Chapter 10

A structure-preserving domain decomposition method for data-driven modeling☆

Shuai Jiang[a], **Jonas Actor**[a], **Scott Roberts**[b], **and Nathaniel Trask**[c,*]

[a]*Center for Computing Research, Sandia National Laboratories, Albuquerque, NM, United States,*
[b]*Engineering Sciences Center, Sandia National Laboratories, Albuquerque, NM, United States,*
[c]*Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia, PA, United States*
[*]*Corresponding author: e-mail address: ntrask@seas.upenn.edu*

Contents

☆ **Funding:** This article has been authored by an employee of National Technology & Engineering Solutions of Sandia, LLC under Contract No. DE-NA0003525 with the U.S. Department of Energy (DOE). The employee owns all right, title and interest in and to the article and is solely responsible for its contents. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this article or allow others to do so, for United States Government purposes. The DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan https://www.energy.gov/downloads/doe-public-access-plan.

Abstract
We present a domain decomposition strategy for developing structure-preserving finite element discretizations from data when exact governing equations are unknown. On subdomains, trainable Whitney form elements are used to identify structure-preserving models from data, providing a Dirichlet-to-Neumann map which may be used to globally construct a mortar method. The reduced-order local elements may be trained offline to reproduce high-fidelity Dirichlet data in cases where first principles model derivation is either intractable, unknown, or computationally prohibitive. In such cases, particular care must be taken to preserve structure on both local and mortar levels without knowledge of the governing equations, as well as to ensure well-posedness and stability of the resulting monolithic data-driven system. This strategy provides a flexible means of both scaling to large systems and treating complex geometries, and is particularly attractive for multiscale problems with complex microstructure geometry. While consistency is traditionally obtained in finite element methods via quasioptimality results and the Bramble-Hilbert lemma as the local element diameter $h \to 0$, our analysis establishes notions of accuracy and stability for finite h with accuracy coming from matching data. Numerical experiments and analysis establish properties for $H(\mathrm{div})$ problems in small data limits ($\mathcal{O}(1)$ reference solutions).

Keywords
Structure preservation, Mortar method, Domain decomposition, Whitney forms, Model reduction, Data-driven modelling, Scientific machine learning

MSC Codes
68T01, 65N30, 65N55

1 Introduction

We consider the problem of identifying a model from data when the governing equations are unknown, but the conservation structure is known. Namely, one may know that fluxes associated with mass, momentum, or energy are conserved, but be unable to derive specific expressions for those fluxes.

We assume a class of models of the form

$$\begin{aligned} \nabla \cdot \boldsymbol{u} &= -f && \text{on } \Omega, \\ \boldsymbol{u} &= h(p;\theta) && \text{on } \Omega, \\ p &= g && \text{on } \partial\Omega \end{aligned} \tag{1.1}$$

where $\Omega \in \mathbb{R}^d$ is a Lipschitz domain, $f \in L^2(\Omega)$ forcing term, g Dirichlet data, and h a closure for the flux of unknown functional form approximated by a family of nonparametric regressors parameterized by θ. We demonstrate on $\Omega \in \mathbb{R}^2$ exclusively, but the techniques shown here generalize to higher dimensions and arbitrary manifolds. For this class of problems, data is provided in the form

$\mathcal{D} = \{(\boldsymbol{u}_k, f_k, h_k, g_k)\}_{k=1}^{N}$ and one identifies parameters θ which minimize error in a suitable norm, providing a model which may generalize by solving for choices of f and h outside the training set.

By casting data-driven modeling in such a *structure-preserving* framework, one aims to identify a model which balances a trade-off between rigorous preservation of physical/algebraic/stability structure while maintaining "black-box" approximation of as large a class of models as possible. This lies on a spectrum of methods in the literature spanning a trade-off between expressivity and exploitable structure. For example, operator regression methods aim to directly identify a solution map $(f, h) \to \boldsymbol{u}$ via interpolation in unconstrained Hilbert spaces (high expressivity), while PDE-constrained optimization (Biegler et al., 2003; Hinze et al., 2008) assumes a known functional form for h which requires only estimation of material parameters (highly structured with simplified analysis).

For the purposes of this work we consider elliptic systems of $H(\text{div})$-type where structure-preservation amounts to preserving notions of flux continuity. In the literature, preservation of other types of structure is a key challenge for data-driven models: gauge invariances associated with nontrivial null-spaces (Trask et al., 2022), geometric structure associated with bracket dynamics (Gruber et al., 2023; Greydanus et al., 2019; Desai et al., 2021; Hernández et al., 2021), group equivariance (Bergomi et al., 2019; Villar et al., 2021) and other structures (Celledoni et al., 2021). Many of these approaches aim to enforce the invariances by construction rather than rely on data or training to "learn" them, allowing better performance in small-data limits and improved theoretical properties.

In our previous works (Actor et al., 2024; Trask et al., 2022), we have developed structure-preserving machine learning frameworks generalizing the discrete exterior calculus (DEC) and finite element exterior calculus (FEEC) (see Subsection 2.1). Both frameworks pose the learning of physics as identifying maps between cochains associated with a de Rham complex, and provide a number of desirable theoretical guarantees: preservation of exact sequence structure (e.g. $\nabla \cdot (\nabla \times) = 0$), exact local conservation of generalized fluxes, an exact Hodge decomposition, a Lax-Milgram stability theory for Hodge Laplacians, well-posedness theory for nonlinear problems, and a framework for treating problems with nontrivial null-spaces (e.g. electromagnetism). In the FEEC setting, a Dirichlet-to-Neumann map prescribing the exchange of generalized fluxes between subdomains is expressed in terms of parameterized Whitney forms, allowing the machine learning of geometric control volumes which optimally admit integral balance laws. While effective for providing rigorous structure-preservation, the scheme provides poor computational scaling whereby the number of degrees of freedom scale as $\mathcal{O}(N^k)$, where N is the number of partitions and k is the order of the Whitney form.

The current work applies a divide-and-conquer strategy to mitigate this by partitioning the domain into disjoint, nonoverlapping subdomains $\Omega = \cup_i \Omega_i$, whose exact specifications will be discussed later, and seeks local models re-

stricted to each Ω_i of the form

$$\begin{aligned} \nabla \cdot \boldsymbol{u}_i &= -f_i, \\ \boldsymbol{u}_i &= h(p_i; \theta_i), \\ p_i &= g_i \qquad\qquad \text{on } \partial\Omega_i, \end{aligned} \tag{1.2}$$

with the subscript $\cdot_i$ denoting appropriate restrictions of fields to Ω_i. The framework for regressing local models is introduced in Section 3. To train subdomain models, we can perform offline training over data $\mathcal{D}_i = \left\{(u_{i,k}, f_{i,k}, h_{i,k}, g_{i,k})\right\}_{k=1}^{N_i}$. This can be obtained either by taking the restriction of global data onto the subdomain ($g_i = p|_{\partial\Omega}$), or by performing simulations directly on each subdomain to identify the local response to a representative mortar space (e.g. $g_i \in \mathbb{P}_m(\partial\Omega_i)$ the space of m^{th}-order polynomials). After obtaining local models, a mortar method is presented in Section 4 which is used to assemble local models into a global model on Ω.

For this data-driven mortar strategy, we impose two desired requirements:

1. **R1: Preservation of structure across both scales:** For the $H(\text{div})$ problems under consideration, the Whitney form construction admits interpretation as an integral balance law where fluxes are discretely treated as equal and opposite, providing a local conservation principle on each subdomain Ω_i. We require that the mortar formulation be compatible with this, so that when local elements are stitched together through the mortar we preserve conservation globally on Ω.
2. **R2: Stability of error at global scale:** If, during pretraining, local models may be obtained to a given optimization error, we would like to quantify the error induced at a global level by the coupling process. Ideally this would be bound by a constant independent of the number of subdomains, so that the global error remains comparable to that of the locally trained models as many elements are coupled together and performance does not degenerate in the limit of many data-driven elements.

We demonstrate both requirements either in analytical proofs in Section 4, or via numerical example in Section 5. Finally, the technical proofs and more details regarding training are shown in Appendix 10.A.

2 Relation to previous work

The proposed strategy exploits a connection to structure-preserving PDE discretization to ensure that physics are enforced by construction, rather than via the penalty formulation typically pursued in the physics-informed machine learning literature. We summarize the relationship between this approach and the literature, as well as how our strategy relates to classical domain decomposition methods.

2.1 Data-driven DEC/FEEC and Dirichlet-to-Neumann maps

In traditional numerical analysis the discrete exterior calculus (DEC) is a framework for constructing and analyzing staggered finite volume schemes (Hirani, 2003; Nicolaides, 1992). The generalized Stokes theorem is used to define discrete vector calculus operators (e.g. grad/curl/div) which map between differential forms on a pair of primal/dual computational meshes. The finite element exterior calculus (FEEC) generalizes DEC by constructing finite element spaces which interpolate differential forms and provides variational extensions (Arnold, 2018).

In the data-driven exterior calculus (DDEC) (Trask et al., 2022), DEC operators are parameterized in a manner allowing the learning of well-posed models on graphs, where data is used to identify the inner-product associated with codifferential operators. In Actor et al. (2024), it was shown that a family of data-driven Whitney forms may be constructed from parameterized partitions-of-unity (POUs). The Whitney forms admit a de Rham complex which encodes POU geometry as differentiable control volumes and their higher order boundaries (faces/edges/etc) without reference to a traditional mesh. An inner-product is induced by the geometry of the control volumes, supporting the discovery of models in terms of control volume balances. This allows a data-driven FEEC extension of DDEC which we use extensively in this work. Furthermore, by posing integral balances as relationships between domains and fluxes on their boundaries, we work with degrees of freedom which naturally conform to the trace spaces necessary for a mortar strategy.

2.2 Structure-preserving ML vs. physics-informed ML

In the recent scientific machine learning literature, physics-informed methods broadly encompass frameworks where physical constraints are incorporated by adding (typically collocation) residuals to a loss function as a Tikhonov regularization with a penalty parameter (Cai et al., 2021). This technique is simple to implement and, when used together with automatic differentiation, admits a simple treatment of inverse problems, discovery of "missing physics" or closures (Karniadakis et al., 2021; Patel et al., 2022), and uncertainty quantification (Yang and Perdikaris, 2019; Zhang et al., 2019).

The flexibility of the framework comes at the expense of solving a multiobjective optimization problem whereby the physics residual must be empirically weighted against the data loss, and can only be enforced to within optimization error (Wang et al., 2021). For certain classes of problems it is necessary to enforce physics to machine precision to obtain qualitatively correct answers; e.g. subsurface transport and lubrication flows depend crucially on exact conservation of mass (Trask et al., 2018), while electromagnetic problems which fail to provide an exactly divergence-free magnetic field predict qualitatively incorrect spectra (Arnold, 2018). In the context of physics-informed learning, some works have pursued a penalty-based domain decomposition strategy with the

goal of efficient distributed computation and more flexibility in neural network approximation (Jagtap and Karniadakis, 2021). While effective, the collocation scheme and penalty formulation complicate analysis and preclude exact conservation, respectively. Because the desired conservation structure only holds to within optimization error, penalization may be insufficient for certain classes of applications.

2.3 Choice of mortar scheme

Domain decomposition is a mature field, with many established options for how to couple solutions across arbitrary finite element subdomains (Toselli and Widlund, 2004; Smith, 1997). Representative rigorous methods range from (e.g. finite element tearing and interconnecting (FETI) Farhat et al. (2001), mortar methods Bernardi et al. (1993), and hybridizable discontinuous Galerkin methods Cockburn et al. (2009)) impose continuity of fluxes and state at subdomain interfaces either strongly via Lagrange multipliers or weakly by using Nietsche's trick to introduce a variational penalty.

For the div-grad problem, there is also a choice of working in either H^1- or $H(\text{div})$-conforming spaces (e.g. $\mathbb{P}_1$/Nedelec or Raviart-Thomas/$\mathbb{P}_0$ mixed spaces), and whether one chooses to apply a mortar on the state or flux variables. Working in $H(\text{div})$ is perhaps most natural, as the mortar space admits interpretation as a conservative flux that trivially preserves conservation structure (Arbogast et al., 2007). However, this requires working with d- and $(d-1)$-dimensional Whitney forms. Our Whitney form construction scales with computational complexity $\mathcal{O}(N^k)$, where N is the dimension of 0^{th}-order Whitney forms and k is the maximal order Whitney form. It is therefore preferable to exploit primal/dual structure and work in H^1, meaning only 0^{th}- and 1^{st}-order Whitney forms are used. This forces us to adopt an H^1 domain decomposition strategy similar to that developed by Glowinski and Wheeler (1987, §7). Further extensions are needed to easily incorporate and analyze the case where data-driven FEEC elements are used as the local solvers.

3 Local learning of Whitney form elements

For brevity, we discuss only the fundamental aspects of data-driven DEC/FEEC necessary to describe the local element construction. For a complete exposition we direct readers to references for: data-driven exterior calculus (Trask et al., 2022), data-driven finite element exterior calculus (Actor et al., 2024), classical finite element exterior calculus for forward simulation (Arnold, 2018), and Whitney forms (Gillette et al., 2016).

Given a compact domain $\omega \in \mathbb{R}^2$ with finite open cover $\{U_i\}_{i=1}^N$, a *partition of unity* (POU) is a collection of functions $\phi_i : \omega \to [0,1]$ such that $\phi_i(\boldsymbol{x}) \geq 0$, $\text{supp}(\phi_i) \subseteq U_i$, $\phi_i < \infty$ and $\sum_i \phi_i(\boldsymbol{x}) = 1$ for all $\boldsymbol{x} \in \omega$. We assume access to

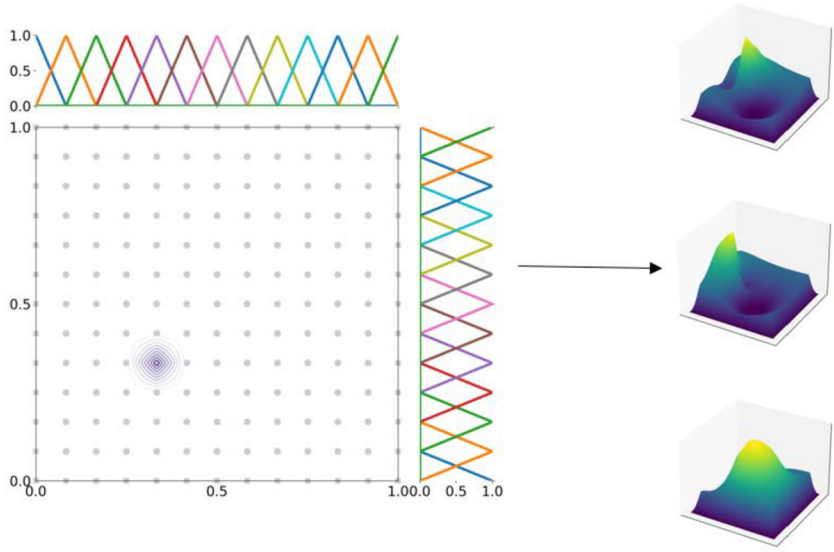


FIGURE 1 To construct a PPOU, we first consider an underlying tensor product grid of B-splines with trainable vertex locations. By taking a trainable convex combination of these shape functions, we arrive at more complex geometries. Noting that B-splines form a partition of unity, and that partitions of unity are closed under convex combination, this process provides a trainable partition of unity which may be integrated exactly via a pull-back onto the fine grid. For purposes of illustration, the underlying tensor product is shown to be uniform in the figure, but they are allowed to shift in the general case.

a *parameterized POU* (PPOU) $\{\phi_i(\boldsymbol{x};\theta)\}_{i=1}^N$, which is continuous with respect to a parameter θ.

To construct Whitney forms, any trainable PPOU may be used, although in this work we adopt the same used in Actor et al. (2024). Starting with tensor-product B-splines on the unit domain, we refer to trainable vertex locations as *fine-scale nodes/knots*. To approximate complex geometries, we consider a coarsening via convex combinations of the knots into our ultimate PPOU, $\{\phi_i(\boldsymbol{x};\theta)\}_{i=1}^N$ where θ denotes parameters corresponding to both knot locations and trainable entries of the convex combination tensor; see Fig. 1 for an illustrative figure of this process.

In Actor et al. (2024), the tensor-product grid points are parameterized using the distances between the grid points to avoid inversion of elements. In particular, we can define the grid points in one dimension $\{t_i\}_{i=0}^n$ with $t_0 = 0$ and $t_1 = 1$ by parameterizing

$$\sigma(\delta)_i = t_{i+1} - t_i, \qquad i \in \{0, \dots, n-1\} \tag{3.1}$$

where δ_i is a trainable parameter, and σ is a sigmoid activation enforcing positivity. To parameterize a map of convex combination of knots, we consider a

trainable two-tensor with softmax activation applied to each row; for details we refer to Actor et al. (2024). In what follows we adopt the simplified notation $\phi_i(\boldsymbol{x};\theta) = \phi_i$.

