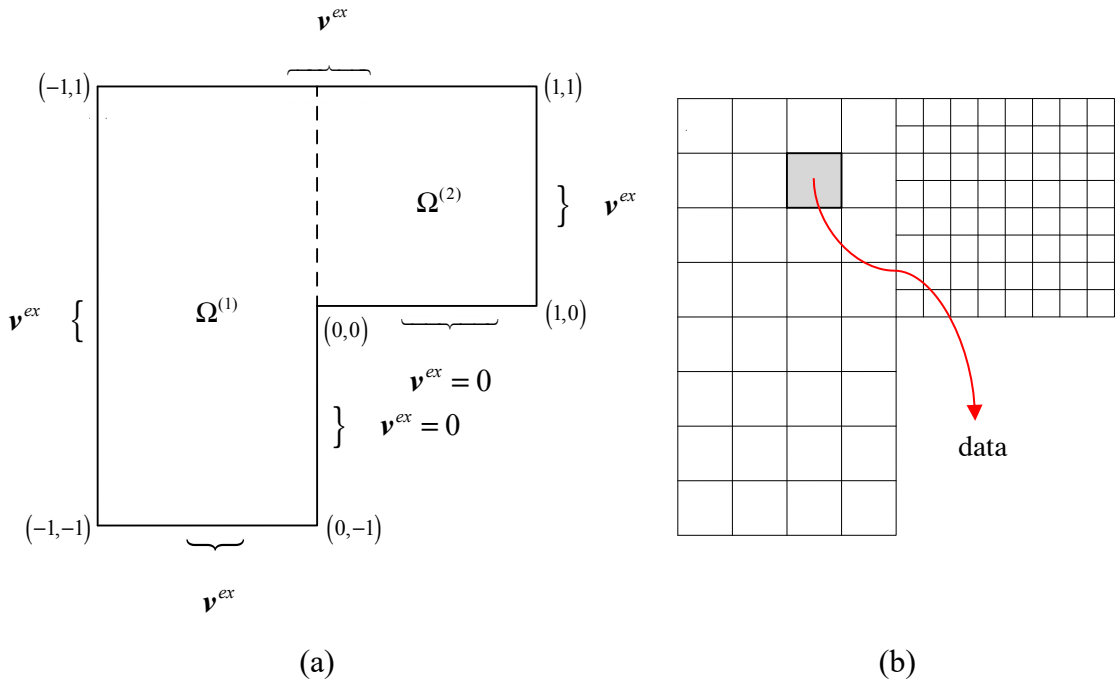
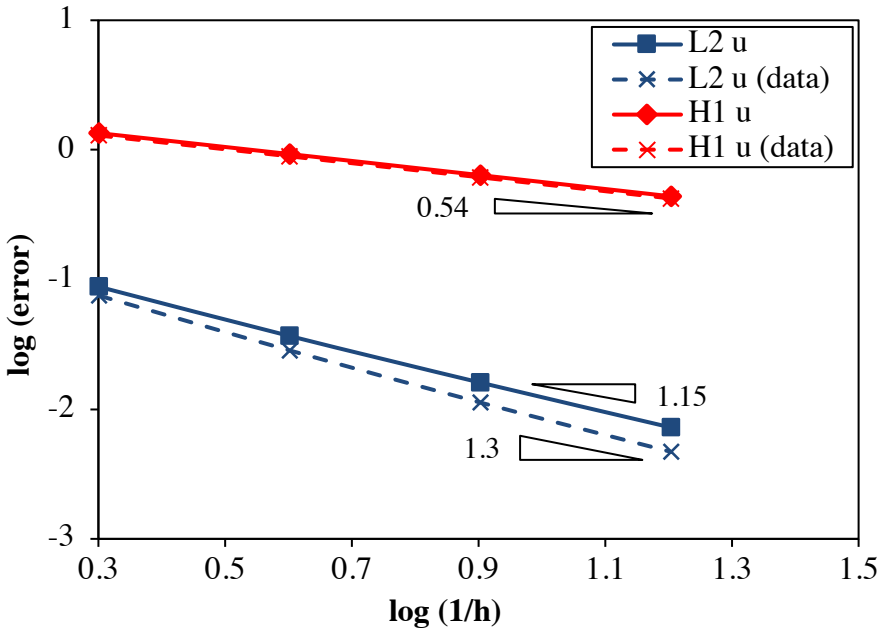
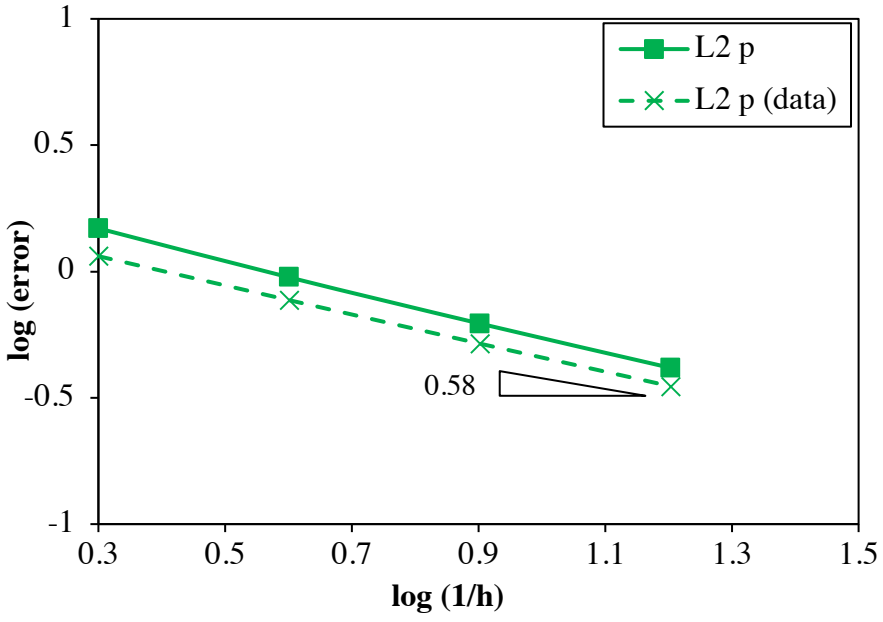


