Hybrid coupling with operator inference and the overlapping Schwarz alternating method

Irina Tezaur[1]*, Eric Parish[1], Anthony Gruber[1], Ian Moore[1,2],
Christopher Wentland[1], Alejandro Mota[1]

[1]Sandia National Laboratories, Livermore, CA, USA

[2]Virginia Polytechnic Institute and State University, Blacksburg, VA, USA

Abstract. This paper presents a novel hybrid approach for coupling subdomain-local non-intrusive Operator Inference (OpInf) reduced order models (ROMs) with each other and with subdomain-local high-fidelity full order models (FOMs) with using the overlapping Schwarz alternating method (O-SAM). The proposed methodology addresses significant challenges in multiscale modeling and simulation, particularly the long runtime and complex mesh generation requirements associated with traditional high-fidelity simulations. By leveraging the flexibility of O-SAM, we enable the seamless integration of disparate models, meshes, and time integration schemes, enhancing computational efficiency while maintaining high accuracy. Our approach is demonstrated through a series of numerical experiments on complex three-dimensional (3D) solid dynamics problems, showcasing speedups of up to $106\times$ compared to conventional FOM-FOM couplings. This work paves the way for more efficient simulation workflows in engineering applications, with potential extensions to a wide range of partial differential equations.

Keywords. Schwarz alternating method (SAM), overlapping domain decomposition (DD), multiscale coupling, solid dynamics, operator inference (OpInf), reduced order model (ROM).

1 Introduction

Multiscale modeling and simulation (mod/sim) is crucial in engineering, as it enables the design, optimization and qualification of complex engineered components and systems across a variety of applications, enhancing the understanding of sophisticated multiscale and multiphysics processes. Unfortunately, analysts running traditional high-fidelity simulation codes often face significant delays due not only to long runtime requirements, but also due to the mesh generation step of the mod/sim workflow: creating a high-quality mesh for a single multiscale component can take weeks, making it "the single biggest bottleneck in [mod/sim analyses]" [24]. During the past two decades, projection-based reduced order models (ROMs) have emerged as a promising data-driven tool that has the potential to reduce the online computational complexity of numerical simulations, especially in multi-query applications, such as design optimization or uncertainty quantification (UQ); however, these models can face their own shortcomings, including a lack of systematic refinement mechanisms, a lack of stability or accuracy in the predictive regime, and lengthy implementation time requirements. While recently-proposed physics-informed neural networks (PINNs) [46] and Physics-Informed Deep Operator Neural Networks (PI-DeepONets) [32] have been advertised as mesh-free methods that can bypass meshing by relying instead on a set of scattered collocation points at which the governing partial differential equations (PDEs) are evaluated, these models lack interpretability, can be incredibly costly to train, and can suffer from convergence issues. This is because they effectively replace a set of linear algebraic solves arising from a traditional (e.g., finite element or finite difference) discretization with a complex nonlinear optimization problem involving the neural network's (NN's) parameters.

This paper develops a novel hybrid domain decomposition- (DD-)based approach that has the potential to mitigate both the meshing and long runtime requirement issues described above by "gluing together," in a plug-and-play fashion, arbitrary combinations of subdomain-local high-fidelity full order models (FOMs) with subdomain-local ROMs using the Schwarz alternating method (SAM) [49]. The key idea behind SAM is to first decompose the physical domain on which a given PDE is posed into smaller subdomains,

---

*Email: ikalash@sandia.gov

and then to solve a sequence of problems on each of the subdomains while exchanging boundary condition information across subdomain interfaces to ensure solution compatibility. The approach presented herein builds on our past work in developing SAM as a means to achieve concurrent multiscale coupling of high-fidelity finite element models in solid mechanics [37, 38], and of intrusive projection-based ROMs in hyperbolic fluid problems discretized using the cell-centered finite volume method [55]. It was demonstrated in these manuscripts that SAM has a number of advantages: (i) it is minimally intrusive to implement in existing HPC software frameworks; (ii) it is capable of coupling regions with different mesh resolutions, different element types, different time integration schemes (e.g., implicit and explicit), and even different models (e.g., FOM and ROM), all without introducing any artifacts exhibited by alternative coupling methods; and (iii) it possesses rigorous convergence properties/guarantees [37, 38]. Additionally, we argue in [55] that DD-based FOM-ROM and ROM-ROM couplings have the potential of improving the predictive viability of ROMs by enabling the spatial localization of ROMs (via domain decomposition), and through the online integration of high-fidelity information into these models (via FOM coupling).

The aim of the present work is to extend SAM further to enable the DD-based coupling of non-intrusive ROMs constructed via an approach known as operator inference, or OpInf, pioneered by Peherstorfer and Willcox [44]. In developing ROM-ROM and FOM-ROM couplings, our goal is to reduce the implementational burden in combining SAM with model order reduction (MOR), making it accessible by a wider range of applications and codes. Traditional intrusive MOR requires access to the underlying FOM code in order to project the governing PDE(s) onto a reduced subspace, posing a high development cost. OpInf instead assumes a functional form (usually linear or quadratic [18]) for the ROM in terms of reduced operators, which are learned offline from data. This procedure significantly reduces both the development time and the time-to-impact.

We focus our attention herein on the simplest version of SAM, namely overlapping SAM (O-SAM), in which the physical geometry is decomposed into overlapping subdomains and the coupling is achieved via Dirichlet transmission boundary conditions on the subdomain boundaries; further extensions to non-overlapping domain decompositions has been explored in the preliminary work [47] and will be the subject of a future publication. Since we are targeting an application space in which system geometries can be too complex to mesh using a monolithic meshing scheme, both *a priori* ROM training and *a posteriori* accuracy and runtime assessments are performed with respect to a SAM-based coupled simulation involving finite element FOMs. The assumption that a FOM-FOM coupled solution on the full physical domain of interest is possible to obtain via SAM enables us to utilize a "top-down" (vs. a "bottom-up" [8]) training approach, which does not require ROM subdomains to be simulated independently, and generally gives rise to more expressive reduced bases and more accurate ROMs. While we develop our method around a solid dynamics exemplar, our SAM-based coupling strategy described herein can be applied to any set of PDEs. We also note that the proposed approach can be applied in the case "bottom-up" training is utilized to build the ROMs being coupled.

We describe the specific contributions and differentiating features of our approach in Section 1.2, after surveying the literature for related past work in Section 1.1.

1.1 Overview of related past work

The method proposed herein is an addition to a growing body of literature on DD-based couplings involving a variety of ROM formulations. For the sake of brevity, we will focus our literature review on Schwarz and Schwarz-like coupling methods for FOM-ROM and ROM-ROM couplings involving non-intrusive ROMs, as this class of methods is most closely related to the approach developed in this paper; for a more detailed overview that includes a comprehensive survey of a variety of coupling methods involving intrusive projection-based ROMs, the interested reader is referred to [55].

Some of the earliest Schwarz-based DD approaches for ROM-ROM and ROM-FOM coupling are based on non-intrusive ROMs constructed via Galerkin-free Proper Orthogonal Decomposition (POD). Galerkin-free POD ROMs are a class of reduced order modeling techniques that aim to efficiently approximate the behavior of complex dynamical systems without relying on the traditional Galerkin projection method. Instead of projecting the governing equations onto a reduced POD basis, as done in traditional intrusive projection-based MOR, Galerkin-free POD ROMs predict the solution by applying interpolation techniques to a POD representation of the primary solution field. Three methods to perform Galerkin-free ROM-FOM and ROM-ROM coupling are presented by Buffoni *et al.* [6]: (i) a Schur iteration where the solution of the PROM is obtained by a projection step in the space spanned by the POD modes, (ii) a Dirichlet–Dirichlet iteration in the frame of a classical Schwarz method, and (iii) an approach obtained by minimizing the residual norm of the canonical approximation in the space

spanned by the POD modes. Another Galerkin-free ROM-FOM and ROM-ROM coupling approach that makes use of SAM is the work of Cinquegrana *et al.* [9], but, unlike our approach, matching meshes at the subdomain interfaces are required. A third Galerkin-free POD approach, termed zonal Galerkin-free POD [4], defines an optimization problem that minimizes the difference between the POD reconstruction and its corresponding FOM solution in the overlapping region between a ROM and a FOM domain.

Schwarz-based methods have also been explored as a means to couple subdomain-local ROMs obtained non-intrusively via the proper generalized decomposition (PGD). In the recent pre-print [14], Discacciati *et al.* develop an algebraic SAM for coupling together subdomain-local non-intrusive PGD ROMs following a domain decomposition of the underlying spatial domain into overlapping subdomains. The proposed method is an improvement of the method originally presented in reference [13]. The coupling strategies proposed in [13, 14] are currently limited to linear elliptic PDEs and ROM-ROM (vs. ROM-FOM) couplings.

Recent years have seen the development of coupling methods for various types of NN-based ROMs. In [53], Wang *et al.* introduce a genomic flow network (GFNet), in which pre-trained PINNs or NNs are trained locally on small "genomes" (subdomains) and stitched together via a Schwarz-like iteration to represent the solution on an arbitrary domain comprised of the subdomain-local genomes. While the approach is presented as a non-overlapping method, convergence requires the introduction of "auxiliary" genomes, which effectively introduce overlap into the domain decomposition. There are several other methods that utilize Schwarz for PINN coupling, namely [30, 31, 51]; however, the primary purpose of these approaches is to accelerate the training stage of the PINN construction process, rather than to glue together pre-trained subdomain-local models, as we attempt to do herein. The work by Discaciatti and Hesthaven [15] similarly proposes a Schwarz-like method leveraging NNs. In this paper, the authors assume that local models assigned to non-overlapping subdomains are glued together using the relaxed Dirichlet-Neumann SAM, and construct a surrogate for the Dirichlet and Neumann maps defining the coupling conditions using ROMs based on kernel interpolation methods and NNs. One disadvantage of the method, as presented in [15], is that it does not automatically provide the local solution in specific subdomains, as its main purpose is boundary map representation for Schwarz transmission conditions. In [54], Wang *et al.* develop a hybrid framework that integrates PI-DeepONets [32] with finite element method (FEM) models via domain decomposition and O-SAM, focusing on a two-dimensional (2D) solid dynamics exemplar. The basic domain decomposition strategy is to assign PI-DeepONets to computationally demanding regions, while resolving the remainder of the computational domain via the classical FEM. Extensive numerical results are presented demonstrating that error margins below 1% are achievable; however, the subdomain-local PI-DeepONets are extremely challenging to train, requiring up to $O(10^6)$ epochs to converge.

The two most closely related works to the present manuscript are [16] and [19], both of which propose Schwarz-like couplings involving non-intrusive OpInf ROMs [44]. In [16], Farcas *et al.* develop an overlapping DD-based methodology for coupling subdomain-local OpInf ROMs by learning appropriate reduced operators responsible for the coupling, and demonstrates the method on a challenging three-dimensional (3D) combustion example. While this coupling framework is not presented in the context of SAM, it is effectively equivalent to applying a single iteration of the parallel (additive) [17, 55] version of O-SAM, rather than iterating to convergence as done within the classical Schwarz framework considered herein. Another recent related work is that of Gkimisis *et al.* [19], which presents a fully non-intrusive hybrid ROM-FOM coupling framework, in which OpInf ROMs are coupled to "sparse FOMs" (sFOMs) via an overlapping Schwarz iteration procedure. The sFOMs are inferred offline from training data collected from an existing high-fidelity simulation code by learning the numerical stencil of an adjacency-based FOM corresponding to the high-fidelity system dynamics. The primary motivation for doing this is to enable one to create ROM-FOM couplings without requiring access to the FOM code.

Finally, the recent pre-print from our group [47] presents the first application, to the authors' knowledge, of the non-overlapping SAM for the coupling of subdomain-local non-intrusive OpInf ROMs with each other and with subdomain-local FOMs. Both Robin-Robin and alternating Dirichlet-Neumann transmission conditions are explored within this coupling framework, with the former yielding improved convergence and accuracy relative to the latter.

1.2 Contributions, differentiating features and organization

While the methodology described herein shares some similarities with the work of Farcas *et al.* [16] and Gkimisis *et al.* [19], both of which propose overlapping Schwarz-like formulations for the DD-based coupling subdomain-local OpInf models, there are also some important distinctions. Whereas [16] and [19]

limit their attention to linear and quadratic OpInf ROMs, we also consider a higher order cubic OpInf ROM, or "COpInf." Moreover, neither [16] nor [19] describe and demonstrate a use case in which different time-integrators with different time steps are coupled together using SAM, and the former work does not consider FOM-OpInf couplings. As explained in Section 1.1, the method developed of Farcas *et al.* [16] can be viewed as an additive Schwarz-like coupling framework in which each set of subdomain problems is solved once, rather than by performing an iteration to convergence. We demonstrate in Section 5.4 that such an approach is inadequate for certain classes of nonlinear problems. Whereas the hybrid FOM-OpInf couplings developed herein require access to the FOM code in a minimally-intrusive way, the work of Gkimisis *et al.* [19] takes the approach of using overlapping SAM to couple subdomain-local OpInf ROMs with sFOMs, sparse FOMs learned offline from available simulation data, as described in more detail in Section 1.1. One advantage of this method is that it yields a coupling framework that is fully non-intrusive even when performing OpInf-FOM coupling. However, learning the sFOM requires the numerical solution of a complex optimization problem, which can be infeasible when the sFOM is posed on a complex, unstructured 3D geometry. This issue is compounded for multi-query scenarios in which the geometry, mesh and/or material parameters may change between simulations, requiring re-learning of the sFOM. Additionally, the learned sFOM in [19] can introduce an additional source of error into the approximation.