We construct finite element spaces consisting of the 0^{th}-, 1^{st}- and 2^{nd}-order Whitney forms from ϕ_i:

$$\begin{aligned} V^0 &:= \operatorname{span}\left\{\phi_i \mid 1 \le i \le N\right\}, \\ V^1 &:= \operatorname{span}\left\{\phi_i \nabla\phi_j - \phi_j \nabla\phi_i \mid 1 \le i, j \le N\right\}, \\ V^2 &:= \operatorname{span}\left\{\phi_i \nabla\phi_j \times \nabla\phi_k - \phi_j \nabla\phi_i \times \nabla\phi_k - \phi_k \nabla\phi_j \times \nabla\phi_i \mid 1 \le i, j, k \le N\right\}, \end{aligned} \tag{3.2}$$

adopting the notation $\psi_{j_1,\ldots,j_K} \in V^{k-1}$, to identify elements of spaces by their constituent 0-forms (e.g. $\psi_{ij} \in V^1$). As shown in Actor et al. (2024), the tensor used to parameterize convex combinations of B-splines may be manipulated to obtain modifications of these spaces with zero trace

$$V_0^k := \left\{ u \in V^k \,\middle|\, u|_{\partial\omega} = 0 \right\}. \tag{3.3}$$

Consider now the variational form of divergence $(q, \nabla\cdot\boldsymbol{u})$ and curl $(\boldsymbol{v}, \nabla\times\boldsymbol{w})$, where $q \in V^0$, $\boldsymbol{u} \in V_0^1$, $\boldsymbol{v} \in V^1$, and $\boldsymbol{w} \in V_0^2$. After integration by parts, Whitney forms induce the following discrete vector calculus operators (Actor et al., 2024, §3)

$$\begin{aligned} (\mathsf{DIV})_{i,(ab)} &:= (\psi_{ab}, -\nabla\psi_i) = \sum_{j\neq i} (\psi_{ab}, \psi_{ij}), \\ (\mathsf{CURL})_{(ij),(abc)} &:= (\psi_{abc}, \nabla\times\psi_{ij}) = 2\sum_{k\neq i,j} (\psi_{abc}.\psi_{ijk}). \end{aligned} \tag{3.4}$$

These discrete exterior derivatives maintain a powerful connection to the graph exterior calculus from combinatorial Hodge theory. Consider a complete graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with the vertex set $\mathcal{V}$, edge set $\mathcal{E}$, and higher-order k-cliques denoted by the oriented tuples $(i_1, \ldots, i_k)$. The standard k^{th}-order coboundary operator δ_k is simply associated with the oriented incidence matrix between $k+1$- and k-cliques. Specifically, the graph gradient δ_0 and graph curl δ_1 are defined by

$$\begin{aligned} (\delta_0 u)_{ij} &= u_j - u_i \\ (\delta_1 u)_{ijk} &= u_{ij} + u_{jk} + u_{ki}, \end{aligned}$$

where u_i denote a scalar value associated with the node i, $u_{ij} = -u_{ji}$ denotes a scalar associated with the edge $(i, j) \in \mathcal{E}$, and u_{ijk} a value associated with the 3-cliques (e.g. faces) which is antisymmetric with respect to the index ordering

$$u_{ijk} = -u_{ikj} = -u_{jik} = -u_{kji} = u_{kij} = u_{jki}.$$

The adjoint of coboundary operators induces the so-called codifferential operators, which in this setting provide definitions of graph divergence and curl:

$$\begin{aligned}(DIV\, u)_i &:= (\delta_0^T u)_i = \sum_{j \neq i} u_{ij}, \\ (CURL\, u)_{ij} &:= (\delta_1^T u)_i = \sum_{k \neq i,\, j} u_{ijk}.\end{aligned} \tag{3.5}$$

These graph operators have a number of properties mimicking the familiar vector calculus, but follow only from the topological properties of graphs. For example, the *exact sequence* property $DIV \circ CURL = 0$ discretely parallels $\nabla \cdot \nabla \times = 0$, and conservation structure is reflected in DIV calculating the sum of antisymmetric generalized fluxes.

The connection between the parameterized Whitney form space and the combinatorial Hodge theory follows by rewriting (3.4) as

$$\mathrm{DIV} = DIV\, \mathbf{M}_1, \quad \mathrm{CURL} = CURL\, \mathbf{M}_2$$

where $(\mathbf{M}_1)_{(ij),(ab)} = (\psi_{ab}, \psi_{ij})$ and $(\mathbf{M}_2)_{(ijk),(abc)} = (\psi_{abc}, \psi_{ijk})$ are mass matrices associated with the finite element spaces V^1 and V^2, respectively. Therefore, we see that the geometry of the PPOUs implicitly induces a weighting on the graph exterior calculus, with the boundaries of learned partitions inducing a topology associated with conservation structure.

We may finally revisit the original task of identifying a model of the form (1.2). Let the Whitney forms associated with subdomain Ω_i be $V^0(\Omega_i)$ and $V^1(\Omega_i)$ by taking $\omega = \Omega_i$. Mirroring (1.2), the model on each individual subdomain is equivalent to the following variational problem: find $(p_i, \boldsymbol{u}_i) \in V^0(\Omega_i) \times V^1(\Omega_i)$ such that for all $(w_i, \boldsymbol{v}_i) \in V_0^0(\Omega_i) \times V^1(\Omega_i)$,

$$\begin{aligned}(\boldsymbol{u}_i, \boldsymbol{v}_i) - (h(p_i; \theta_i), \boldsymbol{v}_i) &= 0 \\ (\boldsymbol{u}_i, \nabla w_i) &= (f_i, w_i)\end{aligned}$$

with Dirichlet boundary condition $p_i = g_i$ on $\partial\Omega$, which is enforced by using a standard lift.

Following the theory laid out in Trask et al. (2022), we could assume the unknown fluxes take the form of a nonlinear perturbation of a diffusive flux while maintaining a tractable stability analysis, e.g.

$$h(p_i; \theta_i) = \nabla p_i + N[p_i; \theta_i]$$

However in the current work, we will consider only the linear case ($N[p_i; \theta_i] = 0$). In this setting the Whitney forms will identify the geometry and properties associated with material heterogeneities under an assumed diffusion process,

providing the following variational problem on each element.

$$\begin{aligned}(\boldsymbol{u}_i, \boldsymbol{v}_i) - (\nabla p_i, \boldsymbol{v}_i) &= 0 \\ (\boldsymbol{u}_i, \nabla w_i) &= (f_i, w_i).\end{aligned} \tag{3.6}$$

Finally we substitute in the discrete exterior derivatives associated with the PPOUs to obtain a discrete parametric model, posing the following equality constrained optimization problem to calibrate the POU geometry to data,

$$\begin{aligned}&\min_{W, \boldsymbol{B}_0, \boldsymbol{B}_1, \boldsymbol{D}_0 \boldsymbol{D}_1} \left\| p_{\text{data}} - \sum_i \hat{p}_i \psi_i \right\|_2^2 + \alpha^2 \left\| F_{\text{data}} - \sum_{ij} \hat{F}_{ij} \psi_{ij} \right\|_2^2 \\ &\text{such that} \begin{bmatrix} \mathbf{M}_1 & -\mathbf{M}_1 \mathbf{D}_1^{-1} \delta_0 \mathbf{D}_0 \\ -\mathbf{B}_0^{-1} \delta_0^T \mathbf{B}_1 \mathbf{M}_1 & \mathbf{0} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{F}} \\ \widehat{\mathbf{p}} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_D \\ -\mathbf{b}_f \end{bmatrix},\end{aligned} \tag{3.7}$$

where $\mathbf{B_k}$ and $\mathbf{D}_k$ are diagonal matrices with trainable positive coefficients, $\mathbf{b}_D$, $\mathbf{b}_f$ the terms arising from the Dirichlet boundary condition and forcing term respectively, α a normalization parameter, and W the remaining weights associated with the POUs such as the location of knots and the convex combination tensor. As shown in Actor et al. (2024), $\mathbf{B_k}$ and $\mathbf{D}_k$ infer metric information from data without impacting the topological structure of the model. For further details regarding the specific construction of POUs we refer to Actor et al. (2024).

Remark 3.1. The Whitney form construction supports a number of theoretical constructions: a Hodge decomposition, Poincare inequality, a corresponding Lax-Milgram theory, a well-posedness theory for certain nonlinear elliptic problems, and discrete preservation of exact sequence properties which exactly preserve conservation structure. When we use the Whitney form elements to construct the subdomain spaces V_i in the mortar method in the following section, we aim to carefully construct the mortar space so that this structure is not lost at the global level.

4 Mortar method

After the local models are trained, we seek to construct a mortar method which is flexible enough to couple FEEC elements on the different subdomains together. Note that since the fine-scale knots are able to move during pretraining, the mortar is necessarily nonconforming, with possible "hanging" mortar nodes which do not coincide with the neighboring local element nodes; this necessitates an analysis of stability associated with projecting between local and mortar spaces. Furthermore, we would like the mortar method to preserve the conservation and stability properties outlined in the introduction (**R1**, **R2**).

As discussed in Section 3, we assume that our data $\{(\boldsymbol{u}(x_k), p(x_k)), g_k\}_{k=0}^N$ (with $x_k \in \Omega$ sampled randomly) satisfy the following variational equation: seek

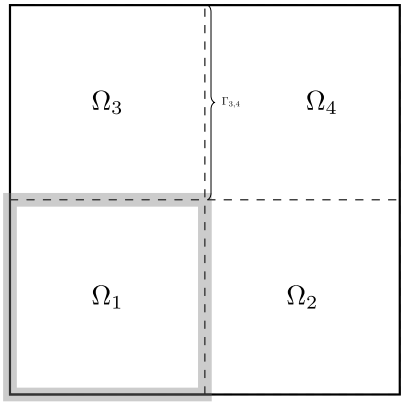


FIGURE 2 Figure of a square domain Ω divided into four subdomains. The edge $\Gamma_{3,4}$ is denoted explicitly and the highlighted boundary is Γ_1.

solution $(\boldsymbol{u}, p) \in (L^2(\Omega)^2, H_g^1(\Omega))$ such that

$$\begin{aligned} (\boldsymbol{u}, \boldsymbol{v}) - (K\nabla p, \boldsymbol{v}) &= 0, \qquad && \forall \boldsymbol{v} \in L^2(\Omega)^2 \\ (\boldsymbol{u}, \nabla w) &= (f, w), && \forall w \in H_0^1(\Omega) \end{aligned} \tag{4.1}$$

where H^1 is the standard Sobolev space and $H_g^1(\Omega) = \{u \in H^1(\Omega) \mid u|_{\partial\Omega} = g_k\}$ (Braess, 2007), and the tensor $K \in L^\infty$ is a positive-definite matrix. Finally, we assume the problem is of at least $p \in H^{3/2}(\Omega)$ regularity, which arises naturally if, for example, $f \in L^2(\Omega)$, $g \in H^{3/2}(\partial\Omega)$ with Lipschitz coefficients K and Ω is convex (Grisvard, 2011). We will see in our numerical results that the above regularity result is a sufficient condition for the error analysis, and not a necessary one.

Let Ω be divided into n nonoverlapping, polygonal subdomain blocks Ω_i of similar aspect ratios. Let Γ_i be the edges of Ω_i, $\Gamma = \cup_i \Gamma_i$ the set of all boundaries of the subdomains (including those intersecting $\partial\Omega$), and let $\Gamma_{ij} = \Gamma_i \cap \Gamma_j$ for all i, j be the boundary between two adjacent subdomains. See Fig. 2 for an illustrative figure.

Define

$$\Lambda := \{v \in L^2(\Gamma) \mid \exists u \in H^1(\Omega), u|_\Gamma = v\} \tag{4.2}$$

as the space of L^2 functions on the interfaces which are the traces of H^1 functions, and the subspaces

$$\begin{aligned} \Lambda_0 &:= \{\lambda \in \Lambda \mid \lambda|_{\partial\Omega} = 0\}, \\ \Lambda_g &:= \{\lambda \in \Lambda \mid \lambda|_{\partial\Omega} = g\}. \end{aligned}$$

Note that since Λ consists of the trace of H^1 functions, we may endow Λ with the $H^{1/2}$ norm on Γ.

4.1 Stability analysis for continuous case

Before proceeding to the model discovery problem and the discrete, we first consider smooth solutions coming from solutions from diffusion problem to guide the design of a suitable mortar method. It is straightforward to decompose (4.1) into problems on the subdomains $\{\Omega_i\}_{i=1}^n$ by introducing a mortar representing the pressure on the space Λ:

Lemma 4.1. *For $1 \leq i \leq n$, let $(\boldsymbol{u}_i, p_i, \lambda) \in (L^2(\Omega_i)^2, H^1(\Omega_i), \Lambda_g)$ such that*

$$\begin{aligned} (\boldsymbol{u}_i, \boldsymbol{v}_i)_{\Omega_i} - (K\nabla p_i, \boldsymbol{v}_i)_{\Omega_i} &= 0, \qquad && \forall \boldsymbol{v}_i \in L^2(\Omega_i)^2 \\ (\boldsymbol{u}_i, \nabla w_i)_{\Omega_i} &= (f, w_i)_{\Omega_i}, && \forall w_i \in H_0^1(\Omega_i) \end{aligned} \tag{4.3}$$

with continuity of state and flux enforced via the boundary condition $p_i|_{\Gamma_i} = \lambda|_{\Gamma_i}$ and weak flux continuity condition

$$\sum_{i=1}^{n} (\boldsymbol{u}_i, \nabla w)_{\Omega_i} = (f, w), \qquad \forall w \in H_0^1(\Omega). \tag{4.4}$$

Then $\boldsymbol{u} = \sum_{i=1}^n \boldsymbol{u}_i \in L^2(\Omega)^2$, $p = \sum_{i=1}^n p_i \in H_g^1(\Omega)$ solves (4.1).

Proof. The existence of functions $(\boldsymbol{u}_i, p_i)$ and λ comes trivially by restricting the solution from (4.1) to the individual subdomains and mortar space.

To see that (4.3) and (4.4) implies (4.1), we note that $L^2(\Omega)^2 = \bigoplus_{i=1}^n L^2(\Omega_i)^2$, and thus by summing the first equation of (4.3) and choosing $\boldsymbol{v}_i = \boldsymbol{v}|_{\Omega_i}$ as test functions, we have

$$\left(\sum_{i=1}^{n} \boldsymbol{u}_i, \boldsymbol{v}\right) - \left(K\nabla \sum_{i=1}^{n} p_i, \boldsymbol{v}\right) = 0, \qquad \forall \boldsymbol{v} \in L^2(\Omega)^2$$

with $\sum_{i=1}^n p_i \in H_g^1(\Omega)$ since continuity is enforced with λ. As for the test functions arising in $w \in H_0^1(\Omega)$, we simply decompose w into $\sum_{i=1}^n w_i + w_0$ where $w_i \in H_0^1(\Omega_i)$ for $1 \leq i \leq n$ and $w_0 := w - \sum_{i=1}^n w_i$, so that the summation of the second equation of (4.3) and (4.4) gives us the desired result. □

The condition (4.4) can be simplified. Consider the space $H^\gamma(\Omega)$ satisfying the decomposition

$$H^1(\Omega) = H_0^1(\Omega_1) \oplus \cdots \oplus H_0^1(\Omega_n) \oplus H^\gamma(\Omega) \tag{4.5}$$

with $H^\gamma \perp H_0^1(\Omega_i)$ relative to the H^1 norm for each i. Then, using to (4.3) and (4.5), (4.4) can be rewritten as

$$\sum_{i=1}^{n}(\boldsymbol{u}_i, \nabla w) = (f, w), \qquad \forall w \in H_0^\gamma(\Omega) \tag{4.6}$$

where $H_0^\gamma := \{u \in H^\gamma(\Omega) \mid u|_{\partial\Omega} = 0\}$. We also define the subset $H_g^\gamma := \{u \in H^\gamma(\Omega) \mid u|_{\partial\Omega} = g\}$. The space H^γ corresponds to a minimal energy extension (Toselli and Widlund, 2004) as the following lemma shows:

Lemma 4.2. *For all $u \in H^1(\Omega)$, there exists a unique decomposition $u = u_\gamma + \sum_{i=1}^n u_i$ such that $u_\gamma \in H^\gamma(\Omega)$, $u_i \in H_0^1(\Omega_i)$. Furthermore, one has*

$$\|u_\gamma\|_{H^1(\Omega)} = \inf_{v \in H^1(\Omega), v|_\Gamma = u} \|v\|_{H^1(\Omega)} \simeq \sum_{i=1}^{n} \|u_\gamma\|_{H^{1/2}(\Gamma_i)}.$$

Proof. Given u, consider $u_I \in H_0^1(\Omega_1) \oplus \cdots \oplus H_0^1(\Omega_n)$ such that for $1 \le i \le n$,

$$(u_I, v_i)_{H^1(\Omega_i)} = (u, v_i)_{H^1(\Omega_i)}, \qquad \forall v_i \in H_0^1(\Omega_i).$$

Then the decomposition is simply $u = \sum_{i=1}^n u_I|_{\Omega_i} + u_\gamma$ where $u_\gamma = u - u_I$. The orthogonality is enforced since, for all w_i in $H_0^1(\Omega_i)$ and $1 \le i \le n$,

$$(u_\gamma, w_i)_{H^1(\Omega)} = (u - u_I, w_i)_{H^1(\Omega_i)} = (u, w_i)_{H^1(\Omega_i)} - (u_I, w_i)_{H^1(\Omega_i)} = 0.$$