Figure 20: Schematic of Stokes flow test case (a) problem description (b) coarsest non-conforming mesh



(a)



(b)

Figure 21: Convergence rates of L_2 norm and H^1 seminorm for (a) velocity field (b) pressure field

Finally, a qualitaive assessment of the solution fields is presented through the contour plots in Figure 22. The effectiveness of the interface operator in Section 5.2 can clearly be seen here as the

velocity fields which are very smooth across the interface. The discontinuity in the pressure field can be seen in Figure 22(c) which causes disruption around the the re-entrant corner. Even with presence of a strong disconuity in the presssue field, we observe an improvement in the accuracy of the pressure field with the introduction of data (Figure 21(b)) which is remarkable.

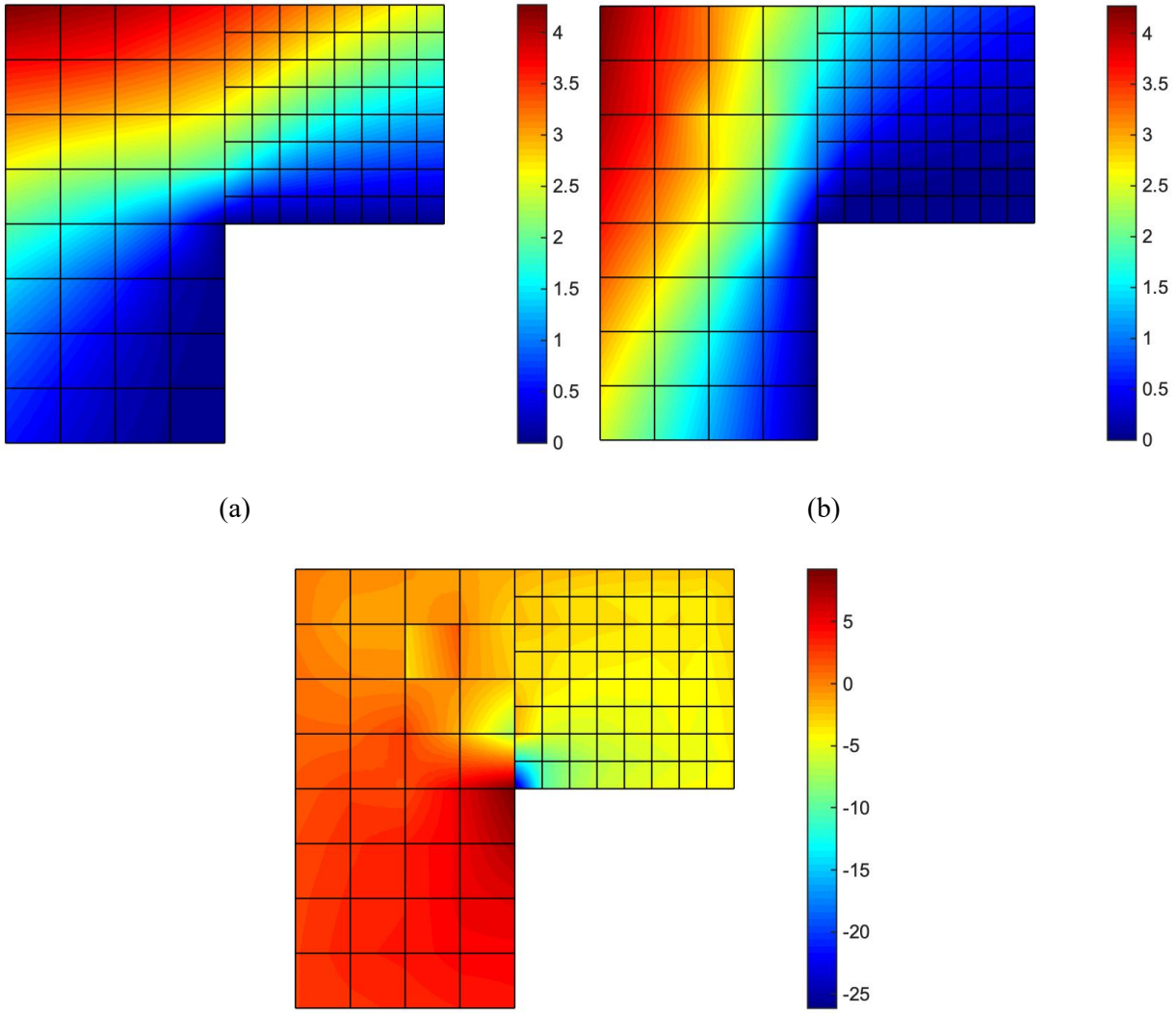


Figure 22: Contour plots for (a) velocity v_x field (b) velocity v_y field (c) pressure field

7.5 Stokes-Darcy Coupled System

As another example of the case of non-overlapping solution decomposition, we consider a coupled Stokes-Darcy problem posed over a bi-unit square domain. The Darcy model occupies the region $\Omega^{(1)} = (0,1)\times(0,\frac{1}{2})$, and the Stokes model occupies the region $\Omega^{(2)} = (0,1)\times(\frac{1}{2},1)$. We devise an exact solution similar to those proposed in [54] that satisfies the interfacial conditions between the two regions:

$$v_x^{(1)} = \sin\left(\frac{x}{G}+\omega\right)e^{y/G} + \omega\sin(\omega x) \tag{174}$$

$$v_y^{(1)} = -\cos\left(\frac{x}{G}+\omega\right)e^{y/G} \tag{175}$$

$$p^{(1)} = \frac{\mu G}{\kappa}\cos\left(\frac{x}{G}+\omega\right)e^{y/G} + \frac{\mu}{\kappa}\cos(\omega x) \tag{176}$$

$$v_x^{(2)} = \sin\left(\frac{x}{G}+\omega\right)e^{y/G} \tag{177}$$