It is worth noting that the use case we target with our SAM-based hybrid approach is similar to that of [19] but different from that of [54]: in a complex multiscale simulation, our strategy is to assign the subdomain containing the more complex, nonlinear dynamics to a FOM, as done in [19], so as to maximize the accuracy of the coupled model. In contrast, [54] (and some other works, e.g., [42]) take the approach of assigning more refined subdomains to a (NN-bassed) ROM, towards maximizing improvements in online CPU time. Further, the OpInf ROMs proposed herein are substantially easier to train than the PI-DeepONets considered in [54].

It is also worth mentioning that the domain decompositions employed herein and in practical problems of interest to us are always physically-motivated; in particular, we are not targeting the use case where a complex domain is "tiled" with hundreds of pre-trained subdomains, as done in [53] and [8, 7]. It is thus common in our applications to have 2–3 subdomain, and we do not envision in general having to couple more than 5–10 subdomains.

While the present work is an extension of some of our earlier research on SAM for FOM-FOM coupling [37, 38] and contact [40] in solid mechanics, intrusive ROM-ROM/ROM-FOM coupling [55, 3], and non-intrusive OpInf-OpInf/OpInf-FOM coupling [35, 34, 47], it contains a number of new contributions. This paper is not only one of the earliest to investigate SAM-based coupling of subdomain-local OpInf ROMs, it is also the first to apply this approach to realistic 3D nonlinear problems in solid mechanics. We feature problems posed on geometries discretized with varying mesh resolutions and element types, and explore the integration of disparate time integration schemes that may have different time-steps into the coupling scheme. We demonstrate that it is possible to obtain accurate and efficient hybrid models for *fully nonlinear* solid dynamics problems by coupling together subdomain-local OpInf ROMs based on *polynomial approximations* of the governing PDEs. Additionally, we propose several novel strategies for improving the efficiency and robustness of O-SAM, including boundary POD bases to reduce the size of boundary operators and methods for optimizing regularization parameters in subdomain-local OpInf ROMs within the Schwarz coupling framework.

The remainder of this paper is organized as follows. In Section 2, we describe our model solid dynamics problem and its monolithic finite element discretization. Section 3 overviews the OpInf approach to building non-intrusive ROMs, assuming a generic cubic functional form for the learned model. In Section 4, we describe our overlapping Schwarz algorithm and how it can be used to create hybrid FOM-OpInf and OpInf-OpInf models for generic solid dynamics applications. Section 5 evaluates the proposed approach's accuracy and efficiency on four solid mechanics test cases, in which subdomains are discretized using disparate meshes/element types and/or advanced forward in time using different time-integrators with disparate time steps. Conclusions are offered in Section 6.

2 Solid mechanics problem formulation and monolithic finite element discretization

Consider the Euler-Lagrange equations for a general solid mechanics problem:

$$\nabla \cdot \boldsymbol{P} + \rho \boldsymbol{R} = \rho \ddot{\boldsymbol{\varphi}} \quad \text{in } I \times \Omega. \tag{1}$$

Here, $\Omega \in \mathbb{R}^d$ for $d \in \{1, 2, 3\}$ is an open and bounded domain, $I := \{t \in [t_0, T]\}$ is a closed interval with $0 \le t_0 < T$, and $\boldsymbol{x} = \boldsymbol{\varphi}(t, \boldsymbol{X}) : I \times \Omega \to \mathbb{R}^d$ describes the mapping from the reference configuration $\boldsymbol{X} \in \Omega$ to the current configuration. The "double-dot" notation $\ddot{\boldsymbol{\varphi}}$ indicates a second derivative in time. The symbol $\nabla \cdot$ represents the row-wise divergence, $\boldsymbol{P} := \boldsymbol{P}(\boldsymbol{\varphi})$ denotes the first Piola-Kirchoff stress tensor, $\boldsymbol{R}$ is the specific body force, and ρ is the mass density defined in the reference configuration. We note that $\boldsymbol{P}$ is described by a constitutive model which is generally nonlinear in the current state $\boldsymbol{\varphi}$. Specifically, $\boldsymbol{P}$ is defined as the derivative of the Helmholtz free-energy density, denoted by $A(\boldsymbol{F})$, with respect to the deformation gradient $\boldsymbol{F}$, that is, $\boldsymbol{P} := \frac{\partial A}{\partial \boldsymbol{F}}$. In the present work, we restrict attention to hyperelastic material models, in which A is not path-dependent (i.e., it does not depend on a collection of internal variables). For a concrete example of $A(\boldsymbol{F})$ and $\boldsymbol{P}$, the reader is referred to Appendix A, where expressions for these variables are given for the linear elastic, Saint Venant–Kirchhoff [22] and Neohookean [39] material models.

To ensure well-posedness, the problem (1) is subject to initial and boundary conditions (BCs),

$$
\begin{aligned}
&\boldsymbol{\varphi}(t_0, \boldsymbol{X}) = \boldsymbol{x}_0 \quad \text{in } \Omega, && \dot{\boldsymbol{\varphi}}(t_0, \boldsymbol{X}) = \boldsymbol{v}_0 \quad \text{in } \Omega, \\
&\boldsymbol{\varphi}(t, \boldsymbol{X}) = \boldsymbol{\chi} \quad \text{on } I \times \partial_\varphi \Omega, && \boldsymbol{P} \boldsymbol{n} = \boldsymbol{T} \quad \text{on } I \times \partial_T \Omega,
\end{aligned}
\tag{2}
$$

where the boundary $\partial \Omega = \partial_\varphi \Omega \cup \partial_T \Omega$ is the union of Dirichlet $\partial_\varphi \Omega$ and Neumann $\partial_T \Omega$ parts, which satisfy the non-overlapping condition $\partial_\varphi \Omega \cap \partial_T \Omega = \emptyset$, and $\boldsymbol{n}$ denotes the unit vector normal to $\partial_T \Omega$.

In most practical applications, we wish to discretize the PDEs above in space using the classical Galerkin FEM. Toward this effect, consider the function space $Q = C^2\big(I; W_1^2(\Omega) \cap C^0(\partial \Omega)\big)$. Let $\mathcal{V} := \{\boldsymbol{\Theta} \in Q : \boldsymbol{\Theta} = \boldsymbol{\chi} \text{ on } I \otimes \partial_\varphi \Omega\}$ and $\mathcal{V}_0 := \{\boldsymbol{\zeta} \in Q : \boldsymbol{\zeta} = \boldsymbol{0} \text{ on } (\partial_\varphi \Omega \times I) \cup (\Omega \times \partial I)\}$ denote the test and trial function spaces, respectively, with $\boldsymbol{\varphi} \in \mathcal{V}$ and $\boldsymbol{\xi} \in \mathcal{V}_0$. It is now straightforward to derive the weak variational form of (1)–(2):

$$
\int_\Omega \left(\rho \ddot{\boldsymbol{\varphi}} \cdot \boldsymbol{\xi} + \boldsymbol{P} : \nabla \boldsymbol{\xi} - \rho \boldsymbol{R} \cdot \boldsymbol{\xi} \right) d\Omega - \int_{\partial_T \Omega} \boldsymbol{T} \cdot \boldsymbol{\xi} \, dS = 0.
\tag{3}
$$

Semi-discretization of (3) in space using standard isoparametric finite elements yields a system of the form

$$
\boldsymbol{M} \ddot{\boldsymbol{\varphi}}_h + \boldsymbol{K}(\boldsymbol{\varphi}_h) = \boldsymbol{f}(\boldsymbol{T}).
\tag{4}
$$

Here, $\boldsymbol{\varphi}_h \in \mathbb{R}^M$ is the discretized solution vector, the mass matrix $\boldsymbol{M} \in \mathbb{R}^{M \times M}$ represents the usual discretization of the ρ-weighted L^2 inner product $(\cdot, \cdot)$, the nonlinear function $\boldsymbol{K}(\boldsymbol{\varphi}_h) \in \mathbb{R}^M$ discretizes the weak-form divergence of $\boldsymbol{P}$, and $\boldsymbol{f} \in \mathbb{R}^N$ captures the contribution from the external force term $\rho \boldsymbol{R}$ along with the traction BC $\boldsymbol{T}$. The non-negative integer $M \in \mathbb{N}$ denotes the number of degrees of freedom (DoFs) of the semi-discrete system (4).

To complete the discrete problem formulation, (4) must be augmented with appropriate Dirichlet BCs. The standard way to impose Dirichlet BCs in the FEM is to split the solution vector $\boldsymbol{u}$ into constrained and unconstrained DoFs, modifying (4) such that only the unconstrained DoFs are solved for. Toward this effect, let us partition the solution vector $\boldsymbol{\varphi}_h$ as $\boldsymbol{\varphi}_h := \big(\boldsymbol{u}^{\mathsf{T}} \quad \boldsymbol{u}_c^{\mathsf{T}} \big)^{\mathsf{T}} := \big(\boldsymbol{u}^{\mathsf{T}} \quad \boldsymbol{\chi}_h^{\mathsf{T}} \big)^{\mathsf{T}}$, where $\boldsymbol{u} \in \mathbb{R}^N$ is the vector of unconstrained DoFs to be solved for, and $\boldsymbol{u}_c = \boldsymbol{\chi}_h \in \mathbb{R}^K$ are discrete values of the Dirichlet data $\boldsymbol{\chi}$, where $N, K \in \mathbb{N}$ and $N + K = M$. Substituting this decomposition into (4), a system of the form

$$
\tilde{\boldsymbol{M}} \ddot{\boldsymbol{u}} + \tilde{\boldsymbol{K}}(\boldsymbol{u}, \boldsymbol{\chi}_h) = \tilde{\boldsymbol{f}}(\boldsymbol{T}, \boldsymbol{\chi}_h, \ddot{\boldsymbol{\chi}}_h)
\tag{5}
$$

is obtained for the unconstrained DoFs $\boldsymbol{u}$, where $\tilde{\boldsymbol{M}} \in \mathbb{R}^{N \times N}$ and $\tilde{\boldsymbol{K}}, \tilde{\boldsymbol{f}} \in \mathbb{R}^N$.

To create a fully-discrete model, (5) must be discretized in time. Herein, the semi-discrete system is time-advanced using the standard Newmark-β time-integration scheme [36], a popular choice in solid mechanics due to its flexibility, convergence and ability to preserve the symplectic structure of the underlying PDEs. More details about this scheme can be found in Section 4.

The functional forms of $\tilde{\boldsymbol{K}}(\boldsymbol{u}, \boldsymbol{\chi}_h)$ and $\tilde{\boldsymbol{f}}(\boldsymbol{T}, \boldsymbol{\chi}_h, \ddot{\boldsymbol{\chi}}_h)$ in (5) depend on the nonlinearities present in the vector of internal forces, $\boldsymbol{K}(\cdot)$. Consider the example of a simple linear elastic material (Appendix A.1), for which $\boldsymbol{K}(\boldsymbol{\varphi}_h) := \boldsymbol{K} \boldsymbol{\varphi}_h$. The reader can observe that the semi-discrete system (5) can be written as

$$
\tilde{\boldsymbol{M}} \ddot{\boldsymbol{u}} + \tilde{\boldsymbol{K}} \boldsymbol{u} = \boldsymbol{B} \boldsymbol{g},
\tag{6}
$$

where $\tilde{\boldsymbol{M}} := \boldsymbol{M}_{[1:N, 1:N]}$, $\tilde{\boldsymbol{K}} := \boldsymbol{K}_{[1:N, 1:N]}$, and

$$
\boldsymbol{B} := \big(\boldsymbol{I}_u, \quad -\boldsymbol{M}_{uc}, \quad -\boldsymbol{K}_{uc} \big) \in \mathbb{R}^{N \times (N + 2K)},
\tag{7}
$$

$$g := \left(\begin{array}{ccc} \boldsymbol{f}_u^{\mathsf{T}}, & \ddot{\boldsymbol{\chi}}_h^{\mathsf{T}}, & \boldsymbol{\chi}_h^{\mathsf{T}} \end{array} \right)^{\mathsf{T}} \in \mathbb{R}^{N+2K}, \tag{8}$$

with $\boldsymbol{M}_{uc} := \boldsymbol{M}_{[1:N,N+1:N]}$, $\boldsymbol{K}_{uc} := \boldsymbol{K}_{[1:N,N+1:M]}$, $\boldsymbol{f}_u := \boldsymbol{f}_{[1:N]}$, $\boldsymbol{I}_u$ denoting the $N \times N$ identity matrix. Here, the notation $\boldsymbol{M}_{[i,j]}$ denotes the $(i,j)^{th}$ component of the matrix $\boldsymbol{M}$ (and similarly for $\boldsymbol{K}$), and $\boldsymbol{f}_{[i]}$ denotes the ith component of the vector $\boldsymbol{f}$.