As for the minimal condition, let $v = u_\gamma + \sum_{i=1}^n v_i$ with $v_i \in H_0^1(\Omega_i)$ arbitrary, then by orthogonality

$$\|v\|_{H^1(\Omega)}^2 = \|u_\gamma\|_{H^1(\Omega)}^2 + \left\|\sum_{i=1}^{n} v_i\right\|_{H^1(\Omega)}^2 \ge \|u_\gamma\|_{H^1(\Omega)}^2$$

and the $H^{1/2}$ equivalence is well known (Bertoluzza and Kunoth, 2000; Cowsar et al., 1995). □

With the above decomposition, we can further reduce (4.1) to be a variational problem only on H^γ and Λ. Let $\lambda, \mu \in H^\gamma$, define the bilinear form and linear functional

$$b(\lambda, \mu) = \sum_{i=1}^{n}(\boldsymbol{u}^*(\lambda), \nabla\mu)_{\Omega_i} \tag{4.7}$$

and

$$L(\mu) = (f, \mu)_\Omega - \sum_{i=1}^{n}(\bar{\boldsymbol{u}}, \nabla\mu)_{\Omega_i} \tag{4.8}$$

where $(\boldsymbol{u}^*(\lambda), p^*(\lambda)) \in (L^2(\Omega)^2, H^1(\Omega))$ solves the local problems, for $1 \leq i \leq n$,

$$\begin{aligned}(\boldsymbol{u}^*(\lambda), \boldsymbol{v}) - (K\nabla p^*(\lambda), \boldsymbol{v}) &= 0, \quad \forall \boldsymbol{v} \in L^2(\Omega_i)^2 \\ (\boldsymbol{u}^*(\lambda), \nabla w) &= 0, \quad \forall w \in H_0^1(\Omega_i)\end{aligned} \tag{4.9}$$

with boundary condition $p^*(\lambda)|_{\Gamma_i} = \lambda|_{\Gamma_i}$, and where $(\bar{\boldsymbol{u}}, \bar{p}) \in (L(\Omega)^2, H_0^1(\Omega))$ solves, for $1 \leq i \leq n$,

$$\begin{aligned}(\bar{\boldsymbol{u}}, \boldsymbol{v}) - (K\nabla \bar{p}, \boldsymbol{v}) &= 0, \quad \forall \boldsymbol{v} \in L^2(\Omega_i)^2 \\ (\bar{\boldsymbol{u}}, \nabla w) &= (f, w), \quad \forall w \in H_0^1(\Omega_i)\end{aligned} \tag{4.10}$$

with boundary condition $\bar{p}|_{\Gamma_i} = 0$. The bilinear form and linear functional closely resemble those of the $H(\mathrm{div})$ case from Arbogast et al. (2007, 2000). Note that the problems (4.9) and (4.10) above are local in nature and can be solved in parallel.

The following lemma shows that one can recover the original variational equations by working with the above bilinear form:

Lemma 4.3. *Let $\lambda \in H_g^\gamma$ be the solution to the variational equation,*

$$b(\lambda, \mu) = L(\mu), \qquad \forall \mu \in H_0^\gamma \tag{4.11}$$

then $\boldsymbol{u} := \boldsymbol{u}^*(\lambda) + \bar{\boldsymbol{u}}$, $p := p^*(\lambda) + \bar{p}$ is the solution to (4.1).

Proof. Summing (4.9) and (4.10) results in

$$\begin{aligned}(\boldsymbol{u}, \boldsymbol{v}) - (K\nabla p, \boldsymbol{v}) &= 0, \quad \forall \boldsymbol{v} \in L^2(\Omega_i)^2 \\ (\boldsymbol{u}, \nabla w) &= (f, w), \quad \forall w \in H_0^1(\Omega_i)\end{aligned}$$

with $p|_\Gamma = \lambda$ for each $1 \leq i \leq n$.

It remains to check (4.6), but this is simply because if (4.11) holds, then

$$\sum_{i=1}^{n} (\boldsymbol{u}, \nabla \mu)_{\Omega_i} = (f, \mu)$$

for all $\mu \in H_0^\gamma$ and the results follows from Lemma 4.1 and (4.6). □

Finally, we note that the variational equation (4.11) is well-defined as the bilinear form is coercive as shown in the following lemma, whose proof is delayed until the appendix:

Lemma 4.4. *The bilinear form (4.7) is symmetric and coercive on Λ_0.*

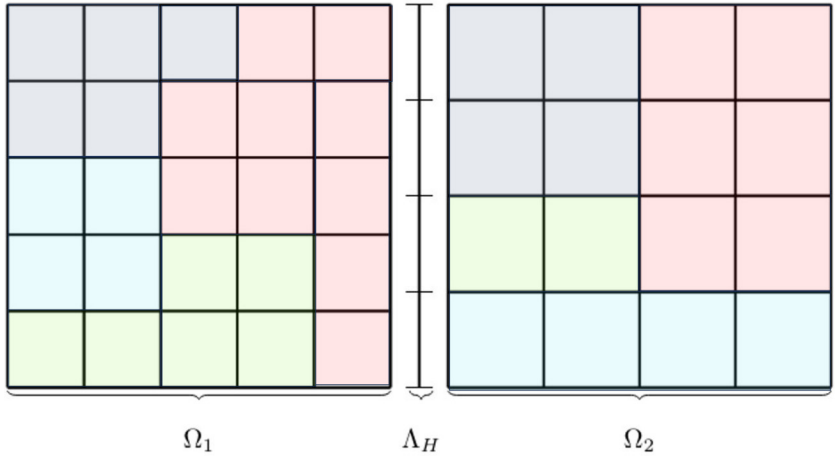


FIGURE 3 Sketch of a 4 element mortar Λ_H and its two adjacent subdomains. The colors on the subdomains represent the PPOUs constructed as convex combinations of a fine-scale B-splines. We note that while the mortar matches the fine-scale nodes on Ω_2, it is disjoint from Ω_1, requiring analysis of a remap/projection between the two meshes. Because the FEEC fine-scale nodes on Ω_i evolve during training, they will generally not coincide with mortar nodes.

4.2 Discretized case

The discrete case is more technical, since both the spaces and the bilinear form are discretized as (4.9) and (4.10) cannot be solved exactly. Further, care must be taken to treat the nonconforming grids that emerge naturally as nodes between adjacent subdomains evolve.

In what follows, the subscripts h, H will denote a discretized version of a continuous space. On each subdomain Ω_i, let W_{hi}, $\boldsymbol{V}_{hi}$ be the discretized versions of $H^1(\Omega_i)$, $L^2(\Omega_i)^2$ respectively. We require the standard inf-sup compatibility between W_{hi}, $\boldsymbol{V}_{hi}$, which in this case is simply the condition $\nabla W_{hi} \subseteq \boldsymbol{V}_{hi}$ (Arnold, 2018; Braess, 2007). In particular, we can choose W_{hi} and $\boldsymbol{V}_{hi}$ to be the spaces V^0 and V^1 defined in (3.2) in the case of FEEC elements; by construction then we have $\nabla V^0 = V^1$. We will also use the case where local elements are taken to be traditional finite elements to show convergence; in this case we will consider V^0 and V^1 as continuous $\mathbb{Q}_1$ and lowest-order Nedelec elements, respectively. Finally, we let $W_{hi,0}$ be the subspace with homogeneous Dirichlet boundary condition (e.g. (3.3)).

On the interfaces, we choose $\Lambda_H \subset \Lambda$ to be the space of continuous, piecewise linear functions. Let $\Lambda_{H,0} := \{\mu_H \in \Lambda_H \mid \mu_H|_{\partial\Omega} = 0\} \subset \Lambda_0$ and similarly let $\Lambda_{H,g} \subset \Lambda_g$ be the subset whereby the boundary is equal to g. We allow the discretization between two subdomains to be different and also allow for the mortars to be nonmatching. See Fig. 3 for a simplified figure where there are nonmatching tensor-product grids.

We define a projection for each subdomain $Q_i : \Lambda_H \to W_{hi}$ such that for all $\lambda_H \in \Lambda_H$

$$(\lambda_H - Q_i\lambda_H, p_h)_{L^2(\Gamma_i)} = 0, \qquad \forall p_h \in W_{hi}|_{\Gamma_i} \tag{4.12}$$

and

$$(Q_i\lambda_H, w)_{H^1(\Omega_i)} = 0, \qquad \forall w \in W_{hi,0}.$$

The first condition simply defines the boundary of $Q_i\lambda_H$ using the L^2-projection[1] on Ω_i while the second condition means that it is the discrete harmonic extension into Ω_i given the boundary $Q_i\lambda_H$ on Γ_i (Toselli and Widlund, 2004). We note that in general, the projections to the left and right of that interface are different since the discretization can be different on either sides as can be seen in Fig. 3.

With the above in hand, we can define the discretized bilinear operator and linear functional similar to (4.7) and (4.8). For $\lambda_H, \mu_H \in \Lambda_H$, let

$$b_h(\lambda_H, \mu_H) := \sum_{i=1}^{n} (\boldsymbol{u}_h^*(Q_i\lambda_H), \nabla(Q_i\mu_H))_{\Omega_i} \tag{4.13}$$

and

$$L_h(\mu_H) := \sum_{i=1}^{n} (f, Q_i\mu_H)_{\Omega_i} - (\bar{\boldsymbol{u}}_h, \nabla(Q_i\mu_H))_{\Omega_i} \tag{4.14}$$

where $p_h^*(Q_i\lambda_H) \in \oplus_{i=1}^n W_{hi}$, $\boldsymbol{u}_h^*(Q_i\lambda_H) \in \oplus_{i=1}^n \boldsymbol{V}_{hi}$ satisfies, for $1 \le i \le n$,

$$(\boldsymbol{u}_h^*(Q_i\lambda_H), \boldsymbol{v}_h) - (K\nabla p_h^*(Q_i\lambda_H), \boldsymbol{v}_h) = 0, \qquad \forall \boldsymbol{v}_h \in \boldsymbol{V}_{hi} \tag{4.15}$$

$$(\boldsymbol{u}_h^*(Q_i\lambda_H), \nabla w_h) = 0, \qquad \forall w_h \in W_{hi,0} \tag{4.16}$$

with $p_h^*(Q_i\lambda_H) = Q_i\lambda_H$ on Γ_i, and $\bar{p}_h \in \oplus_{i=1}^n W_{hi}$, $\bar{\boldsymbol{u}}_h \in \oplus_{i=1}^n V_{hi}$ satisfying

$$(\bar{\boldsymbol{u}}_h, \boldsymbol{v}_h) - (K\nabla\bar{p}_h, \boldsymbol{v}_h) = 0, \qquad \forall \boldsymbol{v}_h \in \boldsymbol{V}_{hi} \tag{4.17}$$

$$(\bar{\boldsymbol{u}}_h, \nabla w_h) = (f, w_h), \qquad \forall w_h \in W_{hi,0} \tag{4.18}$$

with $\bar{p}_h = 0$ on Γ_i. As before, the above problems are defined locally and can be solved in parallel.

We state the discrete variational equation as follows. Find $\lambda_H \in \Lambda_{H,g}$ such that

$$b_h(\lambda_H, \mu_H) = L_h(\mu_H), \qquad \forall \mu_H \in \Lambda_{H,0}. \tag{4.19}$$

[1] We found in our numerical examples that using the interpolant suffices, however we will carry out the analysis using the projection.

The well-posedness of the variational form can be deduced from Lax-Milgram if the coercivity condition

$$b_h(\lambda_H, \lambda_H) \geq \alpha \sum_{i=1}^{n} \|\lambda_H\|^2_{H^{1/2}(\Gamma_i)} \tag{4.20}$$

is true. The coercivity condition (4.20) will require two assumptions which excludes pathological discretizations:

1. Assumption 1 (injectivity): for all $\lambda_H \in \Lambda_H$, there exists a constant C such that

$$\sum_{i=1}^{n} \| Q_i \lambda_H \|_{H^{1/2}(\Gamma_i)} \geq C \sum_{i=1}^{n} \|\lambda_H\|_{H^{1/2}(\Gamma_i)} \tag{4.21}$$

   meaning we have unisolvency when projecting from the mortar space onto the local subdomains.
2. Assumption 2 (strengthened triangle inequality): for each shared edge Γ_{ij} and for all $\lambda_H \in \Lambda_H$, that

$$\frac{C_p}{|\Gamma_{ij}|} \left\| Q_i \lambda_H - Q_j \lambda_H \right\|^2_{L^2(\Gamma_{ij})} \leq \frac{1}{2} (\| Q_i \lambda_H \|^2_{H^{1/2}(\Gamma_{ij})} + \left\| Q_j \lambda_H \right\|^2_{H^{1/2}(\Gamma_{ij})}) \tag{4.22}$$

   where C_p is the Poincare constant arising in Brenner (2003, (1.3)) and $|\Gamma_{ij}|$ is the length of the shared edge. The condition means two adjacent subdomains cannot have too large of a difference in their discretization parameter. In particular, if two adjacent subdomains have the same, symmetric discretization parameters, then the left side of (4.22) is trivially zero.

In the case of data-driven elements, extra care must be paid to Assumption 1 since a training procedure might move the fine-scale nodes such that unisolvency is lost. However, this can be circumvented by either placing restrictions on the movement of the nodes, or, as in some of our numerical examples, using a very coarse mortar space.

With the above assumptions, we can now state the stability result:

Lemma 4.5. *With the above two assumptions, the discretized bilinear form* (4.13) *is coercive (e.g.* (4.20)*) over* $\Lambda_{H,0}$.

The proof of the above lemma is technical and is delayed to the appendix.

Lemma 4.5 means that one is allowed to apply Strang's second lemma to obtain error estimates. We assume that an a priori estimate exists: let δ be a constant such that the discrete approximations on each subdomain $1 \leq i \leq n$ satisfy

$$\left\| p^*(\lambda) - p_h^*(Q_i \lambda) \right\|_{\Omega_i} \leq \delta, \qquad \left\| \boldsymbol{u}^*(\lambda) - \boldsymbol{u}_h^*(Q_i \lambda) \right\|_{\Omega_i} \leq \delta \tag{4.23}$$

$$\left\| \bar{p} - \bar{p}_h \right\|_{\Omega_i} \leq \delta \qquad \|\bar{\boldsymbol{u}} - \bar{\boldsymbol{u}}_h\|_{\Omega_i} \leq \delta \tag{4.24}$$

for all $\lambda \in \Lambda$. The constant δ corresponds to the ability of the local solvers to solve for p^*, $\boldsymbol{u}^*$ accurately for an arbitrary mortar. In the case where standard FEM is used on the subdomain, then δ can be replaced with the respective a priori estimate whereas for the DDEC methods, this corresponds to an optimization threshold.

We can now state a simple convergence guarantee **R2** on the mortar space, whose proof is delayed until the appendix:

Theorem 4.6. *Suppose the solution to the (4.1) is such that $p \in H^2(\Omega)$ with homogeneous Dirichlet boundary condition. Then there exists a constant C independent of λ^* such that*

$$\sum_{i=1}^{n} \left\| \lambda^* - \lambda_H^* \right\|_{H^{1/2}(\Gamma_i)} \leq Cn \left| p \right|_{H^2(\Omega)} (H + h + \delta) \tag{4.25}$$

where λ^* is the true solution to (4.11), and λ_H^* is the solution to (4.19), and H, h are the maximal mesh sizes on Λ_H and the boundary of the subdomains Ω_i, respectively.

Remark 4.7. As mentioned, the constant δ associated with (4.23) corresponds to the accuracy of the local solvers while the h term relates to the accuracy of projecting the mortar to the local subdomains using (4.12), though in general we can assume that $h < H$. We also note that (4.25) implies that a combination of refinement of both the local solvers and the mortar space is needed to obtain convergence.

Finally, we can easily bound the error on the pressure and velocity explicitly.

Lemma 4.8. *With the same assumptions and constants as in Theorem 4.6, there exists a constant C independent of $\boldsymbol{u}$ and p such that*

$$\left\| p - p_h \right\|_{\Omega} + \|\boldsymbol{u} - \boldsymbol{u}_h\|_{\Omega} \leq Cn \left| p \right|_{H^2(\Omega)} (H + h + \delta)$$

where $\boldsymbol{u}$, p are the true solutions arising from (4.3) and $\boldsymbol{u}_h = \sum_{i=1}^{n} \boldsymbol{u}_h^*(Q_i \lambda_H^*) + \bar{\boldsymbol{u}}_h$, $p_h = \sum_{i=1}^{n} p_h^*(Q_i \lambda_H^*) + \bar{p}_h$.