$$v_y^{(2)} = -\cos\left(\frac{x}{G}+\omega\right)e^{y/G} \tag{178}$$

$$p^{(2)} = y + e^{1/2G}\left[\frac{\mu G}{\kappa} - \frac{2\mu}{G}\right]\cos\left(\frac{x}{G}+\omega\right) + \frac{2\mu y}{\kappa}\cos(\omega x) - \frac{1}{2} \tag{179}$$

where the parameter $G = \sqrt{\kappa}/\alpha_o$. The associated body force terms $\rho \boldsymbol{g}_G / g_c, \varphi_G, \boldsymbol{f}_L$ are obtained by substituting these analytical expressions into the governing equations (108), (109), and (133)

(note that $\boldsymbol{g}_G = \mathbf{0}$ by design). Notice that both the tangential component of velocity and the pressure field are discontinuous across $\Gamma_{\mathrm{I},(21)} = (0,1) \times \{\frac{1}{2}\}$ while the normal component of velocity and the stress are continuous. Additionally, careful attention is paid to satisfying the Beavers-Joseph-Saffman law [48] at the lower boundary of the Stokes region. For the numerical simulations that follow, DG interfaces are also inserted to partition the two model regions to allow nonconforming meshes, namely $\Gamma_{\mathrm{I},(11)} = \{\frac{1}{2}\} \times (0,\frac{1}{2})$ and $\Gamma_{\mathrm{I},(22)} = \{\frac{1}{2}\} \times (\frac{1}{2},1)$ (see Figure 23(a)). As boundary conditions, the normal velocity $\boldsymbol{v}^{(1)} \cdot \boldsymbol{n}$ is prescribed on the left edge of the Darcy region along with flux conditions consistent with the pressure $p^{(1)}$ on the lower and right edges. The velocity field $\boldsymbol{v}^{(2)}$ is prescribed on the left edge of the Stokes region along with flux conditions on the upper and right edges. Lastly, the material properties are taken as $\kappa = 1$, $\mu = 1$, $\alpha_o = 0.5$, and $\omega = 1.05$.