In the present work, we focus our attention on developing ROMs for approximations of (4) which have a polynomial (at most, cubic) structure, i.e., systems of the form

$$\boldsymbol{M}\ddot{\boldsymbol{\varphi}}_h + \boldsymbol{K}\boldsymbol{\varphi}_h + \boldsymbol{H}\boldsymbol{\varphi}_h^{\otimes 2} + \boldsymbol{C}\boldsymbol{\varphi}_h^{\otimes 3} = \boldsymbol{f}(\boldsymbol{T}). \tag{9}$$

Here, $\boldsymbol{\varphi}_h^{\otimes k}$ is a Kronecker product of $\boldsymbol{\varphi}_h$ with itself k times, and the matrices $\boldsymbol{M}, \boldsymbol{K} \in \mathbb{R}^{M \times M}$ $\boldsymbol{H} \in \mathbb{R}^{M \times M^2}$, and $\boldsymbol{C} \in \mathbb{R}^{M \times M^3}$ arise via the discretization of differential operators in the governing equations[1], or by performing a Taylor expansion of (4). For example, in the case of linear elasticity, $\boldsymbol{M}, \boldsymbol{K}$ represent the usual mass and stiffness matrices discretizing the L^2 inner product $(\cdot, \cdot)$ and weak-form Laplace operator $(\nabla \cdot, \nabla \cdot)$. As shown in Appendix A,2, the PDEs (1)–(2) reduce to a cubic functional form (9) when a Saint Venant–Kichhoff material model [22] is specified (see Appendix A.2). While, for a general hyperelastic material, the governing PDEs will not reduce to (9), we show in Section 5 that the system (9) can be effective as a surrogate model when combined with Schwarz coupling even when the governing equations under consideration are fully nonlinear.

As for the linear case, to apply Dirichlet BCs within (9), we must substitute the decomposition $\boldsymbol{\varphi}_h := \left(\begin{array}{cc} \boldsymbol{u}, & \boldsymbol{\chi}_h \end{array} \right)^{\mathsf{T}}$ into (9) and bring the terms corresponding to the Dirichlet boundary condition DoFs to the right-hand-side. Doing so gives rise to a system of the form

$$\tilde{\boldsymbol{M}}\ddot{\boldsymbol{u}} + \tilde{\boldsymbol{K}}(\boldsymbol{\chi}_h)\boldsymbol{u} + \tilde{\boldsymbol{H}}(\boldsymbol{\chi}_h)\boldsymbol{u}^{\otimes 2} + \tilde{\boldsymbol{C}}\boldsymbol{u}^{\otimes 3} = \boldsymbol{f}_u - \boldsymbol{M}_{uc}\ddot{\boldsymbol{\chi}}_h - \boldsymbol{K}_{uc}\boldsymbol{\chi}_h - \boldsymbol{h}(\boldsymbol{H}, \boldsymbol{C}, \boldsymbol{\chi}_h), \tag{10}$$

for matrix $\tilde{\boldsymbol{K}}(\boldsymbol{\chi}_h) \in \mathbb{R}^{N \times N}$, vector $\boldsymbol{h} \in \mathbb{R}^N$, and appropriately-sized tensors $\tilde{\boldsymbol{H}}(\boldsymbol{\chi}_h)$ and $\tilde{\boldsymbol{C}}$. The linear and quadratic operators are functions of the boundary data $\boldsymbol{\chi}_h$, as they include contributions from cross terms of the form $\boldsymbol{u} \otimes \boldsymbol{\chi}_h$ and $\boldsymbol{u} \otimes \boldsymbol{\chi}_h \otimes \boldsymbol{\chi}_h$ that arise when substituting the decomposed $\boldsymbol{\varphi}_h$ vector into (9).

In defining the functional form of our OpInf ROMs in Sections 3 and 4, we will approximate the right-hand-side of (10) in several ways. First, since none of the numerical examples considered herein have a body source term or inhomogeneous traction boundary conditions, we will assume from this point forward that $\boldsymbol{f}_u = \boldsymbol{0}$ in (10). We will additionally neglect the $\ddot{\boldsymbol{\chi}}_h$ term in (8); this term drops out if the mass matrix is lumped[2] and is expected to be small in the general case. Finally, we will make the modeling decision to neglect the $\boldsymbol{h}(\boldsymbol{H}, \boldsymbol{C}, \boldsymbol{\chi}_h)$ term in (10) as a first-order approximation. With these assumptions, our approximate semi-discrete FOM model on which the algebraic structure of our OpInf ROMs is based (see Sections 3 and 4) takes the form:

$$\tilde{\boldsymbol{M}}\ddot{\boldsymbol{u}} + \tilde{\boldsymbol{K}}(\boldsymbol{\chi}_h)\boldsymbol{u} + \tilde{\boldsymbol{H}}(\boldsymbol{\chi}_h)\boldsymbol{u}^{\otimes 2} + \tilde{\boldsymbol{C}}\boldsymbol{u}^{\otimes 3} = \tilde{\boldsymbol{B}}\tilde{\boldsymbol{g}}, \tag{11}$$

where

$$\tilde{\boldsymbol{B}} := -\boldsymbol{K}_{uc} \in \mathbb{R}^{N \times K}, \tag{12}$$

and

$$\tilde{\boldsymbol{g}} := \boldsymbol{\chi}_h \in \mathbb{R}^K. \tag{13}$$

As we show in Section 5, subdomain-local OpInf ROMs based on (11), when coupled to other subdomain-local OpInf ROMs and/or subdomain-local FOMs, can yield tremendously accurate approximations despite the fact that a number of assumptions and simplifications have been made to go from (4) to (11).

Remark 1. *The polynomial approximation (11) to (20) is introduced only for the purpose of motivating the cubic OpInf ROMs described in Section 3 and considered herein. As will be made clear in Section 4.2.2, our FOM-OpInf couplings create hybrid models in which a fully nonlinear subdomain-local FOM based on (20) is coupled to a polynomial subdomain-local ROM based on (11).*

---

[1]Since $\boldsymbol{\varphi}_h^{\otimes k}$ is symmetric, it is computed in practice using a compressed vector representation of length $(1/k!)M(M + 1)...(M + k - 1)$. Similarly, the symmetric operators $\boldsymbol{H} \in \mathbb{R}^{M \times M^2}$ and $\boldsymbol{C} \in \mathbb{R}^{M \times M^3}$ are parameterized in terms of their independent degrees of freedom.

[2]Many production codes, such as Sandia's `SIERRA/SM` code [28], utilize a lumped matrix for both implicit and explicit solves.

3 Model reduction via non-intrusive operator inference (OpInf)

Many effective model reduction strategies are intrusive to the underlying FOM discretization. Galerkin ROMs built with a data-driven technique such as POD [21, 23, 50], for example, require the ability to project the discrete operators governing the FOM onto the span of the POD basis, which requires the user to directly modify the FOM implementation. Conversely, many FOMs corresponding to practical applications are proprietary, built on legacy software, or require specialized expertise, in which case the underlying FOM must be treated as inaccessible. This has motivated the development of non-intrusive model reduction strategies such as OpInf [44], which will now be described.

OpInf is a projection-inspired method for model reduction, originally introduced by Peherstorfer and Willcox [44] as a non-intrusive alternative to standard Galerkin ROMs. The algorithm proceeds in two major stages. First, a basis for the solution is built from FOM snapshot data similarly to the intrusive Galerkin case. However, instead of projecting the FOM onto the span of this basis, the second stage infers a data-driven approximation to the necessary projected operators from (projections of) the same FOM snapshot data, allowing for a fully non-intrusive ROM. The remainder of this section describes the particular OpInf strategy that is employed in the present work.

3.1 Stage 1: proper orthogonal decomposition

The first stage of OpInf constructs a reduced basis, commonly with the standard POD algorithm [21, 23, 50]. To illustrate this procedure, suppose that the monolithic problem (9) has been solved in time for $\tau + 1$ separate states $0 = t_0 < t_1 < \cdots < T$, yielding a collection of snapshots of the unconstrained state,

$$\boldsymbol{U} = [\boldsymbol{u}(t_0), \boldsymbol{u}(t_1), \ldots, \boldsymbol{u}(T)] \in \mathbb{R}^{N \times (\tau+1)}. \tag{14}$$

The governing assumption in POD is that the solution space to the FOM (9) has low rank relative to the DoF space $\mathbb{R}^N$. Correspondingly, the snapshot matrix $\boldsymbol{U}$ possesses a low-rank structure, i.e., $\text{rank}(\boldsymbol{U}) \ll N$, which can be exploited for model reduction. Considering the (thin) singular value decomposition (SVD) $\boldsymbol{U} = \boldsymbol{\Phi \Sigma \Psi}^\intercal$, the column-orthonormal matrix $\boldsymbol{\Phi} \in \mathbb{V}_{\text{rank}(\boldsymbol{U})} \left(\mathbb{R}^N\right)$ then spans the range of the discrete dynamics exhibited in $\boldsymbol{U}$; here $\mathbb{V}_K \left(\mathbb{R}^N\right)$ denotes the space of $N \times K$ orthonormal matrices (i.e., $\mathbb{V}_K \left(\mathbb{R}^N\right) := \{\mathbf{X} \in \mathbb{R}^{N \times K} | \mathbf{X}^T \mathbf{X} = \mathbf{I}\}$). Moreover, the first $r \leq \text{rank}(\boldsymbol{U})$ columns of $\boldsymbol{\Phi}$ form a variance-maximizing basis, denoted $\boldsymbol{\Phi}_r$ which is ℓ^2-optimal in the sense that

$$\min_{\boldsymbol{V} \in \mathbb{R}^{N \times r}} \|\boldsymbol{U} - \boldsymbol{V V}^\intercal \boldsymbol{U}\|_F^2 = \|\boldsymbol{U} - \boldsymbol{\Phi}_r \boldsymbol{\Phi}_r^\intercal \boldsymbol{U}\|_F^2 = \sum_{i=r+1}^{\text{rank}(\boldsymbol{U})} \sigma_i^2, \tag{15}$$

where σ_i denotes the ith diagonal entry in $\boldsymbol{\Sigma}$. The basis $\boldsymbol{\Phi}_r$ is employed in all present cases and will be referred to as the POD basis of rank r.

Remark 2. *The discussion above suggests that only the primary solution field, in this case, the displacement field $\boldsymbol{u}$, is used to build the POD basis $\boldsymbol{\Phi}_r$. While this is the approach taken herein, we note that it is possible to augment the snapshot set $\boldsymbol{U}$ in (14) with other data, e.g., snapshots of the velocity and/or acceleration fields, as discussed in [1].*

Remark 3. *While we limit the discussion herein to affine POD bases, we point out that the ROM latent space can also be approximated by nonlinear bases, e.g., quadratic bases [1, 18], convolutional autoencoders [29], or other Artificial Neural Networks (ANNs) [2]. In particular, our SAM-based coupling framework can be used to couple OpInf ROMs with latent spaces spanned by either linear or nonlinear bases.*

3.2 Stage 2: least-squares regression

Following the calculation of the POD basis $\boldsymbol{\Phi}_r$, it is necessary to construct a non-intrusive approximation to the usual projection-based Galerkin ROM, which forms the core of the OpInf routine. Since we know that the algebraic structure of a polynomial FOM is preserved under projection, the POD ROM corresponding to (11) is also cubic and takes the form[3]

$$\hat{\boldsymbol{M}} \ddot{\hat{\boldsymbol{u}}} + \hat{\boldsymbol{K}} \hat{\boldsymbol{u}} + \hat{\boldsymbol{H}} \hat{\boldsymbol{u}}^{\otimes 2} + \hat{\boldsymbol{C}} \hat{\boldsymbol{u}}^{\otimes 3} = \hat{\boldsymbol{B}} \tilde{\boldsymbol{g}}, \tag{16}$$

---

[3]Postulation of (16) assumes that the Dirichlet degrees of freedom have been explicitly removed, unlike the FOM analogues of these matrices in (11).

where $\hat{M} = \boldsymbol{\Phi}_r^\intercal \boldsymbol{M} \boldsymbol{\Phi}_r$, $\hat{K} = \boldsymbol{\Phi}_r^\intercal \boldsymbol{K} \boldsymbol{\Phi}_r$, $\hat{H} = \boldsymbol{\Phi}_r^\intercal \boldsymbol{H} (\boldsymbol{\Phi}_r \otimes \boldsymbol{\Phi}_r)$, $\hat{C} = \boldsymbol{\Phi}_r^\intercal \boldsymbol{C} (\boldsymbol{\Phi}_r \otimes \boldsymbol{\Phi}_r \otimes \boldsymbol{\Phi}_r)$, and $\hat{B} = \boldsymbol{\Phi}_r^\intercal \boldsymbol{B}$ are projections of the FOM operators onto the span of the reduced basis $\boldsymbol{\Phi}_r$. The goal of OpInf is to infer approximations of these objects from the collected snapshot data $\boldsymbol{U}$. Assuming an $\boldsymbol{M}$-orthonormal basis $\boldsymbol{\Phi}_r$, or, non-equivalently, left multiplying by $\hat{M}^{-1}$, the alternative model form referred to as the monolithic OpInf formulation can be written as

$$\ddot{\hat{u}} + \bar{K}\hat{u} + \bar{H}\hat{u}^{\otimes 2} + \bar{C}\hat{u}^{\otimes 3} = \bar{B}\tilde{g}. \tag{17}$$

Here, the low-dimensional operators $\bar{K}, \bar{H}, \bar{B}$ are data-driven surrogates for their intrusive counterparts in (16), which are assumed inaccessible[4]. These operators can be computed as the solution to a convex learning problem with a least-squares objective. More precisely, given projected state data $\bar{U} = \boldsymbol{\Phi}_r^\intercal \boldsymbol{U}$ and projected velocity information $D_t^2(\bar{U}) \approx \boldsymbol{\Phi}_r^\intercal \ddot{\boldsymbol{U}}$ computed with, e.g., a finite difference operator D_t, the OpInf learning problem is given as a minimization of the approximate residual:

$$\underset{\bar{K},\bar{H},\bar{C},\bar{B}}{\arg\min} \ \left\| D_t^2(\bar{U}) + \bar{K}\bar{U} + \bar{H}\bar{U}^{*2} + \bar{C}\bar{U}^{*3} - \bar{B}G \right\|_F^2 + \lambda\left(\|\bar{K}\|_F^2 + \|\bar{H}\|_F^2 + \|\bar{C}\|_F^2 + \|\bar{B}\|_F^2 \right) \tag{18}$$

where $\bar{U}^{*k} \in \mathbb{R}^{r^k \times \tau}$ is the column-wise Kronecker (i.e., Khatri–Rao) power of $\boldsymbol{U}$, G is a matrix whose i^{th} column is $\tilde{g}(t_i)$, and $\lambda \in \mathbb{R}^+$ is a (scalar-valued) regularization parameter. The most basic, linear form of OpInf neglects the $\bar{H}$ and $\bar{C}$ terms, while quadratic OpInf (or QOpInf) only neglects the $\bar{C}$ term. Cubic OpInf (COpInf) retain the all terms in (18).