Proof. By Lemma 4.3, we have

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_{\Omega} \leq \sum_{i=1}^{n} \left\| \boldsymbol{u}^*(\lambda^*) - \boldsymbol{u}_h^*(Q_i \lambda_H^*) \right\|_{\Omega_i} + \|\bar{\boldsymbol{u}} - \bar{\boldsymbol{u}}_h\|_{\Omega_i} .$$

The latter term on the right hand side is bounded by δ by assumption. Thus the result follows by

$$\sum_{i=1}^{n} \left\| \boldsymbol{u}^*(\lambda^*) - \boldsymbol{u}_h^*(Q_i \lambda_H^*) \right\|_{\Omega_i} \leq \sum_{i=1}^{n} \left\| \boldsymbol{u}^*(\lambda^*) - \boldsymbol{u}^*(\lambda_H^*) \right\|_{\Omega_i}$$

$$
\begin{aligned}
&+ \left\| \boldsymbol{u}^*(\lambda_H^*) - \boldsymbol{u}_h^*(Q_i \lambda_H^*) \right\|_{\Omega_i} \\
&\leq n\delta + \sum_{i=1}^{n} \left\| \boldsymbol{u}^*(\lambda^* - \lambda_H^*) \right\|_{\Omega_i} \\
&\leq n\delta + \sum_{i=1}^{n} \left\| \lambda^* - \lambda_H^* \right\|_{H^{1/2}(\Gamma_i)}
\end{aligned}
$$

where we used standard regularity estimates at the last step. The same estimates also follow for the pressure and the result follows from applying Theorem 4.6. □

The above error analysis partially shows that requirement **R2** from the introduction is met, as the total error is indeed controlled by a combination of the local optimization error, and coupling error from the mortars. However, due to the use of crude bounds on the sum, it is not independent with the number of subdomains, though we will later observe it holds numerically (cf. Subsection 5.4).

4.3 Data-driven elements with mortar method

Classical finite elements such as Nedelec elements can be used for the local solvers in (4.15) and (4.17) on the subdomains Ω_i in a straightforward manner (see Subsection 5.1 for an example). However, the true strength of the above mortar method is its ability to interface with the data-driven structure-preserving models discussed in Section 3. We briefly discuss combining the usage of the Whitney form elements with the mortar method.

As before, we assume the data is of the form $\{(\boldsymbol{u}(x_k), p(x_k)), g_k\}_{k=0}^{N}$ with x_k sampled randomly on Ω. This can either be supplied via physical data or high-fidelity PDE solvers. Let M be the total number of unique boundary conditions g_k (e.g. $M = 1$ if all data points originate from the same boundary value problem). We assume Ω is divided into subdomains Ω_i. As with most data-driven applications, a large number of data points N is needed, however, only one boundary condition M is needed (see Subsection 5.4.3 for an example with $M = 1$), though more is always better.

The iterative solving process for the mortar (4.19) involves different Dirichlet boundary conditions λ_H being passed into (4.15), meaning that the ability for the data-driven Whitney form solvers to be able to correctly respond to different Dirichlet data is important. Ideally M is large so that a good sampling of Dirichlet conditions around each Ω_i is achieved.

In cases where simulations on each Ω_i are possible, one should perform simulations to obtain responses to a possible mortar boundary conditions. In particular, in our numerical examples, we choose to use either nodal functions $\{(1-x)(1-y), x(1-y), (1-x)y, xy\}$ or edge Bernstein polynomials. The Bernstein polynomials are chosen as they provide a complete basis on $\partial\Omega_i$

and their gradients are very smooth, however other boundary conditions can be chosen. We note that these data are usually cheaper to generate since the subdomains are smaller than Ω, and they can be performed in parallel.

However, the ability to perform these simulations on each subdomain is not always possible. In this case, a simple approach consisting of taking $g_k := p|_{\Omega_i}$ and the corresponding data points $(\boldsymbol{u}(x_k), p(x_k))$ restricted to each Ω_i can be done. While easier, this does lead to higher errors due to undersampling from certain mortar modes. Nevertheless, the structure-preserving nature of the data-driven elements ensures adherence to the underlying invariance.

With the data on each Ω_i chosen, we then solve the minimization problem (3.7) giving us fine-scale nodes, and a coarsening to POUs. *These data-driven elements are then used as the local solvers for* (4.15) *and* (4.17). Some care must be exercised to ensure that Assumption 1 is satisfied; the projection from the mortar space onto the local solvers must be unique. One can mix and match the local solvers, and only use the data-driven elements where the fluxes are unknown and use traditional finite elements elsewhere; see Subsection 5.3 for an example. Specific details regarding the training process for the numerical examples are given in Appendix 10.A.

4.4 Neumann boundary conditions and conservation

We briefly discuss modifications needed to solve the pure Neumann problem $\boldsymbol{u} \cdot \vec{n} = g$ on $\partial\Omega$, and show that the critical conservation and compatibility property of

$$\int_\Omega f + \int_{\partial\Omega} g = 0 \tag{4.26}$$

is satisfied by the discrete mortar method. Such conservation is exhibited in the FEEC elements also Actor et al. (2024), and thus by showing the mortar method exhibits this behavior as well, requirement **R1** is satisfied.

The assumed global model is now to find $(\boldsymbol{u}, p) \in (L^2(\Omega)^2, H^1(\Omega))$ satisfying

$$\begin{aligned} (\boldsymbol{u}, \boldsymbol{v}) - (K\nabla p, \boldsymbol{v}) &= 0, && \forall \boldsymbol{v} \in L^2(\Omega)^2 \\ (\boldsymbol{u}, \nabla w) &= (f, w) + (g, w)_{\partial\Omega}, && \forall w \in H^1(\Omega) \end{aligned} \tag{4.27}$$

with the condition that $(p, 1) = 0$ for uniqueness.

Due to the differences in boundary conditions, a slightly different choice of spaces and decomposition akin to (4.5) is needed. Define

$$H^\gamma_B(\Omega) := \{u \in H^\gamma \mid u|_{\Gamma_i \setminus \partial\Omega} = 0, \forall 1 \le i \le n\} \tag{4.28}$$

and let H^γ_D be such that

$$H^\gamma = H^\gamma_D \oplus H^\gamma_B. \tag{4.29}$$

The space H_B^γ is simply the subspace which vanishes on the interior mortar spaces, while H_D^γ is its complement. Finally, for each $1 \leq i \leq n$, let

$$H_D^1(\Omega_i) := \{u \in H^1(\Omega_i) \mid u|_{\Gamma_i \setminus \partial\Omega} = 0\}, \tag{4.30}$$

the set of H^1 functions vanishing only on the interior boundary. Note that all functions in H_B^γ can be written as a sum of functions in $H_D^1(\Omega_i)$. Hence, a new decomposition can be written

$$H^1(\Omega) = H_D^1(\Omega_1) \oplus \cdots \oplus H_D^1(\Omega_n) \oplus H_D^\gamma(\Omega). \tag{4.31}$$

With the spaces above, we can introduce a mortar that is equivalent to (4.27), up to a constant: for $1 \leq i \leq n$, let $(\boldsymbol{u}_i, p_i, \lambda) \in (L^2(\Omega_i)^2, H^1(\Omega_i), H_D^\gamma(\Omega))$ satisfy

$$\begin{aligned} (\boldsymbol{u}_i, \boldsymbol{v}_i) - (K\nabla p_i, \boldsymbol{v}_i) &= 0, && \forall \boldsymbol{v}_i \in L^2(\Omega_i)^2 \\ (\boldsymbol{u}_i, \nabla w_i) &= (f, w_i) + (g, w_i)_{\Gamma_i \cap \partial\Omega}, && \forall w_i \in H_D^1(\Omega_i) \end{aligned} \tag{4.32}$$

with the boundary condition that $p|_{\Gamma_i \setminus \partial\Omega} = \lambda|_{\Gamma_i \setminus \partial\Omega}$, and

$$\sum_{i=1}^n (\boldsymbol{u}_i, \nabla w)_{\Omega_i} = (f, w) + (g, w_i)_{\partial\Omega}, \qquad \forall w \in H_D^\gamma(\Omega). \tag{4.33}$$

Finally, we can impose $\int_\Gamma \lambda = 0$ for uniqueness. The proof is similar to that of Lemma 4.1 and is omitted.

With the above, it's easy to define the variational problem as before. Small changes are needed in the bilinear form (4.7) and linear functional (4.8): the definition of $(\boldsymbol{u}^*(\lambda), p^*(\lambda))$, $(\bar{\boldsymbol{u}}, \bar{p})$ should be changed to

$$\begin{aligned} (\boldsymbol{u}^*(\lambda), \boldsymbol{v}) - (K\nabla p^*(\lambda), \boldsymbol{v}) &= 0, \quad && \forall \boldsymbol{v} \in L^2(\Omega_i)^2 \\ (\boldsymbol{u}^*(\lambda), \nabla w) &= 0, \quad && \forall w \in H_D^1(\Omega_i) \end{aligned} \tag{4.34}$$

with boundary conditions $p^*(\lambda)|_{\Gamma_i \setminus \partial\Omega} = \lambda|_{\Gamma_i \setminus \partial\Omega}$, and

$$\begin{aligned} (\bar{\boldsymbol{u}}, \boldsymbol{v}) - (K\nabla \bar{p}, \boldsymbol{v}) &= 0, && \forall \boldsymbol{v} \in L^2(\Omega_i)^2 \\ (\bar{\boldsymbol{u}}, \nabla w) &= (f, w) + (g, w)_{\Gamma_i \cap \partial\Omega}, && \forall w \in H_D^1(\Omega_i) \end{aligned} \tag{4.35}$$

with boundary condition $\bar{p}|_{\Gamma_i \setminus \partial\Omega} = 0$. We note that (4.34) and (4.35) are both well-defined for all subdomains due to the Dirichlet boundary conditions on the mortar space, except for the degenerate case where there is only one subdomain. Finally, the variational form is similar, where we seek $\lambda \in H_D^\gamma$

$$\sum_{i=1}^n (\boldsymbol{u}^*(\lambda), \nabla\mu)_{\Omega_i} = (f, \mu)_\Omega + (g, \mu)_{\partial\Omega} - \sum_{i=1}^n (\bar{\boldsymbol{u}}, \nabla\mu)_{\Omega_i}, \qquad \forall \mu \in H_D^\gamma.$$

Turning to the discrete case, let W_{hi}, $\boldsymbol{V}_{hi}$ be as before and let $W_{hi,D}$ be the discretization of $H^1_D(\Omega_i)$. Let $\Lambda_{H,D} \subset \Lambda_H$ be the discretized mortar space consisting of continuous, piecewise linear functions that vanish where H^γ_D is zero. The projection Q_i should be changed to $Q_i : H^\gamma_D \to W_{hi}$ with the same alteration to (4.12).

Thus, the discretized bilinear form and linear functional is similar to before, with the exception that $p^*_h(Q_i\lambda_H) \in \oplus_{i=1}^n W_{hi}$, $\boldsymbol{u}^*_h(Q_i\lambda_H) \in \oplus_{i=1}^n \boldsymbol{V}_{hi}$ satisfies, for $1 \le i \le n$,

$$(\boldsymbol{u}^*_h(Q_i\lambda_H), \boldsymbol{v}_h) - (K\nabla p^*_h(Q_i\lambda_H), \boldsymbol{v}_h) = 0, \qquad \forall \boldsymbol{v}_h \in \boldsymbol{V}_{hi} \tag{4.36}$$

$$(\boldsymbol{u}^*_h(Q_i\lambda_H), \nabla w_h) = 0, \qquad \forall w_h \in W_{hi,D} \tag{4.37}$$

with $p^*_h(Q_i\lambda_H) = Q_i\lambda_H$ on $\Gamma_i \setminus \partial\Omega$, and $\bar{p}_h \in \oplus_{i=1}^n W_{hi}$, $\bar{\boldsymbol{u}}_h \in \oplus_{i=1}^n V_{hi}$ satisfying

$$(\bar{\boldsymbol{u}}_h, \boldsymbol{v}_h) - (K\nabla \bar{p}_h, \boldsymbol{v}_h) = 0, \qquad \forall \boldsymbol{v}_h \in \boldsymbol{V}_{hi} \tag{4.38}$$

$$(\bar{\boldsymbol{u}}_h, \nabla w_h) = (f, w_h) + (g, w_h)_{\partial\Omega}, \qquad \forall w_h \in W_{hi,D} \tag{4.39}$$

with $\bar{p}_h = 0$ on $\Gamma_i \setminus \partial\Omega_i$. The variational form (written explicitly) is to find $\lambda_H \in \Lambda_{H,D}$, with mean zero, such that

$$\sum_{i=1}^n (\boldsymbol{u}^*_h(Q_i\lambda_H), \nabla(Q_i\mu_H))_{\Omega_i} = \sum_{i=1}^n (f, Q_i\mu_H)_{\Omega_i} + (g, Q_i\mu_H)_{\Omega_i\cap\partial\Omega} - (\bar{\boldsymbol{u}}_h, \nabla(Q_i\mu_H))_{\Omega_i}, \qquad \forall \mu_H \in \Lambda_{H,D}. \tag{4.40}$$

Turning to (4.26) and **R1**, it's easy to see that for a function $\mu_H \in \Lambda_{H,D}$, the projection is exact $Q_i\mu_H|_{\Gamma_i\setminus\partial\Omega} = \mu_H|_{\Gamma_i\setminus\partial\Omega}$ if μ_H is constant on the interior edges (e.g. $\mu_H = C$ on $\Gamma_i \setminus \partial\Omega$). Without loss of generality, let $\Lambda_{H,D} \ni \mu_H = 1$ on all $\Gamma_i \setminus \partial\Omega$, then we can choose w_h such that $w_h + Q_i\mu_H = 1$ on $W_{hi,D}$ for each $1 \le i \le n$. Thus, adding (4.37) and (4.39) for $1 \le i \le n$ to (4.40) and rearranging, we obtain

$$\begin{aligned}
&\sum_{i=1}^n (\boldsymbol{u}^*_h(Q_i\lambda_H) + \bar{\boldsymbol{u}}_h, \nabla(w_h + Q_i\mu_H))_{\Omega_i} \\
&\quad = \sum_{i=1}^n (\boldsymbol{u}^*_h(Q_i\lambda_H) + \bar{\boldsymbol{u}}_h, \nabla 1)_{\Omega_i} \\
&\quad = \sum_{i=1}^n (f, w_h + Q_i\mu_H)_{\Omega_i} + (g, w_h + Q_i\mu_H)_{\partial\Omega} \\
&\quad = \sum_{i=1}^n (f, 1)_{\Omega_i} + (g, 1)_{\partial\Omega} = 0,
\end{aligned}$$

meaning (4.26) is valid even in the discrete case with nonmatching mortars.

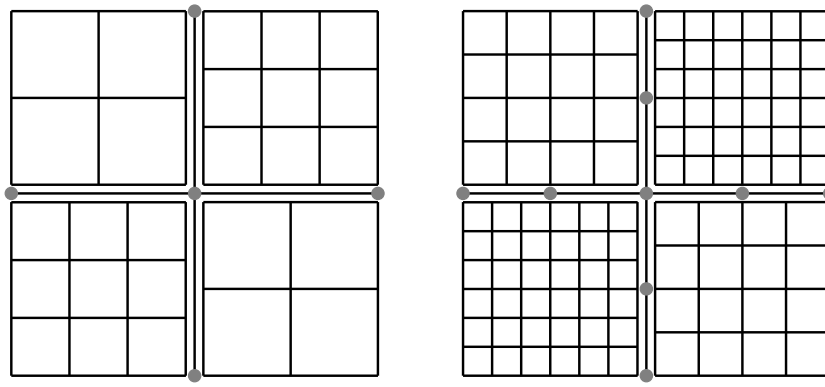


FIGURE 4 Figure illustrating the initial mesh, the corresponding mortar space, and their first refinement for the example in Subsection 5.1.

5 Numerical results

In this section, we present numerical results obtained from applying the above mortar method to several representative examples. All except the first example will involve using pretrained FEEC elements as local subdomain solvers as discussed in Section 3.

5.1 Example 1: pure finite elements

We start by validating the accuracy and well-posedness of the mortar method in the classical setting by using only finite element solvers on each subdomain. In particular, no model training is used for this particular example and we only seek to show that the above mortar method converges in the forward problem. We examine the problem (4.1) with true solution $p(x, y) = xy + y^2$ on the domain $\Omega = [0, 2]^2$ with

$$K = \begin{pmatrix} (x+1)^2 & 0.5 \\ 0.5 & y^2+1 \end{pmatrix}. \tag{5.1}$$

The domain is subdivided into four equal squares. Our initial mesh is depicted in Fig. 4 with only one degree of freedom on the mortar (with the remaining four fixed due to the homogeneous Dirichlet boundary condition). For refinement, we divide each subdomain diameter and the mortar diameter by half; see the right hand side of Fig. 4 for a figure of the first refinement.

On each subdomain, we will use the standard $\mathbb{Q}_1$ space for pressure with Nedelec elements of the lowest order for the velocity. The quantities in Lemma 4.8 can be replaced with results from standard FEM a priori estimates

(Roberts and Thomas, 1991). As a result, we obtain a convergence result of

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_\Omega + \|p - p_h\|_\Omega \leq CH \tag{5.2}$$

where H is the size of the mortar. The $\mathcal{O}(H)$ convergence in the velocity is clearly illustrated in Table 1 while we obtain $\mathcal{O}(H^2)$ superconvergence in the pressure, which was observed in smooth solutions using mortar methods (Arbogast et al., 2007, 2000). Furthermore, since the estimate in Lemma 4.8 is in the $H^{1/2}$ norm, we expect convergence of $\mathcal{O}(H^{3/2})$ as we are measuring the L^2 norm but we also observe a level of superconvergence.