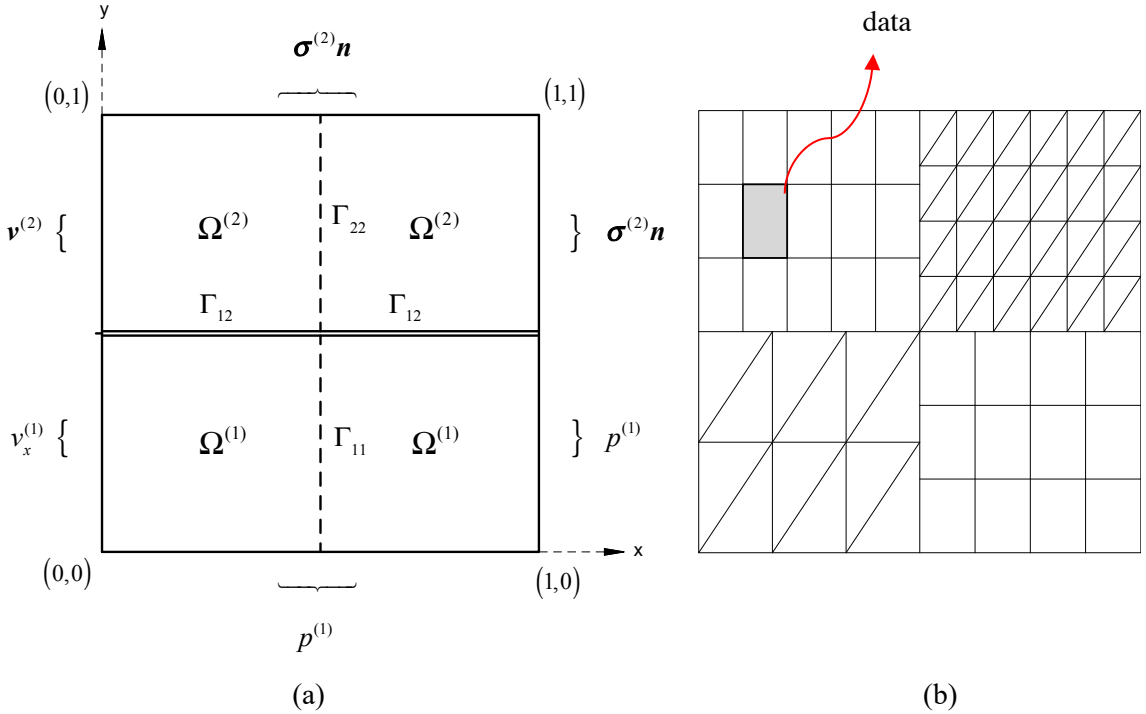
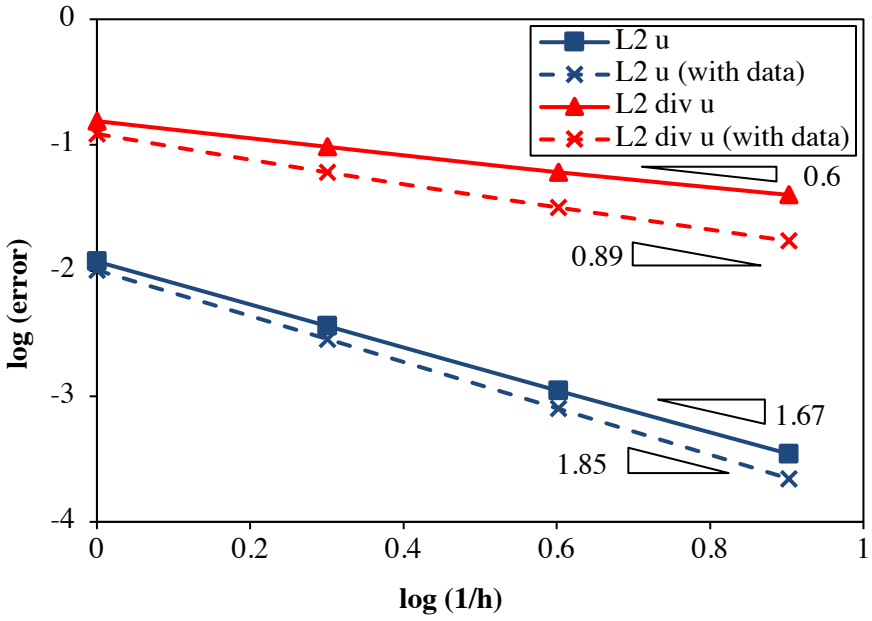
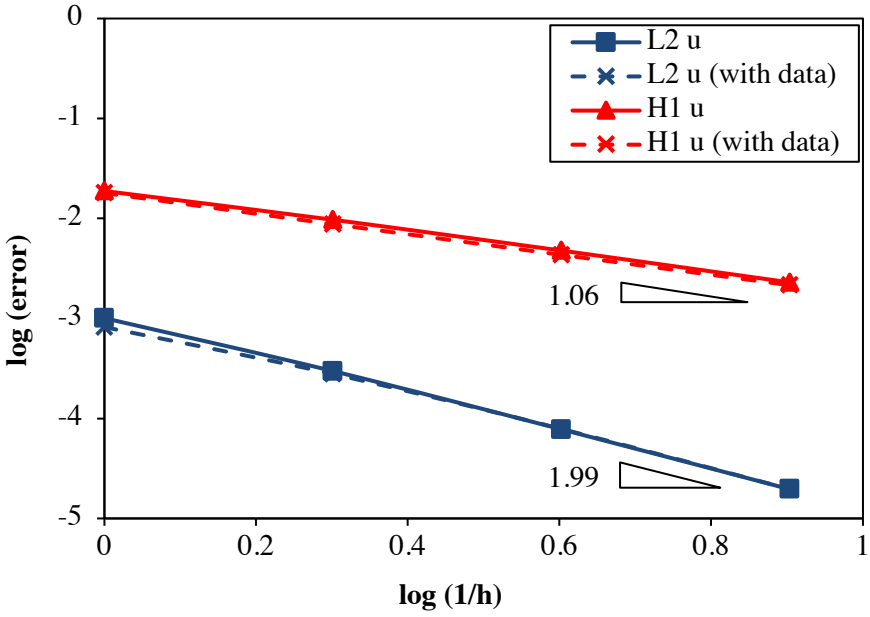