Remark 4. *We note that various OpInf works, e.g., [33, 45], examine vector-valued regularization parameters such that each operator has its own regularization parameter. We do not consider this here as the resulting grid search for the optimal parameter combination scales exponentially with the number of parameters to infer.*

The advantage of this procedure is that the simulation of the approximate ROM (17) is non-intrusive, since the surrogate operators $\bar{K}, \bar{H}, \bar{C}, \bar{B}$ are learned directly from data. Moreover, under the assumptions that (i) the time-integration of the FOM (9) is convergent with decreasing step-size $\Delta t \to 0$, (ii) the data $\bar{U}$ (and its Khatri–Rao powers) have full column rank, and (iii) the time derivative approximations $D_t^2(\boldsymbol{U})$ converge to their instantaneous equivalents with decreasing difference parameter, it follows that for $\lambda = 0$ and any $\varepsilon > 0$, the learned approximations satisfy (c.f. [44])

$$\|\bar{K} - \hat{K}\|_F < \varepsilon, \qquad \|\bar{H} - \hat{H}\|_F < \varepsilon, \qquad \|\bar{B} - \hat{B}\|_F < \varepsilon, \qquad \|\bar{C} - \hat{C}\|_F < \varepsilon, \tag{19}$$

for some $r \leq N$ and $\Delta t > 0$, both depending on ε. Therefore, convergence of the learned operators to their intrusive counterparts is guaranteed in appropriate limits. Conversely, it is clear from the lack of commutativity between Galerkin projection and the flow maps corresponding to (9) and (16) (see, e.g., [20]) that solutions to the OpInf problem (18) will never pre-asymptotically recover the corresponding intrusive operators appearing in (16). This has motivated semi-intrusive methods such as re-projection [43] and rank-aware snapshot collection [48] which are guaranteed to eliminate this troublesome "closure error" that limits ROM performance. Even without these techniques, OpInf remains a powerful method for building and deploying non-intrusive surrogate models in a variety of settings. The next section will discuss its use within the Schwarz coupling framework.

4 The Schwarz alternating method for heterogeneous model coupling

The second ingredient necessary for the present approach is the Schwarz alternating method (SAM). SAM is a "divide-and-conquer" strategy which first decomposes the physical domain on which a given set of PDEs is posed into smaller subdomains. Following this, the governing PDEs are solved by iterating between smaller, subdomain-local problems, with boundary information exchanged to ensure compatibility across subdomain interfaces (see Algorithm 1). After overviewing the overlapping SAM for FOM-FOM coupling in solid dynamics (Section 4.1), we describe our extension of the approach to non-intrusive OpInf ROMs (Section 4.2). We note that this section proposes several novel methodologies for improving the efficiency and robustness of O-SAM for couplings involving OpInf ROMs, in particular: (i) the

---

[4]As before, the operators $\bar{H}$ and $\bar{C}$ are parameterized in terms of their compressed representation, so that a symmetric operator is guaranteed after inference.

introduction of boundary POD bases to reduce the size of the boundary operators $\bar{\boldsymbol{B}}_i$ in the case there are a large number of Dirichlet boundary nodes (Section 4.2.1), and (ii) an efficient and robust algorithm for optimally selecting the regularization parameter in the subdomain-local OpInf least squares optimization problems (Section 4.2.4).

4.1 O-SAM for FOM-FOM coupling

Without loss of generality, consider the decomposition of Ω into two overlapping subdomains, Ω_1 and Ω_2, with boundaries $\partial\Omega_1$ and $\partial\Omega_2$, respectively, such that $\Omega_1 \cap \Omega_2 \neq \emptyset$, as shown in Figure 1. The boundary of each subdomain $\partial\Omega_i = \partial_\varphi\Omega_i \cup \partial_T\Omega_i \cup \partial_S\Omega_i$ is the union of the (exterior) Dirichlet $\partial_\varphi\Omega_i$, the (exterior) Neumann $\partial_T\Omega_i$, and (interior) interface $\partial_S\Omega_i$ parts, which satisfy the non-overlapping condition $\partial_\varphi\Omega_i \cap \partial_T\Omega_i \cap \partial_S\Omega_i = \emptyset$. The interior interface parts are defined as $\partial_S\Omega_1 = \partial\Omega_1 \cap \Omega_2$ and $\partial_S\Omega_2 = \partial\Omega_2 \cap \Omega_1$. We refer the reader to Figure 1 for a depiction.
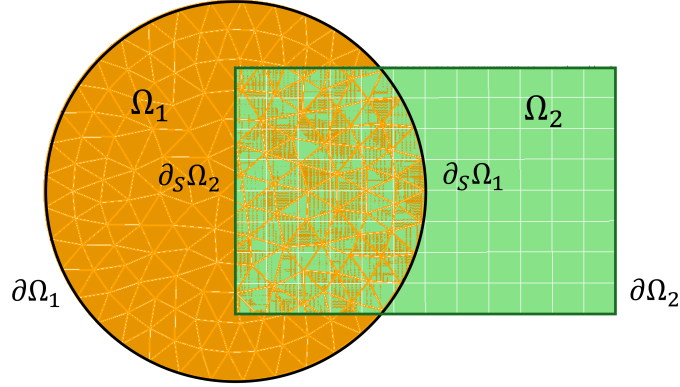

Figure 1: Illustration showing an domain decomposition of a 2D domain Ω into two overlapping subdomains, Ω_1 and Ω_2, for the application of O-SAM.

Furthermore, let the time domain be decomposed into C time intervals (referred to as "controller time-steps" in our earlier work [38]), $I_k = [t_{k-1}, t_k]$, $k = 1, \ldots, C$ with $t_k < t_{k+1}$, $t_C = T$. SAM operates by solving the following sequence of problems for $k = 1, \ldots, C$, with inner Schwarz iterations $n = 1, 2, \ldots$,

$$
\begin{cases}
\nabla \cdot \boldsymbol{P}_1^{(n)} + \rho\boldsymbol{R}_1 = \rho\ddot{\boldsymbol{\varphi}}_1^{(n)} & \text{in } I_k \times \Omega_1, \\
\boldsymbol{\varphi}_1^{(n)}(t, \boldsymbol{X}) = \boldsymbol{\chi} & \text{on } I_k \times \partial_\varphi\Omega_1 \\
\boldsymbol{P}_1^{(n)}\boldsymbol{n} = \boldsymbol{T} & \text{on } I_k \times \partial_T\Omega_1, \\
\boldsymbol{\varphi}_1^{(n)}(t, \boldsymbol{X}) = \Pi_1\boldsymbol{\varphi}_2^{(n-1)}(t, \boldsymbol{X}) & \text{on } I_k \times \partial_S\Omega_1,
\end{cases}
\quad
\begin{cases}
\nabla \cdot \boldsymbol{P}_2^{(n)} + \rho\boldsymbol{R}_2 = \rho\ddot{\boldsymbol{\varphi}}_2^{(n)} & \text{in } I_k \times \Omega_1, \\
\boldsymbol{\varphi}_2^{(n)}(t, \boldsymbol{X}) = \boldsymbol{\chi} & \text{on } I_k \times \partial_\varphi\Omega_2, \\
\boldsymbol{P}_2^{(n)}\boldsymbol{n} = \boldsymbol{T} & \text{on } I_k \times \partial_T\Omega_2, \\
\boldsymbol{\varphi}_2^{(n)}(t, \boldsymbol{X}) = \Pi_2\boldsymbol{\varphi}_1^{(n)}(t, \boldsymbol{X}) & \text{on } I_k \times \partial_S\Omega_2,
\end{cases}
\tag{20}
$$

given an initial guess for $\boldsymbol{\varphi}_2^{(0)}$. In (20), Π_1 denotes the projection of the solution $\boldsymbol{\varphi}_2$ onto $\partial_S\Omega_1$ and Π_2 denotes the projection of the solution $\boldsymbol{\varphi}_1$ onto $\partial_S\Omega_1$. These operators are needed since the subdomains Ω_1 and Ω_2 can be discretized by different meshes having different resolutions and/or element types, as shown in Figure 1. The system (20) is supplemented with initial conditions, as given in (2), restricted to each subdomain. In essence, SAM solves the PDE on Ω_1 using an initial guess to the solution on Ω_2 as a Dirichlet boundary condition on the interface $\partial_S\Omega_1$, and then proceeds to solve the PDE on Ω_2 using the updated solution from Ω_1 as a Dirichlet boundary on the interface $\partial_S\Omega_2$. This iteration is continued until convergence, discussed in more detail below. We emphasize that the Schwarz iteration is not performed in a global space-time setting, but rather over each controller time-step. This is equivalent to a global space-time implementation (due to causality in time), but far more efficient and easier to implement, as discussed in [38].

Remark 5. *Although the discussion and numerical results presented herein assume the coupling of just two subdomains, we emphasize that SAM is capable of coupling an arbitrary number of subdomains. Indeed, we have considered SAM-based couplings of three or more subdomains in several of our past works, e.g., [37, 51, 35, 55]. However, since we consider herein DDs that are physically motivated, we are not*

targeting problems having greater than 10 subdomains; in fact, it is rare to encounter a problem requiring more than 4–5 subdomains.

Algorithm 1: O-SAM for FOM-FOM coupling in solid dynamics and the specific case of two overlapping subdomains.

Input : Overlapping subdomains Ω_1 and Ω_2 each with a mesh, initial condition $(\boldsymbol{\varphi}_1^{(0)}(t_0, \boldsymbol{X})$ and $\boldsymbol{\varphi}_2^{(0)}(t_0, \boldsymbol{X})$, respectively), time integrator, uniform time-step (Δt_1 and Δt_2, respectively).

Output : Converged solutions $\boldsymbol{\varphi}_1(t, \boldsymbol{X})$ and $\boldsymbol{\varphi}_2(t, \boldsymbol{X})$ to the subproblems in (20) in Ω_1 and Ω_2, respectively.