TABLE 1 Table illustrating the absolute errors, and the convergence rates for Example 1. The rates are in agreement with Lemma 4.8.

H	$\|p - p_h\|_\Omega$	$\|\boldsymbol{u} - \boldsymbol{u}_h\|_\Omega$	$\|\lambda - \lambda_h\|_{L^2(\Gamma)}$
1	2.73E-01	4.66E+00	2.44E-01
1/2	6.23E-02	2.16E+00	5.75E-02
1/4	1.49E-02	1.04E+00	1.43E-02
1/8	3.66E-03	5.12E-01	3.56E-03
1/16	9.07E-04	2.54E-01	8.91E-04
1/32	2.31E-04	1.26E-01	2.41E-04
Rate	$\mathcal{O}(H^{2.04})$	$\mathcal{O}(H^{1.04})$	$\mathcal{O}(H^{2.00})$

5.2 Example 2: pure FEEC and pure FEM elements comparison

We now consider the data arising from the problem (4.1) with $\Omega = [0, 3] \times [0, 3]$,

$$f := 2\pi^2 \cos(\pi x) \sin(\pi y), \qquad K = \boldsymbol{I} \tag{5.3}$$

with boundary condition determined by the true solution $p(x, y) = \cos(\pi x) \sin(\pi y)$.

The domain Ω is split into 9 uniform squares whereby either a $\mathbb{Q}_1$ FEM or a pretrained FEEC element is used in each subdomain. The FEEC element is trained on 20,480 uniformly drawn points from $[0, 1]^2$ with 16 POUs on the interior and 16 on the boundary with varying number of fine-scale knots. As discussed in Actor et al. (2024), increasing the number of fine-scale grids is akin to h-refinement in the FEM sense.

To train the FEEC elements, we use data arising from different boundary conditions and forcing terms which corresponds to approximating (4.17) and (4.15):

1. a problem with the same forcing term as in (5.3) but homogeneous zero Dirichlet boundary condition. This corresponds to (4.17).

2. Sixteen different boundary conditions consisting of the Bernstein polynomials of fourth order on the boundary (e.g. x^4y^4, $\binom{4}{1}x^4y(1-y)^3$ etc) and forcing term of $f = 0$. This is needed so that (4.15) can be approximated accurately on the FEEC elements when different boundary conditions are passed in from the mortar.

The solutions to the above boundary value problems were calculated by a low-order finite element solver. For more details regarding the training, we refer the reader to Subsection 10.A.2.

The mortar refinement level was chosen to be $H = 4h$ in for the FEM case. For the FEEC local solvers, we note that the fine-scale nodes can move, resulting in nonuniform meshes; nevertheless, we still choose the same H as the FEM case for comparison's sake.

In Table 2 and Table 3, we show the error resulting from using purely FEEC elements or purely FEM elements on all the subdomain respectively. The convergence rates among the two different solvers are similar, and reflect superconvergence due to the smoothness of the problem. In Fig. 5, we plot the true solution and its fluxes, and the approximate solution and its fluxes on the whole $[0, 3]^2$ domain solved using FEEC elements, while Fig. 6 plots the quantities on the diagonal line from $(0, 0)$ through $(3, 3)$. In both cases, the true solution is well-approximated.

TABLE 2 Table of absolute error for the sine-cosine problem Subsection 5.2 using trained FEEC elements as the subdomain solver. The convergence rates are similar to the method using pure FEM elements. We note that the fine-scale grid roughly corresponds to h-scaling in a standard FEM method (Actor et al., 2024).

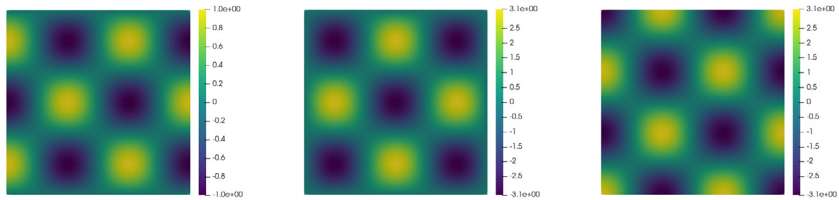
FEEC fine-scale grid and mortar size	$\|p - p_h\|_\Omega$	$\|u - u_h\|_\Omega$	$\|\lambda - \lambda_H\|_{L^2(\Gamma)}$
8×8, $H = 1/2$	1.47E-01	1.48E+00	2.20E-01
12×12, $H = 1/3$	6.66E-02	8.97E-01	8.47E-02
16×16, $H = 1/4$	4.07E-02	5.41E-01	4.41E-02
20×20, $H = 1/5$	2.76E-02	4.40E-01	2.71E-02
24×24, $H = 1/6$	2.10E-02	3.97E-01	1.97E-02
	$\mathcal{O}(h^{1.78})$	$\mathcal{O}(h^{1.25})$	$\mathcal{O}(h^{2.22})$

5.3 Example 3: hybrid methods

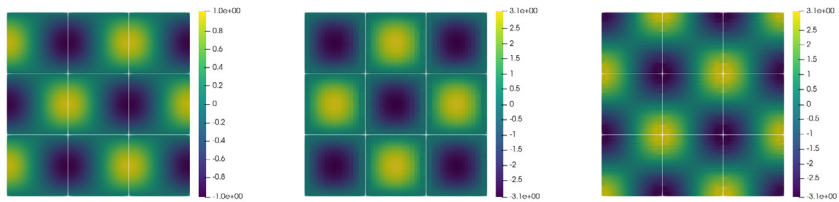
We next showcase the ability to use a hybrid approach whereby standard finite elements are interfaced to FEEC elements allowing for areas with unknown features to be learned using FEEC elements, and smooth areas using classical FEM methods.

TABLE 3 Table of absolute error for the sine-cosine problem Subsection 5.2 with FEM elements as the local solvers.

FEM fine-scale grid and mortar size	$\|p - p_h\|_\Omega$	$\|u - u_h\|_\Omega$	$\|\lambda - \lambda_H\|_{L^2(\Gamma)}$
8×8, $H = 1/2$	1.67E-01	1.54E+00	2.16E-01
12×12, $H = 1/3$	7.04E-02	8.54E-01	8.16E-02
16×16, $H = 1/4$	3.91E-02	5.76E-01	4.31E-02
20×20, $H = 1/5$	2.49E-02	4.30E-01	2.68E-02
24×24, $H = 1/6$	1.72E-02	3.41E-01	1.83E-02
28×28, $H = 1/7$	1.26E-02	2.82E-01	1.33E-02
32×32, $H = 1/8$	9.64E-03	2.40E-01	1.01E-02
36×36, $H = 1/9$	7.61E-03	2.08E-01	7.98E-03
40×40, $H = 1/10$	6.15E-03	1.84E-01	6.45E-03
	$\mathcal{O}(h^{2.04})$	$\mathcal{O}(h^{1.31})$	$\mathcal{O}(h^{2.16})$



(a) True pressure (b) True gradient in x (c) True gradient in y



(d) Estimated pressure (e) Estimated gradient in x (f) Estimated gradient in y

FIGURE 5 Plot of the true (first row) and estimated (second row) solution for the sine-cosine problem Subsection 5.2 with pure FEEC elements consisting of 24×24 fine scale nodes, and $H = 1/6$. As expected, the solution is well-approximated by the FEEC elements.

We assume data is obtained from the problem (4.1) on $\Omega = [-1.5, 1.5]^2$ with the parameters

$$f := 0, \qquad K(\boldsymbol{x}) = \begin{cases} \begin{pmatrix} k & 0 \\ 0 & k \end{pmatrix}, & \|\boldsymbol{x}\| \leq b \\ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, & \|\boldsymbol{x}\| > b \end{cases} \tag{5.4}$$

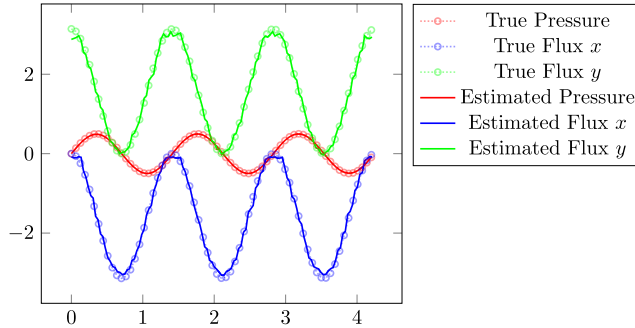


FIGURE 6 Profile of the true and estimate solutions for the sine-cosine problem Subsection 5.2 on the line $(0, 0)$ to $(3, 3)$ using pure FEEC elements with 24×24 fine scale nodes and $H = 1/6$. While there are small fluctuations in the FEEC approximation, it is clear that both the pressure and fluxes are captured.

with $b = .2$, $k = 10$. The Dirichlet boundary imposed such that the true solution is

$$u := \begin{cases} x\left(1 - \frac{b^2(k-1)}{(k+1)(x^2+y^2)}\right) & \|\boldsymbol{x}\| > b \\ \frac{2}{k+1}x & \|\boldsymbol{x}\| \leq b \end{cases}$$

This particular equation arises in electrostatics when examining the case where a conducting cylinder with radius b and capacitance k is placed within a uniform field of strength 1 (Smythe, 1989, §4.03). Note that outside of a radius around the origin, the diffusion problem is easy to solve.

We split the domain into 9 congruent squares with the center square $[-0.5, 0.5]^2$ consisting of a FEEC element to capture the change in material coefficients while the remaining eight subdomains utilizing a simple, low-order FEM space with 8×8 quads. The FEEC element is trained on 12 different boundary conditions corresponding to the 12 third-order Bernstein polynomials on the boundary as in the previous example. We note that in training, only the solution and its fluxes are provided, meaning the material coefficient (5.4) is not fully exposed to the FEEC element. A total of 16 POUs are used on the interior and the boundary. We choose to use a mortar of $H = 1/4$.

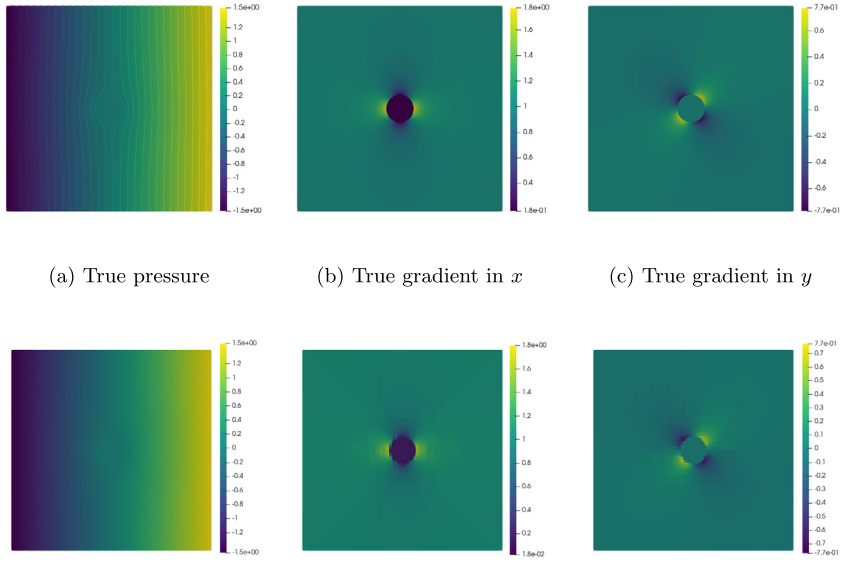
We show the error over the whole domain in Table 4 from only refining the fine-scale grid of the FEEC element in $[-.5, .5]^2$. A full rate of convergence is not expected since Theorem 4.6 requires both the mortar space and the local subdomain solvers to be refined in tandem. We do not consider refinement with the mortar here as Assumption 1 might be violated from either the movement of fine-scale knots of the FEEC elements, or the fact that the mesh size of the FEM solvers are fixed to be very coarse.

In Fig. 7, we plot the true and estimated solution to the problem. Note that the trained FEEC element managed to resolve the circular inclusion and the sub-

TABLE 4 Table of absolute and relative errors for the cylinder problem Subsection 5.3 using a hybrid approach. While we do not expect a full convergence as we are only refining the singular FEEC element on $[-.5, .5]^2$ while keeping the mortar spaces and FEM spaces constant, we do observe that using the finest FEEC element gives significantly better results.

FEEC fine-scale grid	$\|p - p_h\|_\Omega$	$\|u - u_h\|_\Omega$	$\|\lambda - \lambda_H\|_{L^2(\Gamma)}$
8×8	5.69E-03 (0.224%)	1.64E-01 (5.48%)	6.41E-3
16×16	3.07E-03 (0.121%)	1.16E-01 (3.87%)	4.66E-3
24×24	2.01E-03 (0.079%)	8.29E-02 (2.77%)	2.50E-3
	$\mathcal{O}(h^{.939})$	$\mathcal{O}(h^{.607})$	$\mathcal{O}(h^{.814})$

tleties in the fluxes when the true solution is not explicitly given in the training data. Furthermore, we plot the true and estimated solution profiles in Fig. 8. From the plots, it is clear that while there are small spurious fluctuations in the estimated solutions, that the error decreases as we refine the FEEC model. In Fig. 9, we compare the FEEC profiles to the profile obtained using a 24×24 FEM on $[-.5, .5]^2$ instead. Note that the oscillations are greatly reduced by using the FEEC elements due to the adaptivity of the fine-scale mesh.



(a) True pressure (b) True gradient in x (c) True gradient in y

(d) Estimated pressure (e) Estimated gradient in x (f) Estimated gradient in y

FIGURE 7 Figure of the true solution and estimated value for the cylinder problem Subsection 5.3. The estimated solution uses a single FEEC element with 24×24 fine-scale knots in the center-most subdomain with the remaining subdomains using FEM of just 8×8 elements.

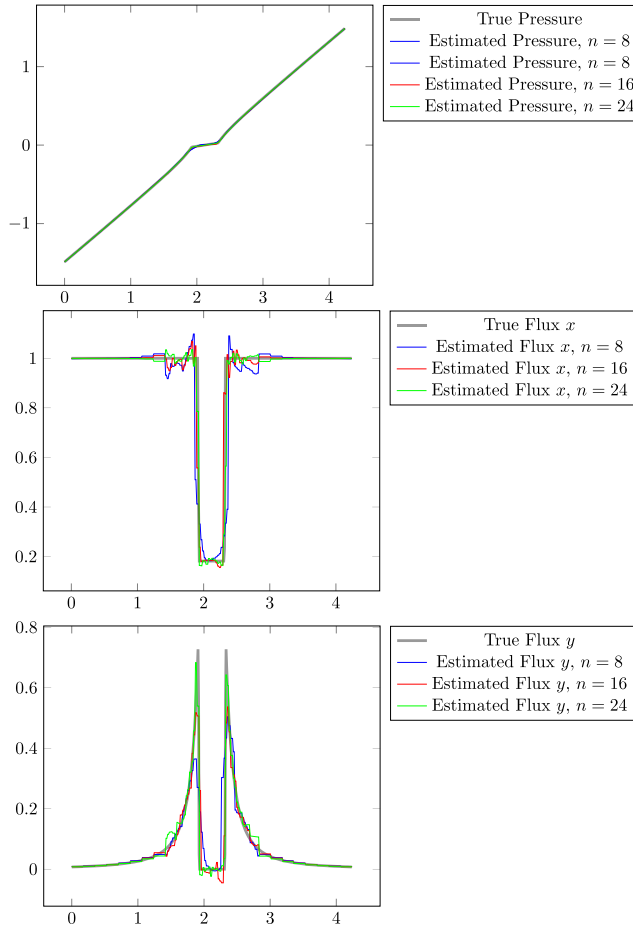
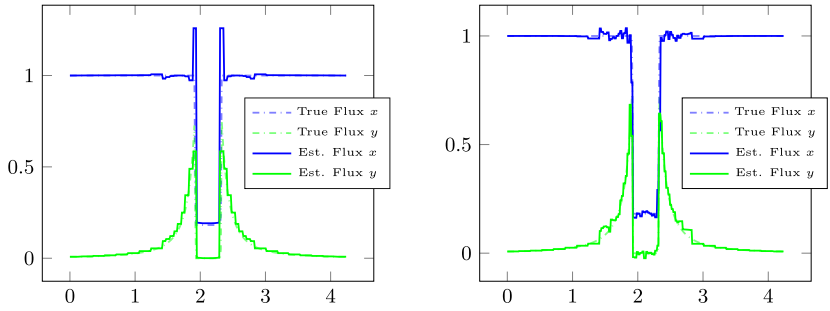


FIGURE 8 Trace plot from $[-1.5, -1.5]$ to $[1.5, 1.5]$ of the cylinder problem Subsection 5.3 for FEEC models with different fine-scale nodes n in the subdomain $[-.5, .5]^2$. As we refine the number of fine-scale nodes we use, the jumps in the fluxes are increasingly more well-resolved with less fluctuations.

5.4 Example 4: subdomain refinement with FEEC

In this next class of examples, we will consider three separate problems whereby the number of subdomains is increased with no further refinement in either the subdomain-level solver, or the number of mortar degrees of freedom per subdomain. This is a nonstandard example case in the context of domain decomposition methods, but is extremely useful in the case where machine-learned elements are used.