Figure 23: Schematic of Stokes-Darcy coupled system (a) problem description (b) corsest discretization

The convergence of the numerical error between the numerical and exact solutions is provided in Figure 24(a) and (b) separately for the Darcy region $\Omega^{(1)}$ and the Stokes region $\Omega^{(2)}$, respectively. The meshes employed have 87, 384, 1536, and 6144 linear triangular and quadrilateral elements. For the Darcy region, the convergence rates of the velocity L_2 norm and H^1 seminorm exhibits a consistent trend of 1.67 and 0.6, respectively. By comparison, the velocity error norms in the Stokes region attain the theoretical rates of 2 and 1 according to the analysis conducted in [26] for a similar formulation. Then, we introduce data as the exact solution of the velocity field at the nodes inside the patch shown in Figure 23(b) and use the formulation in Chapter 6 to run the data-driven problem. The error convergence rates in the Darcy region improve to 1.85 and 0.89 and the absolute values of the L_2 and H^1 error norms also decrease. However, for the Stokes region, we do not see any improvement in either the error norms or the convergence rates which were already optimal without the data. The reason might be that the numerical solution quickly approaches the exact solution and the impact of data is minimal. This trend is similar to the one seen for the pressure field in example 7.3.



(a)

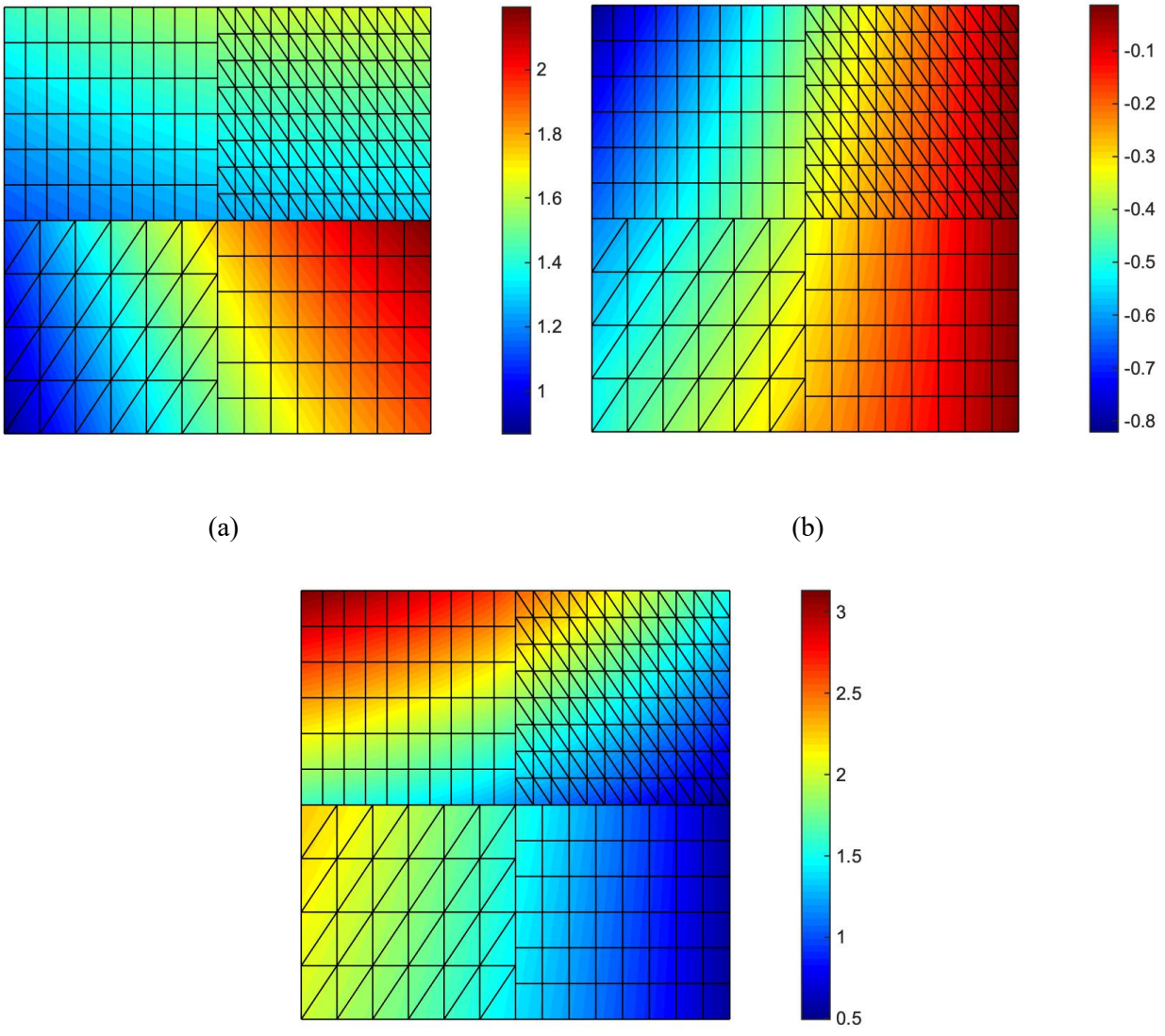


(b)

Figure 24: Convergence rates of L_2 norm and H^1 seminorm for (a) Darcy region (b) Stokes region

Contour plots of the solution obtained on the mesh with 384 elements are presented in Figure 25(a)–(c). We remark that the interfaces between the triangular and quadrilateral elements of each

modeled region are nearly invisible. The major features of the solution field are well resolved; namely, the vertical component of velocity is nearly continuous between the two modeling regions, and the jumps in the pressure and the horizontal velcoity fields are controlled across the Darcy-Stokes interface.



(a) (b)

(c)

Figure 25: Contour plots for (a) velocity v_x field (b) velocity v_y field (c) pressure field