1 Divide the global time $[0, T]$ into C time intervals $I_k = [t_{k-1}, t_k]$, for $k = 1, ..., C$;

2 for $k = 1$ *to* C **do**

3   Initialize $n = 1$;

4   Initialize *SAM converged* to false ;

5   **while** *SAM converged is false* **do**

6   Project $\boldsymbol{\varphi}_2^{(n-1)}(t, \boldsymbol{X})$ onto $\partial_S \Omega_1$ in space and in time to obtain $\Pi_2 \boldsymbol{\varphi}_2^{(n-1)}(t, \boldsymbol{X})$ for $t = t_{k-1}, t_{k-1} + \Delta t_1, t_{k-1} + 2\Delta t_1, ..., t_k$;

7   Advance the Ω_1 subproblem in (20) from time t_{k-1} to t_k to obtain $\boldsymbol{\varphi}_1^{(n)}(t, \boldsymbol{X})$ for $t = t_{k-1}, t_{k-1} + \Delta t_1, t_{k-1} + 2\Delta t_1, ..., t_k$;

8   Project $\boldsymbol{\varphi}_1^{(n)}(t, \boldsymbol{X})$ onto $\partial_S \Omega_2$ in space and in time to obtain $\Pi_1 \boldsymbol{\varphi}_1^{(n)}(t, \boldsymbol{X})$ for $t = t_{k-1}, t_{k-1} + \Delta t_2, t_{k-1} + 2\Delta t_2, ..., t_k$;

9   Advance the Ω_2 subproblem in (20) from time t_{k-1} to t_k to obtain $\boldsymbol{\varphi}_1^{(n)}(t, \boldsymbol{X})$ for $t = t_{k-1}, t_{k-1} + \Delta t_2, t_{k-1} + 2\Delta t_2, ..., t_k$;

10   Check SAM convergence criteria (22) at time t_k ;

11   **if** *SAM convergence criteria satisfied* **then**

12   Set *SAM converged* to true

13   **else**

14   Increment $n = n + 1$;

15   **end**

16   **end**

17 end

In practice, we implement SAM at the fully discrete level, and potentially utilize disparate spatio-temporal discretizations on the different subdomains. Again, without loss of generality, we consider discretization in space using standard isoparametric finite elements and discretization in time using the Newmark-β method. For time discretization, we partition each controller time-step I_k, $k = 1, \ldots, C$ into n_1 and n_2 time intervals with time-step Δt_1 and Δt_2 on Ω_1 and Ω_2, respectively; for notational simplicity, we assume a uniform number of time-steps in each time interval, as shown in Figure 2. The total number of time-steps[5] in Ω_i is thus $\tau_i = n_i C$. The fully discrete SAM iteration[6] for the unconstrained DoFs over the kth time interval is: find $\boldsymbol{u}_1^{j+1,(n)}$, $\boldsymbol{u}_2^{m+1,(n)}$ such that, for $j = n_1(k-1) + 1, \ldots, n_1 k$, $m = n_2(k-1), \ldots, n_2 k$, $n = 1, 2, \ldots$,

$$\frac{\boldsymbol{M}_1}{\Delta t_1 \beta} \boldsymbol{u}_1^{j+1,(n)} + \boldsymbol{K}_1 \left(\boldsymbol{u}_1^{j+1,(n)} \right) = \boldsymbol{f}_1 \left(\boldsymbol{s}_1^{j+1,(n-1)}, \boldsymbol{\chi}_{h,1}^{j+1} \right) + \boldsymbol{M}_1 \left[\frac{1}{\Delta t_1^2 \beta} \boldsymbol{u}_1^j + \frac{1}{\Delta t_1 \beta} \dot{\boldsymbol{u}}_1^j + \frac{1}{\beta} \ddot{\boldsymbol{u}}_1^j \right],$$

$$\frac{\boldsymbol{M}_2}{\Delta t_1 \beta} \boldsymbol{u}_2^{m+1,(n)} + \boldsymbol{K}_2 \left(\boldsymbol{u}_2^{m+1,(n)} \right) = \boldsymbol{f}_2 \left(\boldsymbol{s}_2^{m+1,(n)}, \boldsymbol{\chi}_{h,2}^{m+1} \right) + \boldsymbol{M}_2 \left[\frac{1}{\Delta t_2^2 \beta} \boldsymbol{u}_2^m + \frac{1}{\Delta t_2 \beta} \dot{\boldsymbol{u}}_2^m + \frac{1}{\beta} \ddot{\boldsymbol{u}}_2^m \right],$$

$$\begin{cases} \ddot{\boldsymbol{u}}_i^{\ell+1,(n)} = \frac{1}{\Delta t_i \beta} \left[\boldsymbol{u}_i^{\ell+1,(n)} - \boldsymbol{u}_i^\ell \right] - \frac{1}{\Delta t_i \beta} \dot{\boldsymbol{u}}_i^\ell - \frac{1}{2\beta} (1 - 2\beta) \ddot{\boldsymbol{u}}_i^\ell \\ \dot{\boldsymbol{u}}_i^{\ell+1,(n)} = \dot{\boldsymbol{u}}_i^\ell + (1 - \gamma)\Delta t_i \ddot{\boldsymbol{u}}_i^\ell + \gamma \Delta t_i \ddot{\boldsymbol{u}}_i^{\ell+1,(n)} \end{cases} \quad (i, \ell) = (1, j), (2, m) \tag{21}$$

---

[5]Note that this assumes that the time-steps Δt_i divide evenly the controller time-steps, $t_k - t_{k-1}$. This is depicted in Figure 2.

[6]To simplify the presentation, we note we have suppressed the dependence of stiffness matrices $\boldsymbol{K}_i$ on the Dirichlet and Schwarz boundary conditions.

until convergence is reached based on the absolute or relative error tolerance $\delta_{\text{abs}}, \delta_{\text{rel}} > 0$,

$$
\begin{aligned}
\boldsymbol{u}_i^{n_i k} &= \boldsymbol{u}_i^{n_i k,(n)} \\
\dot{\boldsymbol{u}}_i^{n_i k} &= \dot{\boldsymbol{u}}_i^{n_i k,(n)} \text{ s.t.} \\
\ddot{\boldsymbol{u}}_i^{n_i k} &= \ddot{\boldsymbol{u}}_i^{n_i k,(n)}
\end{aligned}
\left\{
\begin{array}{l}
\sum\limits_{i=1}^{2} \left\| \left(\boldsymbol{u}_i^{n_i k,(n)} - \boldsymbol{u}_i^{n_i k,(n-1)} \right) + \Delta t_i \left(\dot{\boldsymbol{u}}_i^{n_i k,(n)} - \dot{\boldsymbol{u}}_i^{n_i k,(n-1)} \right) \right\|_2^2 < \delta_{\text{abs}}^2, \\
\text{or} \\
\dfrac{\sum\limits_{i=1}^{2} \left\| \left(\boldsymbol{u}_i^{n_i k,(n)} - \boldsymbol{u}_i^{n_i k,(n-1)} \right) + \Delta t_i \left(\dot{\boldsymbol{u}}_i^{n_i k,(n)} - \dot{\boldsymbol{u}}_i^{n_i k,(n-1)} \right) \right\|_2^2}{\sum\limits_{i=1}^{2} \left\| \boldsymbol{u}_i^{n_i k,(n)} + \Delta t \, \dot{\boldsymbol{u}}_i^{n_i k,(n)} \right\|_2^2} < \delta_{\text{rel}}^2.
\end{array}
\right.
\tag{22}
$$

In the above, γ, β are parameters of the Newmark-β method, $\boldsymbol{u}_i^{j,(n)} \in \mathbb{R}^{N_i}$ denotes the solution at the nth Schwarz iteration for the N_i unconstrained DoFs for displacements on Ω_i, $i = 1, 2$, at the jth time-step on the interval. Additionally, $\boldsymbol{M}_i \in \mathbb{R}^{N_i \times N_i}$ and $\boldsymbol{K}_i : \mathbb{R}^{N_i} \to \mathbb{R}^{N_i}$, $i = 1, 2$ are the mass matrix and stiffness, while $\boldsymbol{f}_i : \mathbb{R}^{N_i^S} \times \mathbb{R}^{N_i^B} \to \mathbb{R}^{N_i}$, $i = 1, 2$ are the forcing vectors which account for the boundary conditions $\boldsymbol{s}_i^{j,(n)}$ on the N_i^S interior interface DoFs and $\boldsymbol{\chi}_{h,i}$ on the N_i^B exterior Dirichlet interface DoFs. Critical to the Schwarz formulation, the interior Schwarz boundary terms $\boldsymbol{s}_i^{j,(n)}$ are given as

$$
\boldsymbol{s}_1^{j,(n)} = \left[\boldsymbol{\Pi}_1 \boldsymbol{u}_2^{1,(n)}, \dots, \boldsymbol{\Pi}_1 \boldsymbol{u}_2^{n_2,(n)} \right] \boldsymbol{\Xi}_1^j \in \mathbb{R}^{N_1^S},
$$

$$
\boldsymbol{s}_2^{m,(n)} = \left[\boldsymbol{\Pi}_2 \boldsymbol{u}_1^{1,(n)}, \dots, \boldsymbol{\Pi}_2 \boldsymbol{u}_1^{n_1,(n)} \right] \boldsymbol{\Xi}_2^m \in \mathbb{R}^{N_2^S},
$$

where $\boldsymbol{\Pi}_1 \in \mathbb{R}^{N_1^S \times N_2}$ performs pointwise projection[7] of the spatial DoFs in Ω_2 onto the boundary $\partial_S \Omega_1$, and $\boldsymbol{\Xi}_1^j \in \mathbb{R}^{n_2 \times 1}$ is a temporal interpolant for the case where non-uniform time schemes are used on the different subdomains. Similarly, $\boldsymbol{\Pi}_2 \in \mathbb{R}^{N_2^S \times N_1}$ and $\boldsymbol{\Xi}_2^m \in \mathbb{R}^{n_1 \times 1}$. Note that interpolation in time is only required for non-equivalent temporal discretizations between the two domains; we again refer the reader to Figure 2.
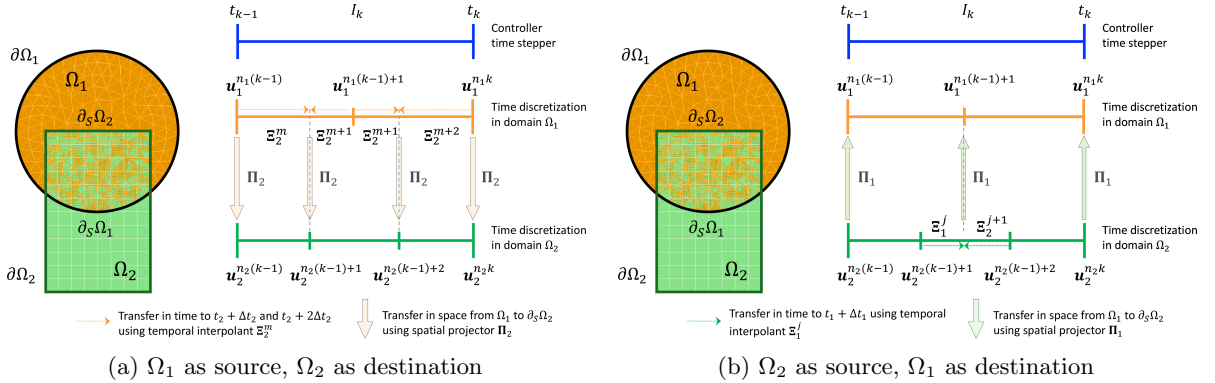

(a) Ω_1 as source, Ω_2 as destination

(b) Ω_2 as source, Ω_1 as destination

Figure 2: Field transfer in the O-SAM algorithm within one controller time-step I_k using the spatial projectors $\boldsymbol{\Pi}_1$ and $\boldsymbol{\Pi}_2$, and temporal interpolants $\boldsymbol{\Xi}_1^j$ and $\boldsymbol{\Xi}_2^m$. (a) Ω_1 serves as the source subdomain, while Ω_2 functions as the destination domain; (b) the roles of the domains are reversed. The super-scripts $j + l$ and $m + l$ correspond to times $t_{k-1} + l\Delta t_1$ and $t_{k-1} + m\Delta t_2$, respectively, and the integers n_i denote the number of time-steps in Ω_i, as described in (21). Thus, $\boldsymbol{u}_i^{n_i(k-1)} \approx \boldsymbol{u}_i(t_{k-1}, \boldsymbol{X})$ and $\boldsymbol{u}_i^{n_i k} \approx \boldsymbol{u}_i(t_k, \boldsymbol{X})$ for $i = 1, 2$.

We emphasize that, while the discussion above details the implementation of O-SAM for the specific case of a Newmark-β time integrator, the subdomain problems in Ω_1 and Ω_2 can be advanced in time using *any* time-integration scheme. This is made clear in Algorithm 1, which outlines in words the general O-SAM scheme as applied to the subdomain-local PDEs in (20). As in the discussion above and in Figure 2, we assume in Algorithm 1 that the subdomain time-steps Δt_1 and Δt_2 divide evenly the controller time-steps.

Remark 6. *The present work restricts attention to the serial version of SAM, commonly known in the linear solver literature as "multiplicative Schwarz" [17]. In particular, the subdomain problems in (20) are solved sequentially. We note that it is possible to improve the efficiency of SAM by implementing its*

---

[7]We note that is possible to use variational projection, as discussed in [40].

"additive" variant [17], in which the subdomain problems in (20) are solved in parallel, with boundary data communicated between subdomains asynchronously as it becomes available. Assessing the performance of aparallelized additive SAM on the test cases considered herein would be an interesting future research endeavor. Our earlier work [55] has shown that, for some 2D nonlinear hyperbolic conservation law problems with a 2×2 domain decomposition, speedups of $\sim 3\times$ are possible by employing parallel additive SAM over its sequential multiplicative analog. In general, the speedup expected will depend on the problem and the domain decomposition.

4.2 O-SAM for OpInf-FOM and OpInf-OpInf coupling

As outlined above, SAM is flexible with respect to different discretizations employed within the different subdomains, and can be formulated for the case where an approximate ROM is used to model a subdomain. We now describe this process for OpInf in terms of the relevant offline and online stages. In the present work, we restrict attention to the case where we construct an OpInf ROM for Ω_1 using snapshots of a corresponding FOM-FOM Schwarz simulation, often referred to in the literature as "top-down" training [8]. The decision to focus on "top-down" training is based on the assumption that a FOM-FOM coupled solution on the full physical domain of interest is generally possible to obtain via SAM using our software frameworks. We note that, since our formulation builds the OpInf ROM on Ω_1 purely from information on Ω_1 and $\partial\Omega_1$, it is, by construction, a *subdomain-local* approach. As such, it is compatible with both "top-down" training and "bottom-up" training.

As in Section 4.1, we present our O-SAM-based OpInf-FOM and OpInf-OpInf formulation assuming all models have been discretized in time using the Newmark-β time-integration scheme. We again emphasize that SAM is extremely flexible and allows for the use of *any* time integration scheme in each subdomain. For clarity, the online and offline stages of O-SAM for OpInf-OpInf coupling with a Newmark-β time-stepping scheme is summarized in Algorithm 2; extensions to OpInf-FOM coupling following the approach described in Section 4.2.2 is straightforward, as is the extension to alternate time integrators.