We hypothesize that smaller subdomains means that there are fewer features for each FEEC element to learn, meaning that the optimization procedure will



(a) Line from $(-1.5, -1.5) \to (1.5, 1.5)$ with 24×24 FEM in center.

(b) Line from $(-1.5, -1.5) \to (1.5, 1.5)$ with 24×24 fine scale FEEC in center

FIGURE 9 Comparison between using FEM (left) and FEEC (right) solvers in the material discontinuity region $[-.5, .5]^2$ for the cylinder problem Subsection 5.3. Note the overshoot in the discontinuity in the x component of the flux for the pure finite elements case, resulting in a relative error of over 25% near the discontinuity. On the other hand, the FEEC element is able to reduce that fluctuation near the discontinuity to less than 5% using the same number of fine-scale knots due to the adaptivity.

usually result in smaller local losses. The smaller number of features to capture also means that we can use FEEC elements without as many fine-scale nodes, decreasing computational costs in training. Furthermore, in the case with large amount of data points, smaller subdomains means that one can speed up the training tremendously as all the training points can now fit on a single GPU.

In the first two examples, we perform a similar training procedure as before where on each subdomain, a suite of boundary conditions are used to train the local Whitney elements. The last example is more representative of a possible usage case where only a single reference solution is provided with realistic multiscale features.

5.4.1 *Stripe problem*

Consider data arising from the problem (4.1) with $\Omega = [0, n] \times [0, n]$ for n a positive integer,

$$f := 0, \quad g := x, \qquad K = \kappa_i \mathbf{I} \tag{5.5}$$

where $\mathbf{I}$ is the $\mathbb{R}^{2\times 2}$ identity matrix, and where if $\lfloor y \rfloor$ is even, then

$$\kappa_i = \begin{cases} 1 & 0 \le y < .4 \\ .4 & .4 \le y < .8 \\ .8 & .8 \le y < 1 \end{cases}$$

otherwise,

$$\kappa_i = \begin{cases} .8 & 0 \leq y < .2 \\ .3 & .4 \leq y < .6 \\ .9 & .8 \leq y < 1 \end{cases}$$

While the true solution for the pressure is trivially $p(x) = x$ for all n, the difficulty lies in the ability of the discrete solution to capture the discontinuous velocities

$$\boldsymbol{u}(x) := \begin{pmatrix} \kappa_i \\ 0 \end{pmatrix}$$

which arises.

Two FEEC elements of size $[0, 1]^2$ are trained: one to capture the case where $\lfloor y \rfloor$ is even, and another for the odd case. For both FEEC elements, a total of 20×20 fine scale nodes were used, which was subsequently compressed down to 14 POUs on the interior and 14 on the boundary. To train the two FEEC systems, we minimize the MSE against only four PDEs corresponding to the Laplace equation $f = 0$ with the boundary conditions xy, $x(1-y)$, $(1-x)y$, $(1-x)(1-y)$ on 20,480 randomly sampled points on $[0, 1]^2$. As for the mortar space, the lowest order space $H = 1$ is used. Note that in this case, Assumption 1 is trivially satisfied.

In Fig. 10, we show the solutions of the pressure for $n = 2, 3, 5$. We see that we recover the true pressure easily as it is just a simple linear function. We note that the notion of convergence is not applicable in this case since the domain and problem itself are actually changing as we increase n.

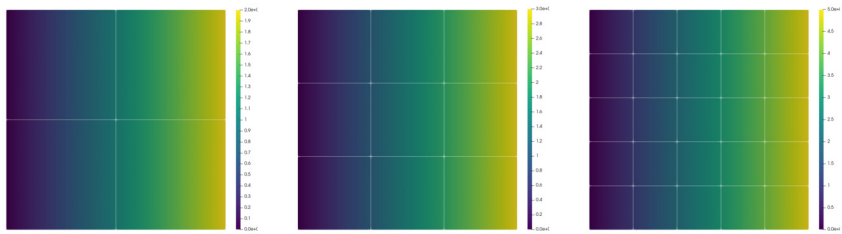


FIGURE 10 Figure of the pressure solutions obtained for the stripes problem Subsection 5.4.1 for on increasingly larger domain $[0, 2]^2$, $[0, 3]^2$, $[0, 5]^2$ using FEEC elements of 14×14 fine-scale knots and a very coarse mortar of $H = 1$. Importantly, we note that, from left to right, the domain Ω of the problem is being increased and we are not depicting a refinement process.

In Fig. 11, we show the x-component of the gradient; it is clear that the stripes structure is well-preserved even as we introduce more subdomains into the mortar space. While the error estimates Lemma 4.8 cannot support this statement due to the usage of crude L^∞ norms, this is indication that, at least

numerically, requirement **R2** is satisfied. We also plot the estimate solution profile on the line $(2.5, 0)$ to $(2.5, 5)$ in Fig. 12 for the case of $n = 5$. From this view, it's clear that the actual numerical values are in good agreement with the true solution.

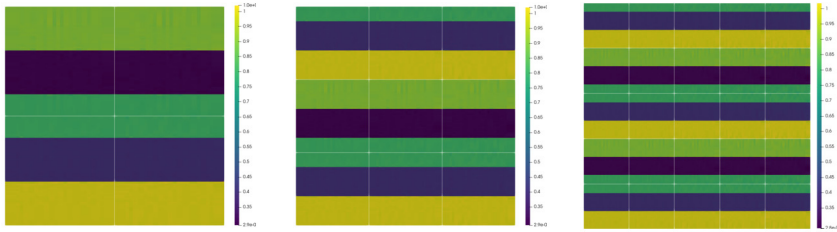


FIGURE 11 Figure of the flux in x of the solutions obtained for the stripes problem Subsection 5.4.1 on $[0, 2]^2$, $[0, 3]^3$, $[0, 5]^2$ with FEEC elements of 14×14 fine-scale knots and $H = 1$. We remark that the discontinuities are well-preserved using the FEEC elements even as the domain of the problem is increased.

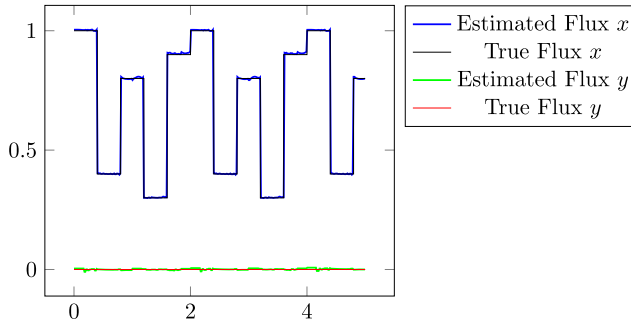


FIGURE 12 Profile on the line $(2.5, 0)$ to $(2.5, n)$ for $n = 5$ of the fluxes of the estimated and true stripes problem Subsection 5.4.1 obtained from the FEEC elements with 14×14 fine-scale knots and $H = 1$ mortar space.

5.4.2 *Path problem*

Consider data arising from the problem (4.1) on $\Omega = [0, 1] \times [0, 1]$ with $f = 0$, $g = x$ and

$$K = \begin{cases} \frac{1}{5}\mathbf{I} & x \in \Omega_{\text{path}} \\ \frac{1}{2}\mathbf{I} & x \in \Omega_{\text{circ}} \\ \mathbf{I} & x \in \text{elsewhere} \end{cases} \tag{5.6}$$

where Ω_{path} is defined as the region lying in

$$R((0, .625), (.375, .875)) \cup R((.375, .125), (.625, .875)) \cup R((.625, .125), (1, .375))$$

with $R(p_1, p_2)$ is the rectangle with lower left point p_1 and upper right corner p_2, and Ω_{circ} are two circles centered at (.125, .25) and (.875, .75) with radius .075. See the first column of Fig. 13 for figures of the true solution.

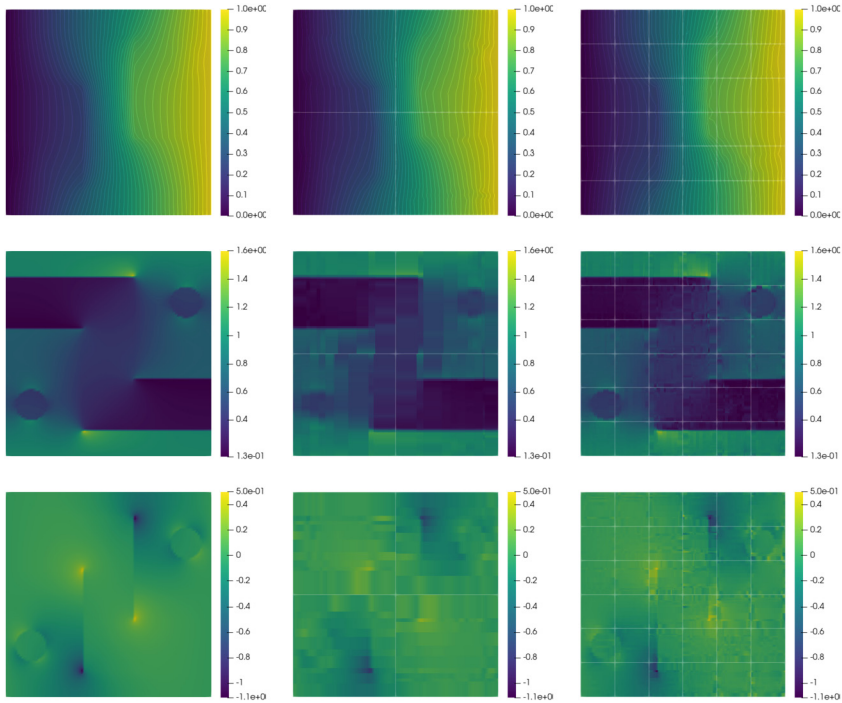


FIGURE 13 Plot of the true solution (first column), and subdomain with 2 (second column) and 6 (third column) refinements for the path problem Subsection 5.4.2. Note that the features are increasingly more refined and match the true solutions as the number of subdomains are increased.

Let our domain Ω be subdivided into n^2 equal squares as our subdomains, and let $H = \frac{1}{4n}$ meaning each subdomain has a total of 16 mortar degrees of freedom. On each of the subdomains, we train a FEEC element on 20,480 uniformly sampled points from the subdomain with 10 fine scale nodes and 14 POUs on the interior and boundary. As before, the FEEC elements are trained on 12 total boundary conditions corresponding to the third order Bernstein polynomials on squares. Rather than refining the mortar discretization relative to the number of subdomains, or increasing the fine-scale nodes on the local solvers, we *strictly increase the number of subdomains in this study.* We reiterate the fact that as the number of subdomains increases, the number of mortar degrees of freedom per subdomain remains the same at 16 and each FEEC element has the same number of parameters (e.g. 10 fine scale nodes and 14 POUs on the interior).

In Table 5, we show the average error resulting from increasing the number of subdomains over five different random seeds for training. We note that while

the error in the pressure is already captured quite accurately by a single FEEC element owing to its almost linear nature on the whole domain, the error in the gradient decreases much more dramatically, due to the higher resolution by increasing the number of subdomains.

TABLE 5 Table of average absolute and relative errors for the path problem Subsection 5.4.2 whereby the domain is increasingly subdivided into finer pieces. While the pressure does not exhibit convergence, the flux converges at a rate of $\mathcal{O}(h)$ and so does the full H^1 norm (see Fig. 14).

Subdomains	Mean $\|p - p_h\|_\Omega$	Mean $\|u - u_h\|_\Omega$
2×2	3.53E-03 (0.596%)	3.47E-02 (5.21%)
3×3	3.25E-03 (0.549%)	2.49E-02 (3.74%)
4×4	3.13E-03 (0.528%)	1.56E-02 (2.34%)
5×5	2.97E-03 (0.501%)	1.33E-02 (1.99%)
6×6	2.70E-03 (0.456%)	9.80E-03 (1.47%)
8×8	3.28E-03 (0.554%)	6.50E-03 (0.97%)

In Fig. 14, we plot the H^1 norm errors of both the individual seeds and the mean. We observe a first-order convergence in the number of subdomains, supporting the notion that our mortar method satisfies requirement **R2** as we increase the number of elements. Unfortunately, the error analysis performed in the previous section is not fine enough to show convergence in this case where we increase the number of subdomains due to the usage of crude triangle inequalities.

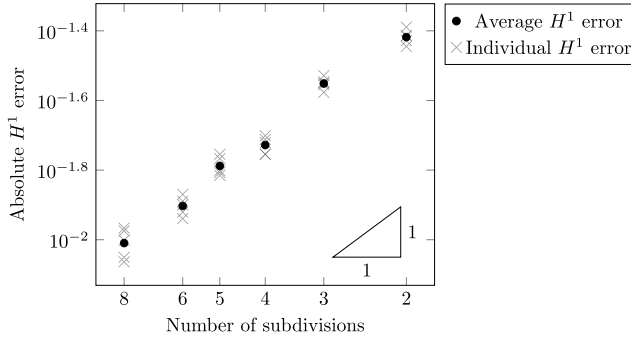


FIGURE 14 Plot of the absolute H^1 error resulting from refinement for the path problem Subsection 5.4.2. We observe a linear convergence rate by dividing the domain into increasingly smaller domains for the FEEC problem.

5.4.3 *Battery problem: single solution training*

We now consider data from the problem (4.1) on $\Omega = [0, 1] \times [0, 1]$ with $f = 0$ and a nontrivial material data and boundary condition corresponding to

a voltage difference across a lithium-ion battery. The true pressure and fluxes, which are sampled at 5.89 million points, are provided via a high-fidelity solver SIERRA/ARIA (Notz et al., 2016) and will be treated as the *only* source of provided data with no additional methods of augmentation. In other words, we assume a full simulation of the response for the subdomains to arbitrary mortars is not available, meaning the local FEEC elements will have to extrapolate the correct Dirichlet-to-Neumann maps. For a figure of the true pressure and flux, see Fig. 15. More details regarding the data can be found in Appendix B of Actor et al. (2024); note that for simplicity, we consider the problem as a purely Dirichlet boundary condition problem whilst (Actor et al., 2024) included Neumann boundary conditions.

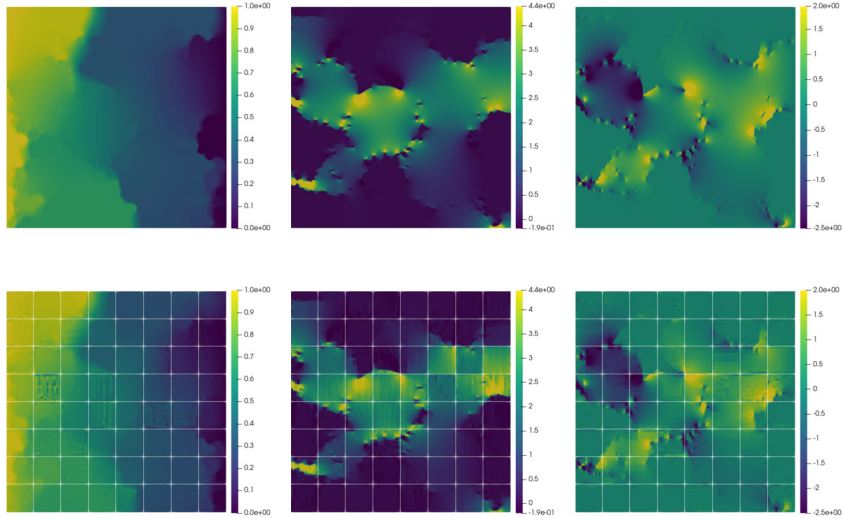


FIGURE 15 Figures of the given pressure/fluxes for the battery problem Subsection 5.4.3 in the first row, and the approximation obtained from solving the Darcy's flow problem in the second row for 8×8 subdivisions. Overall, the estimated solution matches the data fairly accurately with many most small details captured.

We again split the domain $[0, 1]^2$ into n^2 uniform squares, but only employ four mortar degrees of freedom per subdomain with $H = \frac{1}{n}$ (i.e. the mortar degrees of freedom lie on the corners of the subdomain).[2] A FEEC element with 12 fine scale nodes in both the x and y direction, and 12 POUs on the interior and boundary are used on each subdomain.

Since only a single reference solution is provided, we train the FEEC element with boundary condition obtained from interpolating the given solution and the data given (e.g. $g_i = p|_{\partial\Omega_i}$). For example, suppose $n = 2$, then the FEEC element on subdomain corresponding to $\Omega' = [0, .5]^2$ will have $\frac{5.89}{4} \approx 1.5$ mil-

[2] The coarsest mortar mesh is chosen since the fine scale nodes may move substantially, due to only one training set, and violate assumption (4.21).

lion data points, and boundary conditions corresponding to the nearest neighbor interpolation of those points on $\partial\Omega'$. This is unlike Subsection 5.4.2 or even Subsection 5.4.1 where each FEEC element was provided with multiple examples to train on. Note that the number of training data points per FEEC element decrease as we increase the number of subdomains, we have found that it can lead to some instability in pretraining.

TABLE 6 Table of absolute and relative errors for the battery problem Subsection 5.4.3. The right "true mortar" (TM) columns essentially capture the training error by simply fixing the mortar space to the true values, while the left columns result from actually solving the Darcy's flow equations. Similar to Subsection 5.4.2, the error in pressure only decreases slightly with most of the benefits arising from the convergence in the H^1-seminorm.