CHAPTER 8: CONCLUSION

A stabilized interface coupling method has been employed for embedded discrete as well as distributed data in the numerical simulation of the problem. The stabilized interface method was earlier developed for the coupling of discrete interfaces that arise due to nonconforming meshes, material mismatch, disparate governing PDEs, and hierarchal physical models with physics of increasing complexity on adjacent physics subdomains. In the VMDG method the interface coupling operators are derived through modeling of fine scales and exploiting the presence of fine-scale in the coarse-scale continuity equations. Physics based data is subsequently introduced in the domain as variational residual at the location of internal discontinuity, and it exploits the least-squares from of the interface coupling terms. The least-squares form which has an intrinsic structure of a linear regression type function drives the numerical solution towards target values. In the numerical test problems, exact solution of the underlying mathematical problem is considered as high-fidelity data, and objective is to see that when this high-fidelity information on the physical behavior of the system is furnished to the discrete problem which is otherwise driven by the boundary conditions and body forces, how does it drive the discrete system to higher spatial accuracy. A range of numerical tests were designed to investigate this point. Test cases in fluids and in solids show improvement in the computed solution which is attributed to variational embedding of the physics-based data. Elasticity problem with stress singularity seems to show significant improvement in accuracy of the solution, while in Stokes flow for a similar test case with weak singularity, the improvement was not substantial. However, an important point to notice is that in both the cases variational embedding of data did not pollute the accuracy of the solution in the surrounding neighborhood of the interface. The future work will include extending the role of data via the fine scale modeling feature facilitated by the Variational Multiscale method.