Remark 7. *We intentionally do not perform training on the monolithic full domain Ω and generate subdomain-local POD bases by restricting or projecting the full domain snapshots onto the subdomains Ω_i (as done in e.g., [35]), since many of our problems of interest are posed on complex geometries that are extremely difficult to mesh without performing a domain decomposition into simpler subdomains.*

4.2.1 Offline stage: basis construction and operator inference

As discussed in Section 3, we postulate an OpInf model for the unconstrained DoFs on Ω_1 of the form

$$\ddot{\hat{\boldsymbol{u}}}_1 + \bar{\boldsymbol{K}}_1 \hat{\boldsymbol{u}}_1 + \bar{\boldsymbol{H}}_1 \hat{\boldsymbol{u}}_1^{\otimes 2} + \bar{\boldsymbol{C}}_1 \hat{\boldsymbol{u}}_1^{\otimes 3} = \bar{\boldsymbol{B}}_1 \hat{\boldsymbol{g}}_1, \tag{23}$$

where $\hat{\boldsymbol{u}}_1(t) \in \mathbb{R}^{r_1}$ are the reduced coordinates for the solution on Ω_1 with the associated basis $\boldsymbol{\Phi}_1 \in \mathbb{V}_{r_1}\left(\mathbb{R}^{N_1}\right)$, and $\bar{\boldsymbol{K}}_1 \in \mathbb{R}^{r_1 \times r_1}$, $\bar{\boldsymbol{H}}_1 \in \mathbb{R}^{r_1 \times r_1^2}$, $\bar{\boldsymbol{C}}_1 \in \mathbb{R}^{r_1 \times r_1^3}$ are the linear, quadratic, and cubic reduced operators, respectively. Lastly, key to the Schwarz formulation, $\bar{\boldsymbol{B}}_1$ and $\hat{\boldsymbol{g}}_1$ are used to model the Dirichlet boundary contributions; this will now be discussed.

Unlike the monolithic OpInf formulation presented in Section 3, in the Schwarz formulation the boundary term must model both the essential Dirichlet boundary condition, $\boldsymbol{\varphi}_1(t, \boldsymbol{X}) = \boldsymbol{\chi}$ on $I \times \partial_\varphi\Omega_1$, as well as the Schwarz overlap boundary condition, $\boldsymbol{\varphi}_1(t, \boldsymbol{X}) = \boldsymbol{\varphi}_2(t, \boldsymbol{X})$ on $I \times \partial_S\Omega_1$. The approach employed in [35] is to set

$$\hat{\boldsymbol{g}}_1 := \left(\begin{array}{cc} \boldsymbol{\Pi}_1^\mathsf{T}\boldsymbol{u}_2^\mathsf{T}, & \boldsymbol{\chi}_{h,1}^\mathsf{T} \end{array} \right)^\mathsf{T} \in \mathbb{R}^{N_1^S + N_1^B}.$$

This approach, however, results in an OpInf operator that scales with the FOM dimension as $\bar{\boldsymbol{B}}_1 \in \mathbb{R}^{r_1 \times \left(N_1^S + N_1^B\right)}$.

For 3D problems, the number of interface DoFs for $N_1^S + N_1^B$ can be large. To mitigate this issue, we employ a two-stage approach for the boundary operator wherein we employ POD to reduce the dimension of the boundary condition forcing vector and solve an OpInf problem in a reduced space. We set

$$\hat{\boldsymbol{g}}_1 := \left(\begin{array}{cc} \boldsymbol{\Phi}_{\partial_S\Omega_1}^\mathsf{T}\boldsymbol{\Pi}_1^\mathsf{T}\boldsymbol{u}_2^\mathsf{T}, & \boldsymbol{\Phi}_{\partial_\varphi\Omega_1}^\mathsf{T}\boldsymbol{\chi}_{h,1}^\mathsf{T} \end{array} \right)^\mathsf{T} \in \mathbb{R}^{r_1^S + r_1^B},$$

where $\boldsymbol{\Phi}_{\partial_\varphi\Omega_1} \in \mathbb{V}_{r_1^B}\left(\mathbb{R}^{N_1^B}\right)$ with $r_1^B \ll N_1^B$ and $\boldsymbol{\Phi}_{\partial_S\Omega_1} \in \mathbb{V}_{r_1^S}\left(\mathbb{R}^{N_1^S}\right)$ with $r_1^S \ll N_1^S$ are orthonormal bases for the essential Dirichlet and Schwarz Dirichlet boundary conditions, respectively. The above scales independently of the FOM.

To construct the OpInf operators and bases, we suppose that the Schwarz FOM problem (21) has been solved in time for $\tau_1 + 1$ separate states[8], yielding the collection of unconstrained state snapshots on Ω_1

$$\boldsymbol{U}_1 = [\boldsymbol{u}_1^0, \boldsymbol{u}_1^2, \ldots, \boldsymbol{u}_1^{\tau_1}] \in \mathbb{R}^{N_1 \times \tau_1 + 1}, \tag{24}$$

where $\boldsymbol{u}_1^j \approx \boldsymbol{u}_1(j\Delta t_1)$. We additionally assume access to the set of snapshots for the essential Dirichlet boundary conditions and Schwarz Dirichlet boundary conditions,

$$\boldsymbol{G}_1 = \left[\boldsymbol{\chi}_{h,1}^0, \ldots, \boldsymbol{\chi}_{h,1}^{\tau_1} \right] \in \mathbb{R}^{N_1^B \times \tau_1 + 1},$$
$$\boldsymbol{S}_1 = \left[\boldsymbol{s}_1^0, \ldots, \boldsymbol{s}_1^{\tau_1} \right] \in \mathbb{R}^{N_1^S \times \tau_1 + 1}.$$

Similarly to Section 3, we construct bases $\boldsymbol{\Phi}_1 \in \mathbb{R}^{N_1 \times r_1}$, $\boldsymbol{\Phi}_{\partial_\varphi \Omega_1}$, $\boldsymbol{\Phi}_{\partial_S \Omega_1}$ for the unconstrained DoFs on Ω_1, essential Dirichlet boundary conditions, and Schwarz Dirichlet boundary conditions by employing POD on the snapshot matrices $\boldsymbol{U}_1$, $\boldsymbol{G}_1$ and $\boldsymbol{S}_1$, respectively. We again emphasize that the training data could be constructed using alternate approaches, e.g., a "bottom-up" approach by simulation the FOM on Ω_1 only for, e.g., varying boundary conditions.

Using the collected snapshot matrices $\boldsymbol{U}_1$, $\boldsymbol{G}_1$, $\boldsymbol{S}_1$ and the reduced bases $\boldsymbol{\Phi}_1$, $\boldsymbol{\Phi}_{\partial_\varphi \Omega_1}$, and $\boldsymbol{\Phi}_{\partial_S \Omega_1}$, the OpInf operators are inferred from solving the least-squares problem

$$\underset{\bar{\boldsymbol{K}}_1, \bar{\boldsymbol{H}}_1, \bar{\boldsymbol{C}}_1, \bar{\boldsymbol{B}}_1}{\arg\min} \left\| D_t^2(\bar{\boldsymbol{U}}_1) + \bar{\boldsymbol{K}}_1 \bar{\boldsymbol{U}}_1 + \bar{\boldsymbol{H}}_1 \bar{\boldsymbol{U}}_1^{*2} + \bar{\boldsymbol{C}}_1 \bar{\boldsymbol{U}}_1^{*3} - \bar{\boldsymbol{B}}_1 \left[\left[\boldsymbol{\Phi}_{\partial_\varphi \Omega_1} \boldsymbol{G}_1^\intercal \right]^\intercal \left[\boldsymbol{\Phi}_{\partial_S \Omega_1}^\intercal \boldsymbol{S}_1^\intercal \right]^\intercal \right]^\intercal \right\|_F^2 +$$
$$\lambda \left(\|\bar{\boldsymbol{K}}_1\|_F^2 + \|\bar{\boldsymbol{H}}_1\|_F^2 + \|\bar{\boldsymbol{C}}_1\|_F^2 + \|\bar{\boldsymbol{B}}_1\|_F^2 \right). \tag{25}$$

Similar to standard OpInf presented in Section 3, the formulation (25) is non-intrusive and the OpInf ROM can be inferred directly from solving a convex minimization problem with a least-squares objective.

4.2.2 Online stage: OpInf-FOM coupling

Discretizing the OpInf ROM with Newmark-β, the fully discrete O-SAM iteration for an OpInf-FOM coupled system over the kth controller time interval are, for $j = n_1(k-1) + 1, \ldots, n_1 k$, $m = n_2(k-1), \ldots, n_2 k$, $n = 1, 2, \ldots$,

$$\frac{\hat{\boldsymbol{u}}_1^{j+1,(n)}}{\Delta t_1 \beta} + \bar{\boldsymbol{K}}_1 \hat{\boldsymbol{u}}_1^{j+1,(n)} + \bar{\boldsymbol{H}}_1 \hat{\boldsymbol{u}}_1^{j+1,(n)\otimes 2} + \bar{\boldsymbol{C}}_1 \hat{\boldsymbol{u}}_1^{j+1,(n)\otimes 3} = \bar{\boldsymbol{B}}_1 \hat{\boldsymbol{g}}_1^{j+1,(n-1)} + \frac{1}{\Delta t_1^2 \beta} \hat{\boldsymbol{u}}_1^j + \frac{1}{\Delta t_1 \beta} \dot{\hat{\boldsymbol{u}}}_1^j + \frac{1}{\beta} \ddot{\hat{\boldsymbol{u}}}_1^j,$$

$$\frac{\boldsymbol{M}_2}{\Delta t_2 \beta} \boldsymbol{u}_2^{m+1,(n)} + \boldsymbol{K}_2 \left(\boldsymbol{u}_2^{m+1,(n)} \right) = \boldsymbol{f}_2 \left(\tilde{\boldsymbol{s}}_2^{m+1,(n)}, \boldsymbol{\chi}_h^{m+1} \right) + \boldsymbol{M}_2 \left[\frac{1}{\Delta t_2^2 \beta} \boldsymbol{u}_2^m + \frac{1}{\Delta t_2 \beta} \dot{\boldsymbol{u}}_2^m + \frac{1}{\beta} \ddot{\boldsymbol{u}}_2^m \right],$$

$$\ddot{\hat{\boldsymbol{u}}}_1^{j+1,(n)} = \frac{1}{\Delta t_1 \beta} \left[\hat{\boldsymbol{u}}_1^{j+1,(n)} - \hat{\boldsymbol{u}}_1^j \right] - \frac{1}{\Delta t_1 \beta} \dot{\hat{\boldsymbol{u}}}_1^j - \frac{1}{2\beta} (1 - 2\beta) \ddot{\hat{\boldsymbol{u}}}_1^j,$$

$$\dot{\hat{\boldsymbol{u}}}_1^{j+1,(n)} = \dot{\hat{\boldsymbol{u}}}_1^j + (1 - \gamma) \Delta t_1 \ddot{\hat{\boldsymbol{u}}}_1^j + \gamma \Delta t_1 \ddot{\hat{\boldsymbol{u}}}_1^{j+1,(n)},$$

$$\ddot{\boldsymbol{u}}_2^{m+1,(n)} = \frac{1}{\Delta t_2 \beta} \left[\boldsymbol{u}_2^{m+1,(n)} - \boldsymbol{u}_2^m \right] - \frac{1}{\Delta t_2 \beta} \dot{\boldsymbol{u}}_2^m - \frac{1}{2\beta} (1 - 2\beta) \ddot{\boldsymbol{u}}_2^m,$$

$$\dot{\boldsymbol{u}}_2^{m+1,(n)} = \dot{\boldsymbol{u}}_2^m + (1 - \gamma) \Delta t_2 \ddot{\boldsymbol{u}}_2^m + \gamma \Delta t_2 \ddot{\boldsymbol{u}}_2^{m+1,(n)} \tag{26}$$

where the boundary couplings are defined as

$$\hat{\boldsymbol{g}}_1^{j,(n)} = \left[\left[\boldsymbol{\Phi}_{\partial_S \Omega_1}^\intercal \boldsymbol{s}_1^{j,(n)} \right]^\intercal, \ldots, \left[\boldsymbol{\Phi}_{\partial_\varphi \Omega_1}^\intercal \boldsymbol{\chi}_h^j \right]^\intercal \right]^\intercal$$
$$\tilde{\boldsymbol{s}}_2^{m,(n)} = \left[\boldsymbol{\Pi}_2 \boldsymbol{\Phi}_1 \hat{\boldsymbol{u}}_1^{1,(n)}, \ldots, \boldsymbol{\Pi}_2 \boldsymbol{\Phi}_1 \hat{\boldsymbol{u}}_1^{n_1,(n)} \right] \boldsymbol{\Xi}_2^m.$$

The iteration is continued until convergence, as described in (22). We emphasize the following aspects of the formulation (26):

1. As stated earlier in Remark 1, while the subdomain-local OpInf ROM in an OpInf-FOM coupling is based on a polynomial approximation of the form (16), the subdomain-local FOM to which it is coupled is based on the full nonlinear PDEs (20). This is evident in (26).