Subdomains	$L^2(\Omega)$	H^1-seminorm	TM $L^2(\Omega)$	TM H^1-seminorm
2×2	7.15E-03 (1.27%)	1.22E+00 (67.6%)	5.37E-03	1.39E+00
3×3	3.01E-03 (0.54%)	6.45E-01 (35.7%)	2.76E-03	6.03E-01
4×4	2.71E-03 (0.48%)	4.57E-01 (25.3%)	2.42E-03	3.27E-01
6×6	2.46E-03 (0.44%)	2.44E-01 (13.5%)	1.67E-03	1.37E-01
8×8	2.40E-03 (0.43%)	1.43E-01 (7.92%)	1.41E-03	1.19E-01

In Table 6, we show the absolute MSE of the L^2 and the H^1 seminorm resulting from solving the Darcy's flow equation with the trained FEEC elements. In the case of 2×2 refinement, the error is quite large since the mortar only has one degree of freedom in the interior (cf. Fig. 4) and the boundary conditions are not even well-resolved; however, it's clear that as additional refinements are made that the relative error decreases. In addition, we also show the absolute MSE of the "true mortar" (TM) which is obtained by setting the mortar degrees of freedom to be the interpolant from the data set. This "true mortar" indicates how much of the error is due to the training procedure as no actual solves of the bilinear form is performed and allows us to see how much error arises from the actual mortar coupling. Since this true mortar errors are similar to the errors obtained from solving the bilinear form, this suggests that very little error arises due to the mortar coupling. In Fig. 16, we observe that the error obtained from solving the Darcy flow equation decreases as we increase the number of subdomains, with the finest level obtaining a better H^1 error than the errors obtained in Actor et al. (2024).

Appendix 10.A Technical details

10.A.1 Technical proofs

Proof of Lemma 4.4. By (4.5), for any $\mu \in H_0^{\gamma}(\Omega)$, we can decompose it as

$$\mu = p^*(\mu) + \sum_{i=1}^{n} p_{i0}$$

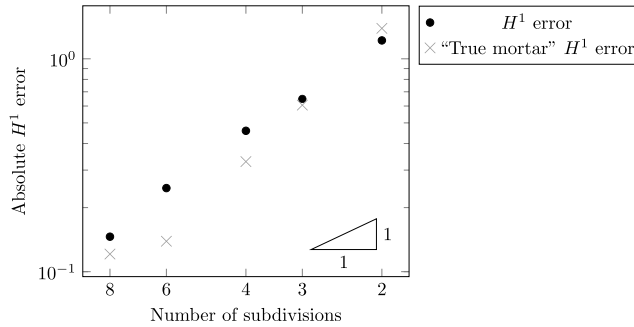


FIGURE 16 Plot of the H^1 error and true mortar error resulting from refinement for the battery problem Subsection 5.4.3. For an explanation of what the true mortar error is, we refer the reader to the corresponding discussion Subsection 5.4.3. We note that the error is quite close to the true mortar error, meaning that the coarse mortar space does not negatively effect the convergence that much.

where $p^*(\mu) \in H_0^1(\Omega)$ satisfying (4.9) (hence $p^*(\mu)|_\Gamma = \mu|_\Gamma$) and $p_{i0} \in H_0^1(\Omega_i)$ are bubble functions.

Thus,

$$\begin{aligned} b(\lambda, \mu) &= \sum_{i=1}^{n} \left(\boldsymbol{u}^*(\lambda), \nabla p^*(\mu) + \sum_{i=1}^{n} \nabla p_{i0} \right)_{\Omega_i} \\ &= \sum_{i=1}^{n} \left(\boldsymbol{u}^*(\lambda), \nabla p^*(\mu) \right)_{\Omega_i} = \sum_{i=1}^{n} \left(K \nabla p^*(\lambda), \nabla p^*(\mu) \right)_{\Omega_i} \end{aligned} \tag{10.A.1}$$

since (4.9) implies the inner products of $\boldsymbol{u}^*(\lambda)$ with the gradient of bubble functions are zero. From the above, the bilinear form is clearly symmetric and positive definite.

For coercivity, using (10.A.1), Poincare inequality and trace inequality (Evans, 2022),

$$\begin{aligned} b(\lambda, \lambda) &\geq \left\| \nabla p^*(\lambda) \right\|^2 \geq \frac{1}{C} \left\| p^*(\lambda) \right\|^2_{H^1(\Omega)} = \frac{1}{C} \sum_{i=1}^{n} \left\| p^*(\lambda) \right\|^2_{H^1(\Omega_i)} \\ &\geq \frac{1}{C} \sum_{i=1}^{n} \|\lambda\|^2_{H^{1/2}(\Gamma_i)}, \end{aligned}$$

meaning $b(\lambda, \lambda) \geq \alpha \sum_{i=1}^{n} \|\lambda\|^2_{H^{1/2}(\Gamma_i)} \sim \alpha \|\lambda\|^2_{H^\gamma}$ for some constant α independent of λ. □

The remaining proofs are for the coercivity and the error estimate for the discrete mortar. We introduce the shorthand notation $p_h^*(Q\lambda_H) :=$

$\sum_{i=1}^{n} p_h^*(Q_i\lambda_H)$, and let $\|\cdot\|_\Omega$ denote the L^2 norm over the domain Ω unless otherwise stated. Before proceeding, we define the inclusion map $P_i : \Lambda_H \subset \Lambda \to H^\gamma|_{\Omega_i}$ through the isomorphism. We need a preparatory lemma:

Lemma 10.A.1. *Let $\delta := \frac{2}{C_p+1}$, where C_p is the constant arising from Corollary 6.3 of Brenner (2003), then*

$$\begin{aligned}
&\frac{\delta}{2}\left\| p_h^*(Q\lambda_H)\right\|^2_{H^1(\Omega)} \\
&\leq \left[\left\|\nabla p_h^*(Q\lambda_H)\right\|^2_\Omega + \frac{C_p}{C_p+1}\sum_{\Gamma_{ij}} \frac{1}{|\Gamma_{ij}|}\left\| Q_i\lambda_H - Q_j\lambda_H\right\|^2_{L^2(\Gamma_{ij})}\right].
\end{aligned}$$

Proof. By a simple application of Corollary 6.3 of Brenner (2003):

$$\begin{aligned}
&\left\| p_h^*(Q\lambda_H)\right\|^2_{H^1(\Omega)} \\
&\leq C_p\left[(1+\frac{1}{C_p})\left\|\nabla p_h^*(Q\lambda_H)\right\|^2_\Omega + \sum_{\Gamma_{ij}} \frac{1}{|\Gamma_{ij}|^2}\left(\int_{\Gamma_{ij}} Q_i\lambda_H - Q_j\lambda_H\, ds\right)^2\right] \\
&\leq C_p\left[(1+\frac{1}{C_p})\left\|\nabla p_h^*(Q\lambda_H)\right\|^2_\Omega + \sum_{\Gamma_{ij}} \frac{1}{|\Gamma_{ij}|}\left\| Q_i\lambda_H - Q_j\lambda_H\right\|^2_{L^2(\Gamma_{ij})}\right]
\end{aligned}$$

where we used Cauchy-Schwarz inequality on $(\int_\sigma f)^2 \leq |\sigma|\left\| f\right\|^2$. □

Proof of Lemma 4.5. An identity like (10.A.1) can also be verified for the discrete version as well since on any subdomain i and $\mu_H \in \Lambda_H$, $Q_i\mu_H = p_h^*(Q_i\mu_H) + p_{hi}$ where $p_{hi} \in W_{hi,0}$ bubble functions, one has

$$\begin{aligned}
b_h(\lambda_H, \mu_H) &= \sum_{i=1}^{n} \left(\boldsymbol{u}_h^*(Q_i\lambda_H), \nabla p_h^*(Q_i\mu_H) + \nabla p_{hi}\right)_{\Omega_i} \\
&= \sum_{i=1}^{n} \left(\boldsymbol{u}_h^*(Q_i\lambda_H), \nabla p_h^*(Q_i\mu_H)\right)_{\Omega_i} \qquad (10.A.2) \\
&= \sum_{i=1}^{n} \left(K\nabla p_h^*(Q_i\lambda_H), \nabla p_h^*(Q_i\mu_H)\right)_{\Omega_i}.
\end{aligned}$$

Thus, the bilinear form b_h is symmetric, and, at least, positive semidefinite. Coercivity requires a bit more work.

Since for each subdomain i, $p_h^*(Q_i\lambda_H)|_{\Gamma_i} = Q_i\lambda_H|_{\Gamma_i}$, we add by zero and expand to obtain

$$\begin{aligned}
b_h(\lambda_H,\lambda_H) &= \sum_{i=1}^{n} \left(K\nabla p_h^*(Q_i\lambda_H), \nabla p_h^*(Q_i\lambda_H)\right)_{\Omega_i} \\
&\quad + \delta\langle Q_i\lambda_H - p_h^*(Q_i\lambda_H), Q_i\lambda_H\rangle_{H^{1/2}(\Gamma_i)} \\
&\geq \sum_{i=1}^{n} \left\|\nabla p_h^*(Q_i\lambda_H)\right\|_{\Omega_i}^2 + \delta\|Q_i\lambda_H\|_{H^{1/2}(\Gamma_i)}^2 \\
&\quad - \frac{\delta}{2}\left\|p_h^*(Q_i\lambda_H)\right\|_{H^{1/2}(\Gamma_i)}^2 - \frac{\delta}{2}\|Q_i\lambda_H\|_{H^{1/2}(\Gamma_i)}^2 \\
&\geq \sum_{i=1}^{n} \left\|\nabla p_h^*(Q_i\lambda_H)\right\|_{\Omega_i}^2 + \frac{\delta}{2}\|Q_i\lambda_H\|_{H^{1/2}(\Gamma_i)}^2 \\
&\quad - \frac{\delta}{2}\left\|p_h^*(Q_i\lambda_H)\right\|_{H^1(\Omega_i)}^2 \\
&= \left\|\nabla p_h^*(Q\lambda_H)\right\|_{\Omega}^2 - \frac{\delta}{2}\left\|p_h^*(Q\lambda_H)\right\|_{H^1(\Omega)}^2 \\
&\quad + \sum_{\Gamma_{ij}} \frac{\delta}{2}(\|Q_i\lambda_H\|_{H^{1/2}(\Gamma_{ij})}^2 + \left\|Q_j\lambda_H\right\|_{H^{1/2}(\Gamma_{ij})}^2)
\end{aligned}$$

by using Cauchy-Schwarz, the trace inequality, and the trivial inequality $ab \leq 2a^2 + 2b^2$.

Now, using the assumption (4.22)

$$\begin{aligned}
b_h(\lambda_H,\lambda_H) &\geq \left\|\nabla p_h^*(Q\lambda_H)\right\|^2 - \frac{\delta}{2}\left\|p_h^*(Q\lambda_H)\right\|_{H^1(\Omega)}^2 \\
&\quad + \sum_{\Gamma_{ij}} \frac{\delta C_p}{2|\Gamma_{ij}|}\left\|Q_i\lambda_H - Q_j\lambda_H\right\|_{L^2(\Gamma_{ij})}^2 \\
&\quad + \sum_{\Gamma_{ij}} \frac{\delta}{4}(\|Q_i\lambda_H\|_{H^{1/2}(\Gamma_{ij})}^2 + \left\|Q_j\lambda_H\right\|_{H^{1/2}(\Gamma_{ij})}^2) \\
&= \left\|\nabla p_h^*(Q\lambda_H)\right\|^2 - \frac{\delta}{2}\left\|p_h^*(Q\lambda_H)\right\|_{H^1(\Omega)}^2 \\
&\quad + \sum_{\Gamma_{ij}} \frac{C_p}{(C_p+1)|\Gamma_{ij}|}\left\|Q_i\lambda_H - Q_j\lambda_H\right\|_{L^2(\Gamma_{ij})}^2 \\
&\quad + \sum_{\Gamma_{ij}} \frac{\delta}{4}(\|Q_i\lambda_H\|_{H^{1/2}(\Gamma_{ij})}^2 + \left\|Q_j\lambda_H\right\|_{H^{1/2}(\Gamma_{ij})}^2)
\end{aligned}$$

Then, by Lemma 10.A.1 and the assumption (4.21)

$$\begin{aligned} b_h(\lambda_H, \lambda_H) &\geq \sum_{\Gamma_{ij}} \frac{\delta}{4} (\| Q_i \lambda_H \|^2_{H^{1/2}(\Gamma_{ij})} + \| Q_j \lambda_H \|^2_{H^{1/2}(\Gamma_{ij})}) \\ &\geq \sum_{i=1}^{n} \frac{\delta}{4} \| Q_i \lambda_H \|^2_{H^{1/2}(\Gamma_i)} \geq \frac{\delta}{4} \sum_{i=1}^{n} \| \lambda_H \|^2_{H^{1/2}(\Gamma_i)} . \end{aligned}$$ □

For the sake of notation, we assume that $\|\cdot\|_{H^{1/2}}$ denote the sum of the $H^{1/2}$ norms over all the Γ_i unless otherwise denoted:

Proof of Theorem 4.6. By Strang's second lemma, there exists a constant C such that

$$\begin{aligned} &\left\| \lambda^* - \lambda_h^* \right\|_{H^{1/2}} \\ &\leq C \left(\inf_{\mu_H \in \Lambda_0} \left\| \lambda^* - \mu_H \right\|_{H^{1/2}} + \sup_{\mu_H \in \Lambda_0} \frac{\left| b_h(\lambda^*, \mu_H) - L_h(\mu_H) \right|}{\left\| \mu_H \right\|_{H^{1/2}}} \right). \end{aligned}$$

The first term, otherwise known as the approximation error, is bounded by our assumption that $\left| p \right|_{H^2} < \infty$, meaning that the traces on the interior are at least in $H^{3/2}(\Gamma_i)$ for all $1 \leq i \leq n$, hence

$$\inf_{\mu_H \in \Lambda_0} \left\| \lambda^* - \mu_H \right\|_{H^{1/2}} \leq H \sum_{i=1}^{n} \left\| \lambda^* \right\|_{H^{3/2}(\Gamma_i)} \leq H \left| p \right|_{H^2(\Omega)}$$

by standard approximation results.

For the consistency error, we substitute the definition into the definition of our bilinear form and linear functional in, and noting that $-\nabla K \nabla p = f$ by definition of our problem, we have for all $\mu_H \in \Lambda_0$

$$\begin{aligned} &\frac{\left| b_h(\lambda^*, \mu_H) - L_h(\mu_H) \right|}{\left\| \mu_H \right\|_{H^{1/2}}} \\ &= \frac{\left| \sum_{i=1}^{n} (\boldsymbol{u}_h^*(Q_i \lambda^*) + \bar{\boldsymbol{u}}_h, \nabla(Q_i \mu_H)) - (f, Q_i \mu_H) \right|}{\left\| \mu_H \right\|_{H^{1/2}}} \\ &= \frac{\left| \sum_{i=1}^{n} (K \nabla(p_h(\lambda^*) - p), \nabla Q_i \mu_H) + (K \nabla p, \nabla Q_i \mu_H) - (-\nabla K \nabla p, Q_i \mu_H) \right|}{\left\| \mu_H \right\|_{H^{1/2}}} \\ &= \frac{\left| \sum_{i=1}^{n} (K \nabla(p_h(\lambda^*) - p), \nabla Q_i \mu_H) - \int_{\Gamma_i} K \nabla p Q_i \mu_H \cdot \boldsymbol{n}_i \, ds \right|}{\left\| \mu_H \right\|_{H^{1/2}}} \end{aligned}$$

where $\boldsymbol{n}_i$ is the outward normal to the subdomain Ω_i, and $p_h(\lambda^*) := p^*(Q_i\lambda^*) + \bar{p}$. The first term can be estimate using Cauchy-Schwarz inequality,

$$\begin{aligned}
&\frac{\left|\sum_{i=1}^n (K\nabla(p_h(\lambda^*) - p), \nabla Q_i\mu_H)\right|}{\left\|\mu_H\right\|_{H^{1/2}}} \\
&\leq \frac{\sum_{i=1}^n \left\|K\nabla(p_h(\lambda^*) - p)\right\|_{\Omega_i} \left\|\nabla Q_i\mu_H\right\|_{\Omega_i}}{\left\|\mu_H\right\|_{H^{1/2}}} \\
&\leq n \max_i \left\|K\nabla(p_h(\lambda^*) - p)\right\|_{\Omega_i} \frac{\sum_{i=1}^n \left\|\nabla Q_i\mu_H\right\|_{\Omega_i}}{\left\|\mu_H\right\|_{H^{1/2}}} \\
&\leq Cn \max_i \left\|K\nabla(p_h(\lambda^*) - p)\right\|_{\Omega_i} \leq Cn\delta,
\end{aligned}$$

where we use the fact that

$$\left\|\nabla Q_i\mu_H\right\|_{\Omega_i} \leq \left\|Q_i\mu_H\right\|_{H^1(\Omega_i)} \leq C\left\|Q_i\mu_H\right\|_{H^{1/2}(\Gamma_i)} \leq C\left\|\mu_H\right\|_{H^{1/2}(\Gamma_i)}$$

where we used the properties of discrete harmonic extensions (Toselli and Widlund, 2004), and the fact that L^2 projection is stable in $H^{1/2}$ due to interpolation (Bramble and Xu, 1991).