REFERENCES

1. R. Agarwal, V. Dhar, Big data, data science, and analytics: The opportunity and challenge for IS research, Information Systems Research 25(3) (2014) 443–448.
2. M. Ainsworth, J.T. Oden, A posteriori error estimation in finite element analysis, Computer Methods in Applied Mechanics and Engineering 142 (1997) 1–88.
3. I.M. Babuška, T. Strouboulis, A. Mathur, C.S. Upadhyay, Pollution-error in the h-version of the finite-element method and the local quality of a-posteriori error estimators, Finite Elements in Analysis and Design 17 (1994) 273–321.
4. C. Baiocchi, F. Brezzi, L.P. Franca, Virtual bubbles and the Galerkin/least-squares method, Computer Methods in Applied Mechanics and Engineering 105 (1993) 125–142.
5. H.J.C. Barbosa, T.J.R. Hughes, The finite element method with Lagrange multipliers on the boundary: circumventing the Babuska–Brezzi condition, Computer Methods in Applied Mechanics and Engineering 85 (1991) 109–128.
6. F. Brezzi, M.-O. Bristeau, L.P. Franca, M. Mallet, G. Rog., A relationship between stabilized finite element methods and the Galerkin method with bubble functions, Computer Methods in Applied Mechanics and Engineering 96 (1992) 117–129.
7. F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Methods, Springer Series in Computational Mathematics. Vol. 15 Springer: New York (1991).
8. F. Brezzi, L.P. Franca, T.J.R. Hughes, A. Russo, $b = \int g$,. Computer Methods in Applied Mechanics and Engineering 145(3-4) (1997) 329-339.
9. F. Brezzi, T.J.R. Hughes, L.D. Marini, A. Masud, Mixed Discontinuous Galerkin methods for Darcy flow, SIAM J. Scientific Computing 22 (2005) 119-145.

10. F. Brezzi, A. Russo, Choosing bubbles for advection-diffusion problems, Mathematical Models and Methods in Applied Sciences 4(04) (1994) 571–587.
11. G. Capuano, J.J. Rimoli, Smart finite elements: A novel machine learning application, Computer Methods in Applied Mechanics and Engineering, 345 (2019) 363-381.
12. A.L.G.A. Coutinho, C.M. Dias, J.L.D. Alves, L. Landau, A.F.D. Loula, S.M.C. Malta, R.G.S. Castro, E.L.M. Garcia, Stabilized methods and post-processing techniques for Darcy flow and related problems, Computer Methods in Applied Mechanics and Engineering 193(15-16) (2004) 1421–1436.
13. Y. Efendiev, T.Y. Hou, V. Ginting, Multiscale finite element methods for nonlinear problems and their applications, Communications in Math and Science 2(4) (2004) 553-589.
14. K. Eriksson, D. Estep, P. Hansbo, C. Johnson, Introduction to adaptive methods for differential equations, Acta Numerica 4 (1995) 105–158.
15. C. Farhat, I. Harari, L.P. Franca, The discontinuous enrichment method, Computer Methods in Applied Mechanics and Engineering 190(48) (2001) 6455–6479.
16. T.P. Fries, T. Belytschko. The extended/generalized finite element method: an overview of the method and its applications, International Journal of Numerical Methods in Engineering 84(3) (2010) 253-304.
17. C. Gao, X.J. Wu, Kernel support tensor regression, Procedia Engineering 29 (2012) 3986-3990.
18. V. Girault, G. Kanschat, B. Rivière, On the coupling of incompressible Stokes or Navier–Stokes and Darcy flows through porous media. Modelling and Simulation in Fluid Dynamics in Porous Media, Springer Proceedings in Mathematics & Statistics 28 (2013) 1–25.

19. G. Hauke, M. Doweidar, M. Miana, The multiscale approach to error estimation and adaptivity, Computer Methods in Applied Mechanics and Engineering 195 (2006) 1573–1593.
20. G. Hauke, D. Fuster, M. Doweidar, Variational Multiscale a-posteriori error estimation for multi-dimensional transport problems, Computer Methods in Applied Mechanics and Engineering 197 (2008) 2701–2718.
21. G. Hlepas, T.J. Truster, A. Masud, A heterogeneous modeling method for porous media flows, International Journal for Numerical Methods in Fluids. 75(7) (2014) 487-518.
22. A. Huerta, P. Diez, Error estimation including pollution assessment for nonlinear finite element analysis, Computer Methods in Applied Mechanics and Engineering 181 (2000) 21–41.
23. T.J.R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Prentice-Hall: Englewood Cliffs, NJ, 1987; Dover edition, 2000.
24. T.J.R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Computer Methods in Applied Mechanics and Engineering 127 (1995) 387-401.
25. T.J.R. Hughes, G.R. Fèijo, L. Mazzei, J. Quincy, The variational multiscale method–a paradigm for computational mechanics, Computer Methods in Applied Mechanics and Engineering 166(1–2) (1998) 3–24.
26. T.J.R. Hughes, L.P. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuska–Brezzi condition: a stable Petrov–Galerkin formulation of the stokes problem accommodating equal-order interpolations, Computer Methods in Applied Mechanics and Engineering 59 (1986) 85–99.
27. T.J.R. Hughes, A. Masud, J. Wan, A discontinuous-Galerkin finite element method for Darcy flow, Computer Methods in Applied Mechanics and Engineering 195 (2006) 3347–3381.