---

[8]Note that the number of snapshots in Ω_1 and Ω_2, denoted by τ_1 and τ_2, respectively, can be different, as our algorithm allows one to use different time-steps in different subdomains when applying O-SAM.

2. The OpInf ROM and FOM are coupled through forcing terms that describe the Schwarz Dirichlet boundary condition.

3. When communicating the ROM information on Ω_1 to the FOM on Ω_2 through the term $\tilde{\boldsymbol{s}}_2^m$, the boundary term needs to be computed in the full order space. This is achieved via the trial basis, $\boldsymbol{\Phi}_1$, which in practice can be restricted to the relevant boundary DoFs to accelerate the computation.

4. When communicating the FOM information on Ω_2 to the ROM on Ω_1 through the term $\boldsymbol{s}_1^j$, the FOM information is projected onto the reduced boundary space described by $\boldsymbol{\Phi}_{\partial_S\Omega_1}$.

5. For clarity, the OpInf-OpInf coupling algorithm described above is summarized at a relatively high-level in Algorithm 2 in terms of the offline and online stages of the model reduction and coupling procedure.

4.2.3 Online stage: OpInf-OpInf coupling

We highlight that the above formulation can be easily translated to perform Schwarz-based couplings for various subdomain-local OpInf models. Specifically, one can repeat Section 4.2.1 for Ω_2, and a coupled OpInf-OpInf system can be formulated as

$$\frac{\hat{\boldsymbol{u}}_1^{j+1,(n)}}{\Delta t_1\beta} + \bar{\boldsymbol{K}}_1\hat{\boldsymbol{u}}_1^{j+1,(n)} + \bar{\boldsymbol{H}}_1\hat{\boldsymbol{u}}_1^{j+1,(n)\otimes 2} + \bar{\boldsymbol{C}}_1\hat{\boldsymbol{u}}_1^{j+1,(n)\otimes 3} = \bar{\boldsymbol{B}}_1\hat{\boldsymbol{g}}_1^{j+1,(n-1)} + \frac{1}{\Delta t_1^2\beta}\hat{\boldsymbol{u}}_1^j + \frac{1}{\Delta t_1\beta}\dot{\hat{\boldsymbol{u}}}_1^j + \frac{1}{\beta}\ddot{\hat{\boldsymbol{u}}}_1^j,$$

$$\frac{\hat{\boldsymbol{u}}_2^{m+1,(n)}}{\Delta t_2\beta} + \bar{\boldsymbol{K}}_1\hat{\boldsymbol{u}}_2^{m+1,(n)} + \bar{\boldsymbol{H}}_1\hat{\boldsymbol{u}}_2^{m+1,(n)\otimes 2} + \bar{\boldsymbol{C}}_1\hat{\boldsymbol{u}}_2^{m+1,(n)\otimes 3} = \bar{\boldsymbol{B}}_1\hat{\boldsymbol{g}}_2^{m+1,(n)} + \frac{1}{\Delta t_2^2\beta}\hat{\boldsymbol{u}}_2^m + \frac{1}{\Delta t_2\beta}\dot{\hat{\boldsymbol{u}}}_2^m + \frac{1}{\beta}\ddot{\hat{\boldsymbol{u}}}_2^m,$$

along with the associated Newmark-β updates. The boundary couplings are now defined as

$$\hat{\boldsymbol{g}}_1^{j,(n)} = \left[\left[\boldsymbol{\Phi}_{\partial_S\Omega_1}^{\intercal}\tilde{\boldsymbol{s}}_1^{j,(n)}\right]^{\intercal}, \ldots, \left[\boldsymbol{\Phi}_{\partial_\varphi\Omega_1}^{\intercal}\boldsymbol{\chi}_h^j\right]^{\intercal}\right]^{\intercal}, \qquad \hat{\boldsymbol{g}}_2^{m,(n)} = \left[\left[\boldsymbol{\Phi}_{\partial_S\Omega_2}^{\intercal}\tilde{\boldsymbol{s}}_2^{m,(n)}\right]^{\intercal}, \ldots, \left[\boldsymbol{\Phi}_{\partial_\varphi\Omega_2}^{\intercal}\boldsymbol{\chi}_h^m\right]^{\intercal}\right]^{\intercal},$$

$$\tilde{\boldsymbol{s}}_1^{j,(n)} = \left[\boldsymbol{\Pi}_1\boldsymbol{\Phi}_2\hat{\boldsymbol{u}}_2^{1,(n)}, \ldots, \boldsymbol{\Pi}_1\boldsymbol{\Phi}_2\hat{\boldsymbol{u}}_2^{n_2,(n)}\right]\boldsymbol{\Xi}_1^j, \qquad \tilde{\boldsymbol{s}}_2^{m,(n)} = \left[\boldsymbol{\Pi}_2\boldsymbol{\Phi}_1\hat{\boldsymbol{u}}_1^{1,(n)}, \ldots, \boldsymbol{\Pi}_2\boldsymbol{\Phi}_1\hat{\boldsymbol{u}}_1^{n_1,(n)}\right]\boldsymbol{\Xi}_2^m.$$

It is straightforward to modify Algorithm 2 to perform OpInf-FOM coupling.

4.2.4 Offline stage: Selection of regularization parameter

The stability and accuracy of the OpInf ROM is strongly dependent on the value of the regularization parameter, λ. Various strategies have been proposed to identify the optimal regularization parameter, and here we use an extension of the approach proposed in [33]. In essence, the approach proposed in [33] repeatedly solves the inference problem for various regularization parameters. For each regularization parameter, the resulting OpInf ROM is integrated in time using the same initial condition as the training data. The regularization parameter resulting in the OpInf ROM with the lowest trajectory error (as measured against the training data) is then selected.

The above approach is made more complicated in the present setting as the OpInf ROM is coupled to another subdomain model through the Schwarz Dirichlet boundary condition, and as such cannot be integrated independently. While one could solve the coupled system for each regularization parameter, this would be prohibitively expensive for the case where the coupled subdomain model is a FOM. To mitigate this issue, we employ a subdomain-local training process for the OpInf ROM regularization parameter by interpreting the OpInf ROM as a monolithic problem with boundary conditions taken from training data. That is, for each regularization parameter considered for the OpInf ROM, we advance the ROM according to the OpInf specific portion of (26) with the Schwarz boundary condition $\boldsymbol{s}_i^{(n)}$ taken from training data.

5 Numerical results

We now present some numerical results that illustrate the performance of our proposed O-SAM-based couplings involving disparate models, meshes, time integration schemes, and time-steps. Our Schwarz-based coupling approach has been implemented within `Norma.jl` [27], an open-source[9] three-dimensional

---

[9]`Norma.jl` is available on GitHub: https://github.com/sandialabs/Norma.jl. For the numerical results presented herein, the following version of `Norma.jl` was utilized: `d11e20114a9aeb9d2427c03a946c89fee729b3d1`.

Algorithm 2: O-SAM for OpInf-OpInf coupling in solid dynamics and the specific case of two overlapping subdomains advanced forward in time using a Newmark-β time-integration scheme.

Offline stage:

1 Perform O-SAM-based FOM-FOM coupled simulations on two overlapping subdomains Ω_1 and Ω_2;

2 Collect snapshot matrices $\boldsymbol{U}_i$, $\boldsymbol{G}_i$, and $\boldsymbol{S}_i$ from the above coupled simulation in Ω_i, $i = 1, 2$.;

3 Compute POD bases $\boldsymbol{\Phi}_1$ and $\boldsymbol{\Phi}_2$ in Ω_1 and Ω_2 using the snapshot sets $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$. ;

4 Compute boundary POD bases $\boldsymbol{\Phi}_{\partial_\varphi \Omega_1}$, $\boldsymbol{\Phi}_{\partial_\varphi \Omega_2}$, $\boldsymbol{\Phi}_{\partial_S \Omega_1}$, and $\boldsymbol{\Phi}_{\partial_S \Omega_2}$ from the snapshots $\boldsymbol{G}_1, \boldsymbol{G}_2$, $\boldsymbol{S}_1$, and $\boldsymbol{S}_2$. ;

5 Assume a functional form for the ROM in each subdomain, e.g., the cubic form (17) ;

6 Compute OpInf operators in each subdomain by solving the regularized least-squares minimization problems (18) in each subdomain with optimal regularization parameter selection described in Section 4.2.4;

Online stage:

1 Apply the O-SAM coupling iteration procedure in Section 4.2.3, with Schwarz BC transfer via the pre-learned boundary $\bar{\boldsymbol{B}}_1$ and $\bar{\boldsymbol{B}}_2$ in (4.2.3) ;

(3D) Julia-based finite element code designed for the rapid prototyping of algorithms and ideas for domain coupling and contact in solid mechanics. At the time of writing of this paper, the main branch of `Norma.jl` is able to perform: (i) overlapping and non-overlapping coupling of high-fidelity quasistatic [37] and dynamic [38] solid mechanics models in solid mechanics, (ii) overlapping coupling of dynamic non-intrusive linear, quadratic and cubic OpInf models, (iii) overlapping coupling of dynamic non-intrusive NN-based models with prescribed symmetry and definiteness properties [41], and (iv) contact simulations using a recently-proposed non-overlapping SAM-based contact enforcement algorithm [40]. The code additionally supports several elastic and hyper-elastic material models, including linear elastic (Appendix A.1), Saint Venant–Kirchhoff [22] (Appendix A.2) and Neohookean [39] (Appendix A.3). For time integration, `Norma.jl` employs the usual Newmark-β scheme [36], which can be run either implicitly or explicitly. In the explicit variant of the scheme, mass lumping is employed by default, so as to improve efficiency by avoiding linear solves during the time integration procedure.

The OpInf models coupled herein are trained using a "top-down" approach, in which training data are generated by running an O-SAM-based FOM-FOM coupled simulation on the same physical geometries of interest. Once training data are generated by performing a coupled FOM-FOM O-SAM simulation using `Norma.jl`, subdomain-local and boundary POD bases are constructed for the vector-valued displacement degrees of freedom and the appropriate OpInf operators are learned offline with the help of the open-source[10] `norma-opinf` Python package [26] (Algorithm 2, steps 3–4). Finally, an online coupled ROM-ROM or FOM-ROM simulation (Section 4.2) is performed by running `Norma.jl` once again, after feeding it the learned operators and bases obtained from `norma-opinf`. In general, the POD basis dimension is selected based on an energy criterion, so that the modes retained capture some (large) percentage of the total variance in the snapshot set. That is, we define the POD energy as

$$E_r = \frac{\sum_{i=1}^{r} \sigma_i^2}{\sum_{i=1}^{N} \sigma_i^2} \in [0, 1], \tag{27}$$

where σ_i are the ordered singular values of the snapshot matrix (see Section 3.1), and select the reduced dimension r such that $E_r < \delta_E$, where $\delta_E \in (0, 1)$. The quantity $100 E_r$ is typically referred to as the snapshot energy percentage captured by a given reduced basis $\boldsymbol{\Phi}_r$. In addition to representing the solution living in the subdomain interiors using a POD basis, we also construct separate POD bases for the DoFs at which either Schwarz or system Dirichlet boundary conditions are imposed to reduce further the online evaluation costs of the ROMs, as discussed earlier in Section 4.2.1.

All numerical experiments discussed in this paper were performed on a Linux RHEL9 workstation, known as `Rigel`, which has two AMD EPYC 9634 ("Genoa") 84-core processors and 1.5 TB DDR5 ECC RAM. Since `Norma.jl` does not, at the present time, possess MPI parallelism, all simulations were performed on one core. While `Norma.jl` has hyper-threading capabilities, hyper-threading was not

---

[10]`norma-opinf` is available on GitHub: https://github.com/sandialabs/norma-opinf. For the numerical results presented herein, the following version of `norma-opinf` was utilized: `763e37198e197919f4f78dcee4b26efb4809c1cd`.

utilized in the experiments reported on herein. All test cases considered, including the 1D linear elastic wave propagation discussed in Section 5.1, were run as three-dimensional problems. Solution accuracy was assessed in terms of the mean square relative error within a given subdomain Ω for a given solution field $\boldsymbol{u}$ with respect to a reference solution $\boldsymbol{u}_{\text{ref}}$, defined as:

$$\mathcal{E}_\Omega(\boldsymbol{u}) := \frac{\sqrt{\sum_{k=0}^\tau ||\boldsymbol{u}(t_k) - \boldsymbol{u}_{\text{ref}}(t_k)||_2^2}}{\sqrt{\sum_{k=0}^\tau ||\boldsymbol{u}_{\text{ref}}(t_k)||_2^2}}, \tag{28}$$

where $\boldsymbol{u}(t_k)$ is the solution at time t_k (and similarly for $\boldsymbol{u}_{\text{ref}}(t_k)$) and τ is the total number of time intervals considered. We take as the reference solution $\boldsymbol{u}_{\text{ref}}(t)$ either the exact analytical solution (for the problem in Section 5.1) or the FOM solution in a given subdomain Ω_i obtained by performing a FOM-FOM coupling using O-SAM (for the problems in Sections 5.2–5.4). We do not perform comparisons with respect to a monolithic solution computed on the full domain Ω, as it is often very difficult (and sometimes impossible) to robustly mesh a full geometry of practical interest (e.g., the bolted joint problem considered in Section 5.2). For the 1D linear elastic wave propagation problem considered in Section 5.1, we also calculated projection errors associated with a given POD basis $\boldsymbol{\Phi}_r$, domain Ω and snapshot set $\boldsymbol{u}$, defined as:

$$e_\Omega(\boldsymbol{u}, \boldsymbol{\Phi}_r) := \frac{||\boldsymbol{u} - \boldsymbol{\Phi}_r \boldsymbol{\Phi}_r^\intercal \boldsymbol{u}||_2}{||\boldsymbol{u}||_2}. \tag{29}$$

The projection error $e(\boldsymbol{u}, \boldsymbol{\Phi}_r)$ is a measure of how well the basis $\boldsymbol{\Phi}_r$ is capable of representing the solution $\boldsymbol{u}$, and sets a lower bound on the error that it is achievable by a ROM defined in the latent space spanned by $\boldsymbol{\Phi}_r$.