As for the second term, we note that if two subdomains Ω_i, Ω_j are adjacent, then $\boldsymbol{n}_i = -\boldsymbol{n}_j$ meaning

$$\begin{aligned}
&\frac{\left|\sum_{i=1}^n \int_{\Gamma_i} K\nabla p Q_i\mu_H \cdot \boldsymbol{n}_i \, ds\right|}{\left\|\mu_H\right\|_{H^{1/2}}} \\
&\leq \frac{\sum_{\Gamma_{ij}} \left|\int_{\Gamma_{ij}} K\nabla p(Q_i\mu_H - Q_j\mu_H)\cdot\boldsymbol{n}\right|_i}{\left\|\mu_H\right\|_{H^{1/2}}} \\
&\leq \frac{\sum_{\Gamma_{ij}} \left\|K\nabla p\cdot\boldsymbol{n}_i\right\|_{H^{1/2}(\Gamma_{ij})} \left\|Q_i\mu_H - Q_j\mu_H\right\|_{H^{-1/2}(\Gamma_{ij})}}{\left\|\mu_H\right\|_{H^{1/2}}} \\
&\leq Cn\left|p\right|_{H^2(\Omega)} \max_i \frac{\left\|(I - Q_i)\mu_H\right\|_{H^{-1/2}(\Gamma_{ij})}}{\left\|\mu_H\right\|_{H^{1/2}}} \\
&\leq Cn\left|p\right|_{H^2(\Omega)} \max_i h_i
\end{aligned}$$

where we used the inequality $\left\|Q_i - Q_j\right\| \leq \|I - Q_i\| + \left\|I - Q_j\right\|$, the trace inequality on normal derivatives (Grisvard, 2011, Thm. 1.5.1.2), L^2 projection approximation properties (Arbogast et al., 2007, (3.5)), and where h_i denotes the maximal mesh-size on each subdomain Ω_i. □

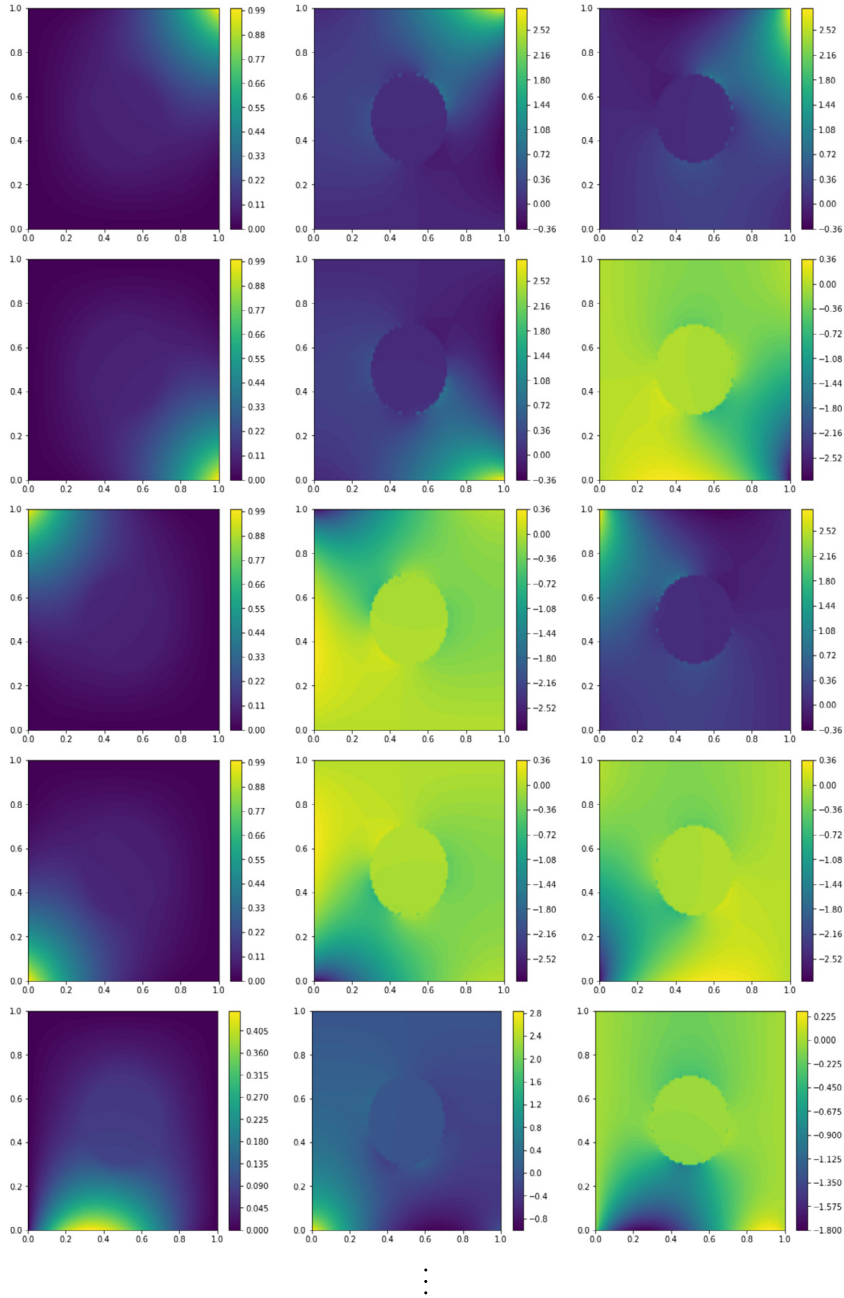


FIGURE 17 Plots illustrating some of the training data used for Subsection 5.3 with pressure, flux of x and flux of y in the columns respectively. The data is generated from a low order FEM method with $h = 1/100$. The key differences between each data set are that the boundary conditions are varied so that the element can respond to the different mortars.

10.A.2 FEEC element training

For each of the FEEC elements used in Examples 2 through 4 with the exception of the battery example (discussed below), a "monolithic" approach is used. For concreteness, we will exposit the details fully for Example 3 as the other examples only differ by model hyperparameters described in the relevant section and the training data.

The data used to train the FEEC elements are generated from 20,480 randomly sampled points from $[0, 1]^2$ evaluated by interpolating the solution of an elementary finite element solver. In the case of the FEEC element in Example 3, a grand total of 12 different solutions each with different boundary conditions, corresponding to the third-order Bernstein polynomials on the boundary (e.g. x^3y^3, $\binom{3}{1}x^3y^2(1-y)$, $\binom{3}{2}x^3y(1-y)^2$ etc), are used alongside the forcing term of $f = 0$. The Bernstein polynomials were used instead of simple hat functions as we found the additional smoothness meant pretraining of the FEEC element was more stable. In Fig. 17, we plot the first five, out of twelve, of the training data we generated for Subsection 5.3.

Let ξ correspond to all the hyperparameters in the FEEC model (e.g. knot location, POU coefficients, scaling coefficients). The loss function we use is

$$\min_{\xi} \sum_{k=1}^{12} \frac{\left\| p_{\xi,k} - p_{\text{data},k} \right\|_{MSE}}{\left\| p_{\text{data},k} \right\|_{\ell_2}} + \frac{\left\| \boldsymbol{u}_{\xi,k} - \boldsymbol{u}_{\text{data},k} \right\|_{MSE}}{\left\| \boldsymbol{u}_{\text{data},k} \right\|_{\ell_2} + 0.001} \tag{10.A.3}$$

where $p_{\xi,k}$, $\boldsymbol{u}_{\xi,k}$ are the FEEC solutions with the kth boundary condition, and $p_{\text{data},k}$, $\boldsymbol{u}_{\text{data},k}$ are the data for the kth boundary condition subject to the constraint. This is exactly (3.7), except we summed over all the different boundary conditions and minimized against all the boundary conditions in a single epoch (e.g. a monolithic approach). The computation of the loss is efficient since $p_{\xi,k}$ for $k = 1, \ldots, 12$ can be solved with a single linear solver step because their systems only differ in their right hand sides from the boundary conditions. Thus, the expensive stiffness matrix generation only has to be performed once at each optimization step. The standard Adams optimizer were used in each case as discussed in Actor et al. (2024).

As a result of the monolithic training and the basis generation of FEEC, the FEEC element will be able to accurately solve for the flux and pressure even when faced with Dirichlet boundary conditions which it has not seen before. For example, in Fig. 18, we plot the true and predicted solution of (5.4) with a boundary condition of y. Note that, while the boundary condition was never explicitly given in the training data, that the FEEC element was able to reproduce the behavior around the material discontinuity quite accurately.

As noted in Subsection 5.4.3, the battery example assumes only a single data set is available, with no additional data generation with varying boundary conditions as above. The data for each subdomain are simply obtained via a restriction operator, and the loss is exactly (3.7).

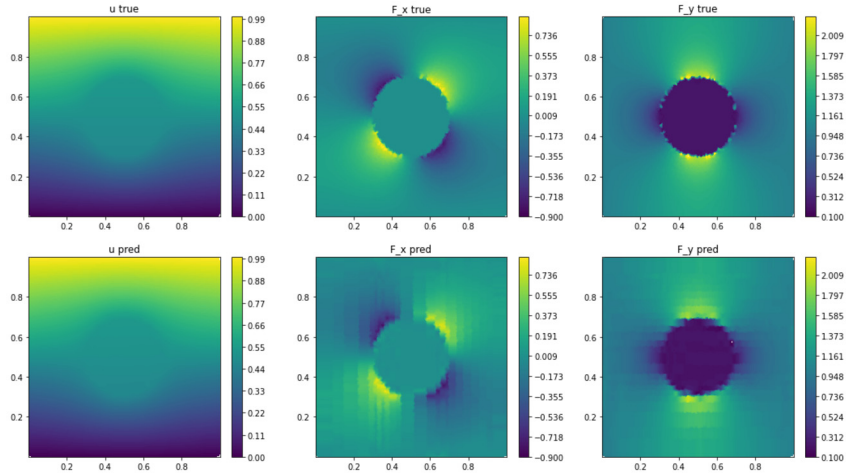


FIGURE 18 Plot of the true (first row) and predicted using a 24×24 fine-scale FEEC element (second row) solution to (5.4) with the boundary condition y on the domain $[-.5, .5]^2$. Note that while the boundary condition is not explicitly included in the training data, but rather a linear combination, we are able to reproduce the true solution accurately due to training against a large suite of boundary conditions.

References

Actor, J.A., Hu, X., Huang, A., Roberts, S.A., Trask, N., 2024. Data-driven Whitney forms for structure-preserving control volume analysis. Journal of Computational Physics 496, 112520.

Arbogast, T., Cowsar, L.C., Wheeler, M.F., Yotov, I., 2000. Mixed finite element methods on nonmatching multiblock grids. SIAM Journal on Numerical Analysis 37, 1295–1315.

Arbogast, T., Pencheva, G., Wheeler, M.F., Yotov, I., 2007. A multiscale mortar mixed finite element method. Multiscale Modeling & Simulation 6, 319–346.

Arnold, D.N., 2018. Finite Element Exterior Calculus. SIAM.

Bergomi, M.G., Frosini, P., Giorgi, D., Quercioli, N., 2019. Towards a topological–geometrical theory of group equivariant non-expansive operators for data analysis and machine learning. Nature Machine Intelligence 1, 423–433.

Bernardi, C., Maday, Y., Patera, A.T., 1993. Domain decomposition by the mortar element method. In: Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters. Springer, pp. 269–286.

Bertoluzza, S., Kunoth, A., 2000. Wavelet stabilization and preconditioning for domain decomposition. IMA Journal of Numerical Analysis 20, 533–559.

Biegler, L.T., Ghattas, O., Heinkenschloss, M., van Bloemen Waanders, B., 2003. Large-scale pde-constrained optimization: an introduction. In: Large-Scale PDE-Constrained Optimization. Springer, pp. 3–13.

Braess, D., 2007. Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics. Cambridge University Press.

Bramble, J.H., Xu, J., 1991. Some estimates for a weighted l^2 projection. Mathematics of Computation 56, 463–476.

Brenner, S.C., 2003. Poincaré–Friedrichs inequalities for piecewise h 1 functions. SIAM Journal on Numerical Analysis 41, 306–324.

Cai, S., Mao, Z., Wang, Z., Yin, M., Karniadakis, G.E., 2021. Physics-informed neural networks (pinns) for fluid mechanics: a review. Acta Mechanica Sinica 37, 1727–1738.

Celledoni, E., Ehrhardt, M.J., Etmann, C., McLachlan, R.I., Owren, B., Schonlieb, C.-B., Sherry, F., 2021. Structure-preserving deep learning. European Journal of Applied Mathematics 32, 888–936.

Cockburn, B., Gopalakrishnan, J., Lazarov, R., 2009. Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. SIAM Journal on Numerical Analysis 47, 1319–1365.

Cowsar, L.C., Mandel, J., Wheeler, M.F., 1995. Balancing domain decomposition for mixed finite elements. Mathematics of Computation 64, 989–1015.

Desai, S.A., Mattheakis, M., Sondak, D., Protopapas, P., Roberts, S.J., 2021. Port-Hamiltonian neural networks for learning explicit time-dependent dynamical systems. Physical Review E 104, 034312.

Evans, L.C., 2022. Partial Differential Equations, vol. 19. American Mathematical Society.

Farhat, C., Lesoinne, M., LeTallec, P., Pierson, K., Rixen, D., 2001. Feti-dp: a dual–primal unified feti method—part I: a faster alternative to the two-level feti method. International Journal for Numerical Methods in Engineering 50, 1523–1544.

Gillette, A., Rand, A., Bajaj, C., 2016. Construction of scalar and vector finite element families on polygonal and polyhedral meshes. Computational Methods in Applied Mathematics 16, 667–683.

Glowinski, R., Wheeler, M.F., 1987. Domain decomposition and mixed finite element methods for elliptic problems. Tech. Report.

Greydanus, S., Dzamba, M., Yosinski, J., 2019. Hamiltonian neural networks. Advances in Neural Information Processing Systems 32.

Grisvard, P., 2011. Elliptic Problems in Nonsmooth Domains. SIAM.

Gruber, A., Lee, K., Trask, N., 2023. Reversible and irreversible bracket-based dynamics for deep graph neural networks. arXiv preprint. arXiv:2305.15616.

Hernández, Q., Badías, A., González, D., Chinesta, F., Cueto, E., 2021. Structure-preserving neural networks. Journal of Computational Physics 426, 109950.

Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S., 2008. Optimization with PDE Constraints, vol. 23. Springer Science & Business Media.

Hirani, A.N., 2003. Discrete Exterior Calculus. California Institute of Technology.

Jagtap, A.D., Karniadakis, G.E., 2021. Extended physics-informed neural networks (xpinns): a generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations. In: AAAI Spring Symposium: MLPS, vol. 10.

Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L., 2021. Physics-informed machine learning. Nature Reviews Physics 3, 422–440.

Nicolaides, R.A., 1992. Direct discretization of planar div-curl problems. SIAM Journal on Numerical Analysis 29, 32–56.

Notz, P.K., Subia, S.R., Hopkins, M.M., Moffat, H.K., Noble, D.R., Okusanya, T.O., 2016. Sierra multimechanics module: aria user manual–version 4.40. Tech. Report. Sandia National Lab. (SNL-NM), Albuquerque, NM (United States).

Patel, R.G., Manickam, I., Trask, N.A., Wood, M.A., Lee, M., Tomas, I., Cyr, E.C., 2022. Thermodynamically consistent physics-informed neural networks for hyperbolic systems. Journal of Computational Physics 449, 110754.

Roberts, J.E., Thomas, J.-M., 1991. Mixed and hybrid methods.

Smith, B.F., 1997. Domain decomposition methods for partial differential equations. In: Parallel Numerical Algorithms. Springer, pp. 225–243.

Smythe, W.R.w.R., 1989. Static and Dynamic Electricity. International Series in Pure and Applied Physics. Taylor & Francis, Philadelphia, PA.

Toselli, A., Widlund, O., 2004. Domain Decomposition Methods-Algorithms and Theory, vol. 34. Springer Science & Business Media.

Trask, N., Huang, A., Hu, X., 2022. Enforcing exact physics in scientific machine learning: a data-driven exterior calculus on graphs. Journal of Computational Physics 456, 110969.

Trask, N., Maxey, M., Hu, X., 2018. A compatible high-order meshless method for the Stokes equations with applications to suspension flows. Journal of Computational Physics 355, 310–326.

Villar, S., Hogg, D.W., Storey-Fisher, K., Yao, W., Blum-Smith, B., 2021. Scalars are universal: equivariant machine learning, structured like classical physics. Advances in Neural Information Processing Systems 34, 28848–28863.

Wang, S., Teng, Y., Perdikaris, P., 2021. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. SIAM Journal on Scientific Computing 43, A3055–A3081.

Yang, Y., Perdikaris, P., 2019. Adversarial uncertainty quantification in physics-informed neural networks. Journal of Computational Physics 394, 136–152.

Zhang, D., Lu, L., Guo, L., Karniadakis, G.E., 2019. Quantifying total uncertainty in physics-informed neural networks for solving forward and inverse stochastic problems. Journal of Computational Physics 397, 108850.