28. R. Ibanez, E. Abisset-Chavanne, J.V. Aguado, D. Gonzalez, E. Cueto, F. Chinesta, A manifold learning approach to data-driven computational elasticity and inelasticity, Archives of Computational Methods in Engineering 25(1) (2018) 47–57.
29. T. Kirchdoerfer, M. Ortiz, Data-driven computational mechanics, Computer Methods in Applied Mechanics and Engineering 304 (2016) 81–101.
30. P.S. Koutsourelakis, Stochastic upscaling in solid mechanics: An excercise in machine learning, Journal of Computational Physics 226(1) (2007) 301-325.
31. M. Larson, A. Målqvist, Adaptive variational multiscale methods based on a posteriori error estimation: duality techniques for elliptic problems, in: B. Engquist, P. L.tstedt, O. Runborg (Eds.), Lecture Notes in: Computational Science and Engineering 44 Springer Berlin (2005) 181–193.
32. Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521(7553) (2015) 436.
33. A. Masud, A stabilized mixed finite element method for Darcy–Stokes flow, International Journal for Numerical Methods in Fluids 54(3) (2007) 4341–4370.
34. A. Masud, L.P. Franca, A hierarchical multiscale framework for problems with multiscale source terms, Computer Methods in Applied Mechanics and Engineering 197 (2008) 2692-2700.
35. A. Masud, R. Kannan, A multiscale framework for computational nanomechanics: Application to carbon nanotubes, International Journal for Numerical Methods in Engineering 78 (2009) 863-882.
36. A. Masud, R.A. Khurram, A multiscale finite element method for the incompressible Navier–Stokes equations, Computer Methods in Applied Mechanics and Engineering 195 (2006) 1750-1777.

37. A. Masud, G. Scovazzi, A heterogeneous multiscale modeling framework for hierarchical systems of partial differential equations, International Journal for Numerical Methods in Fluids 65 (2011) 28–42.
38. A. Masud, T.J. Truster, L. Bergman, A unified formulation for interface coupling and frictional contact modeling with embedded error estimation, International Journal of Numerical Methods in Engineering 92 (2012) 141–177.
39. A. Masud, T.J. Truster, L. Bergman, A variational multiscale a posteriori error estimation method for mixed form of nearly incompressible elasticity, Computer Methods in Applied Mechanics and Engineering 200 (2011) 3453–3481.
40. A. Masud, K. Xia, A stabilized mixed finite element method for nearly incompressible elasticity, Journal of Applied Mechanics 72 (2005) 711–720.
41. K.B. Nakshatrala, A. Masud, K.D. Hjelmstad, On finite element formulations for nearly incompressible linear elasticity, Computational Mechanics 41 (2008) 547–561.
42. J. Nitsche, Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. InAbhandlungen aus dem mathematischen Seminar der Universität Hamburg 36 (1971) 9–15.
43. A.K. Noor, Global-local methodologies and their application to nonlinear analysis, Finite Element in Analysis and Design 2(4) (1986) 333–346.
44. J.T. Oden, Y. Feng, Local and pollution error estimation for finite element approximations of elliptic boundary value problems, Journal of Computational and Applied Mathematics 74 (1996) 245–293.
45. A. Oishi, G. Yagawa, Computational mechanics enhanced by deep learning, Computer Methods in Applied Mechanics and Engineering 327 (2017) 327-351.

46. P.A. Raviart, J.M. Thomas, A mixed finite element method for second order elliptic problems. In Mathematical Aspects of the Finite Element Method. Lecture Notes in Mathematics, vol. 606, Galligani I, Magenes E (eds.). Springer: New York, 1977.
47. A. Russo, A posteriori error estimators via bubble functions, Mathematical Models and Methods in Applied Sciences 6 (1996) 33–41.
48. P. Saffman, On the boundary condition at the surface of a porous media, Studies in Applied Mathematics 50 (1971) 93–101.
49. S. Shalev-Shwartz, S. Ben-David, Understanding machine learning: From theory to algorithms, Cambridge University Press (2014).
50. S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, McGraw-Hill Book Company, 1970.
51. T.J. Truster, A variational multiscale computational framework for nonlinear interfacial solid mechanics, PhD Dissertation University of Illinois at Urbana-Champaign; Urbana (2013).
52. T.J. Truster, M. Eriten, A.A. Polycarpou, L.A. Bergman, A. Masud, Stabilized interface methods for mechanical joints: physics-based models and variationally consistent embedding, International Journal of Solids and Structures 50 (2013) 2132–2150.
53. T.J. Truster, A. Masud, Primal interface formulation for coupling multiple PDE: A consistent derivation through the variational multiscale method, Computer Methods in Applied Mechanics and Engineering 268 (2014) 194–224.
54. D. Vassilev, I. Yotov, Coupling Stokes–Darcy flow with transport, SIAM Journal of Scientific Computing 35(5) (2009) 3661–3684.
55. R. Verfürth, A posteriori error estimators for the Stokes equations, Numerische Mathematik 55 (1989) 309–325.