We build and assess subdomain-local OpInf ROMs of all three types described in Section 3: linear, quadratic and cubic, referred to as OpInf, QOpInf and COpInf, respectively. The first two problems, the 1D linear elastic wave propagation problem (Section 5.1) and the 3D bolted joint problem (Section 5.2) specify material models that give rise to PDEs with linear and cubic nonlinearities, respectively; hence, we assess OpInf- and COpInf-based couplings for these problems, respectively. While the second two problems, the 3D torsion and 3D tension specimen problems, specify material models which give rise to PDEs with generic (non-polynomial) nonlinearities, we construct and assess couplings involving QOpInf-based models, which we demonstrate are reasonable local surrogates having moderate computational complexity. We utilize the algorithm detailed in Section 4.2.4 to find the "optimal" values of the regularization parameters in the OpInf least-squares minimization problems. In particular, we search over a logarithmically-spaced grid of regularization parameters ranging from 10^{-4} to 1, i.e., we search over $\lambda = \{1 \times 10^{-4+\Delta[j-1]}\}_{j=1}^{n_\lambda}$, with $\Delta = \frac{4}{n_\lambda - 1}$; in our experiments we set $n_\lambda = 40$. For convergence of the Schwarz alternating method, we employ both an absolute and a relative convergence criterion based on the displacement as well as the velocity fields, see (22).

When it comes to deciding which model (e.g., ROM vs. FOM) to assign to which subdomain, we generally take the strategy reducing the subdomains with the less complex dynamics. While, admittedly, smaller online CPU time improvements are anticipated with this approach, we expect that it will generate a coupled model that is more accurate and robust, a necessity for solid mechanics analyses. Our model assignment strategy is similar to the one considered in [19] but contrary to the one in [54] and [42], which opts to reduce subdomains discretized with finer meshes in order to achieve the best online speed-ups.

While this work builds on the preliminary studies presented in [35, 47], we emphasize that this paper is the *first* to demonstrate domain decomposition-based couplings involving non-intrusive OpInf ROMs using O-SAM for: (i) realistic 3D nonlinear problems in solid mechanics with geometries discretized using different mesh resolutions and element types, (ii) the coupling of disparate time integration schemes with possibly different time-steps, and (iii) the coupling of higher-order QOpInf and COpInf models with each other and with FOMs. This also differentiates the present work from [16] and [19].

Before showcasing our results, we note that SI units of measurement are employed throughout this paper, unless otherwise noted. All meshes employed in our simulations were generated using the CUBIT meshing software [25] developed at Sandia National Laboratories. We remind the reader that, as stated in Section 1, the domain decompositions we consider in the present work and in general are physically-motivated, meaning that the number of subdomains being coupled will in generally be small (at most 5–10, but usually 2–3). As discussed in Remark 5, our method is applicable to an arbitrary number of subdomains. We additionally remind the reader that only the sequential multiplicative Schwarz variant is considered herein, as detailed in Remark 6.

More information about the code versions used and where to find input decks to reproduce the results presented herein can be found in the "Code availability and reproducibility" section at the end of this paper.

16

5.1   1D linear elastic wave propagation problem

The first test case we consider is the so-called 1D linear elastic wave propagation problem, variants of which can be found in [38, 1, 47]. The purpose of this example is to verify our method's ability to couple disparate subdomain-local models (FOMs and OpInf ROMs) discretized using different time integration schemes (implicit and explicit Newmark-β) having potentially different time-steps.

Consider a simple beam geometry having a length of 1 m in the z–dimension and a cross-sectional area of 1×1 mm^2, so that $\Omega = (-5.0 \times 10^{-4}, 5.0 \times 10^{-4})^2 \times (-0.5, 0.5)$. The objective is to specify a 1D problem using the 3D `Norma.jl` code. Toward this effect, we set a homogeneous Dirichlet boundary condition on the x– and y–displacement at the $x, y = \pm 5.0 \times 10^{-4}$ boundaries. We additionally assume that the beam is clamped at the ends in the z–dimension, which translates to a homogeneous Dirichlet boundary condition on the z–displacement at the $z = -0.5$ and $z = 0.5$ boundaries. We prescribe within Ω a simple linear elastic material model (Appendix A.1) with Young's modulus $E = 1$ GPa, density $\rho = 1000$ kg/m^3 and Poisson's ratio $\nu = 0$. We initialize the problem by specifying an initial displacement of the form:

$$\boldsymbol{u}(\boldsymbol{x}, 0) = \begin{pmatrix} 0, & 0, & f(z) \end{pmatrix}^{\mathsf{T}}, \tag{30}$$

for a specified function $f(z)$ with $z \in \Omega$, and a zero initial velocity $\boldsymbol{v}(\boldsymbol{x}, 0) = \boldsymbol{0}$, where $\boldsymbol{x} := \begin{pmatrix} x, & y, & z \end{pmatrix}^{\mathsf{T}}$ is the coordinate vector. We will assume the problem is run from time $t = 0$ to time $t = T = 1.0 \times 10^{-3}$ s. It is straightforward to show using the method of characteristics that the exact analytical displacement solution to the problem described above is

$$\boldsymbol{u}_{\text{ref}}(\boldsymbol{x}, t) = \begin{pmatrix} 0, & 0, & \frac{1}{2}f(z-ct) + \frac{1}{2}f(z+ct) - \frac{1}{2}f(z-c(T-t)) - \frac{1}{2}(z+c(T-t)) \end{pmatrix}^{\mathsf{T}}, \tag{31}$$

where $c := \sqrt{E/\rho}$ is the speed of sound [38]. The exact analytical solution for the velocity can be derived by differentiating (31) in time, and is not given here for the sake of brevity.

We will consider herein two initial conditions for our 1D linear elastic wave propagation problem, termed the "Symmetric Gaussian" and the "Rounded Square" initial condition, respectively, and summarized in Table 1. As can be seen by comparing the solutions plotted in Figures 10 and 14, the Rounded Square variant of the problem gives rise to a solution with sharper gradients, which are more difficult to resolve, especially using data-driven models such as OpInf ROMs.

Table 1: 1D linear elastic wave propagation problem: initial conditions considered.

Initial Condition	$f(z)$	a	b	s
Symmetric Gaussian	$a\exp\left(-\frac{(z-b)^2}{2s^2}\right)$	1.0×10^{-3}	0	2.0×10^{-2}
Rounded Square	$a\tanh(-b(z+0.5-s))$ $+ \, a\tanh(b(z-0.5+s))$	5.0×10^{-4}	100	0.6

The first step in applying O-SAM to this problem is to generate a domain decomposition of the physical geometry Ω into overlapping subdomains Ω_i, and to define spatial as well as temporal discretizations of the Ω_i. Toward this effect, we will consider a simple decomposition of Ω into two subdomains, Ω_1 and Ω_2, where $\Omega_1 = (-5.0 \times 10^{-4}, 5.0 \times 10^{-4})^2 \times (-0.5, 0.25)$ and $\Omega_2 = (-5.0 \times 10^{-4}, 5.0 \times 10^{-4})^2 \times (-0.25, 0.5)$, so that the overlap region $\Omega_1 \cap \Omega_2$ has a length of 0.5 m in the z–dimension. Since the problem considered in this section is effectively 1D, we will use uniform conformal hexahedral discretizations of Ω_1 and Ω_2 having spatial increments $\Delta x = \Delta y = \Delta z = 1.0 \times 10^{-3}$ m, but will focus on assessments involving various combinations of subdomain-local FOMs and OpInf ROMs, as well as time-steppers and time steps (see Table 2). All time-steppers utilized are of the Newmark-β type, with $\beta = 0.25$, $\gamma = 0.5$ for the implicit variant, and $\beta = 0$, $\gamma = 0.5$ for the explicit variant. It is straightforward to show that the stable time step according to the Courant–Friedrichs–Levy (CFL) condition for the explicit Newmark-β stepper is $\Delta t = \Delta z/c = 1.0 \times 10^{-6}$ s. The reader can observe from Table 2 that all time steps employed are approximately one order of magnitude smaller than this value, so that stability and accuracy are assured. A controller time-step of 1.0×10^{-7} s was employed for all runs.

For couplings involving OpInf ROMs, we consider three different training/testing scenarios:

- *Scenario 1: reproductive Symmetric Gaussian.* Here, we train the OpInf ROM(s) by simulating the 1D linear elastic wave propagation problem with the Symmetric Gaussian initial condition, and assess the coupled models' ability to reproduce the problem solution with the same initial condition.

Table 2: 1D linear elastic wave propagation problem: models, Newmark-β time discretizations and time steps considered for the couplings evaluated herein.

Ω_1	Ω_2	Time-stepper Ω_1	Ω_2	Time step (s) Ω_1	Ω_2
FOM	FOM	Implicit	Implicit	1.0×10^{-7}	
FOM	OpInf	Explicit	Implicit	1.0×10^{-7}	
OpInf	OpInf	Explicit	Explicit	1.0×10^{-7}	
		Explicit	Implicit	1.0×10^{-7}	2.0×10^{-7}

- *Scenario 2: reproductive Rounded Square.* We next train the OpInf ROM(s) by simulating the 1D linear elastic wave propagation problem with the Rounded Square initial condition, and again evaluate the coupled models' skill in reproducing the problem solution with this initial condition.

- *Scenario 3: predictive Symmetric Gaussian.* Finally, we train the OpInf ROM(s) using solution data from the Rounded Square variant of the 1D linear elastic wave propagation problem, and study the coupled models' ability to predict the solution to the problem with the Symmetric Gaussian initial condition.

Although we performed FOM-FOM couplings involving all four time discretizations summarized in Table 2, we trained all the OpInf models using snapshot data obtained by running the Implicit-Implicit FOM-FOM.

As mentioned earlier, errors for all the coupled models for the 1D linear elastic wave propagation problem were calculated with respect to the exact analytical solution, derived from (31). Since the problem is effectively one-dimensional and the meshes used to discretize the subdomains Ω_i are conformal, it is straightforward to obtain a single-domain solution from our O-SAM coupled solutions by simply averaging the solution values in the overlap region and applying the mean square error formula (28). This allows us to report a single error value, corresponding to the error in the full domain Ω, in assessing the accuracy of our models. For all O-SAM coupled runs, very tight Schwarz tolerances of $\delta_{abs} = 1.0 \times 10^{-8}$ and $\delta_{rel} = 1.0 \times 10^{-12}$ were utilized. For the implicit Newmark-β runs, the relative and absolute linear solver tolerances were 1.0×10^{-10} and 1.0×10^{-6}, respectively.

Before presenting results for the three scenarios described above and the various couplings summarized in Table 2, we perform some diagnostic assessments to determine how accurate we expect our O-SAM coupled models involving OpInf ROMs to be. Figure 3(a) shows the singular value decay as a function of the POD basis size for the Symmetric Gaussian and Rounded Square variants of the 1D linear elastic wave propagation problem. The reader can observe that significantly more POD modes are needed to capture the same fraction of the snapshot energy for the Rounded Square version of this problem than for the Symmetric Gaussian version, as expected: to capture 99.9999% of the snapshot energy, 56 POD modes are required for the former, compared to 28 modes for the latter. Moreover, the singular value decay is much slower for the Rounded Square problem variant. The fact that the singular value decay in Ω_2 is identical to that in Ω_1 is expected, given the symmetry of the problem solutions (31) with respect to $z = 0$. For all test cases reported on here, we select boundary bases based on a 99.9999% energy criterion, which gives rise to bases consisting of just one POD mode per boundary.

Having examined the POD singular value decay, we now look at projection errors for the three problem scenarios described above, calculated using (29) for Ω_2 (without loss of generality) and reported in Figures 3(b)–(c). It can be seen from this figure that approximately 30 modes are sufficient to capture the displacement field for all three problem variants to an error of $\mathcal{O}(1\%)$, including the predictive one. A greater number of modes is needed achieve the same projection error for the velocity field, especially for the Rounded Square version of this problem which exhibits a steep gradient (Figure 3(c)). This is expected for two reasons: (i) the velocity field is generally harder to capture, and (ii) the reduced basis Φ_r is built from snapshots of only the displacement field, as discussed in Remark 2.

5.1.1 Scenario 1: reproductive Symmetric Gaussian

We first discuss results for Scenario 1, in which the Symmetric Gaussian variant of the 1D linear elastic wave propagation problem is run in the reproductive regime. Figure 4 is a convergence plot showing the displacement and velocity relative errors with respect to the exact analytic solution (31) as a function of the number of POD modes for a variety of FOM-OpInf and OpInf-OpInf couplings. For comparison
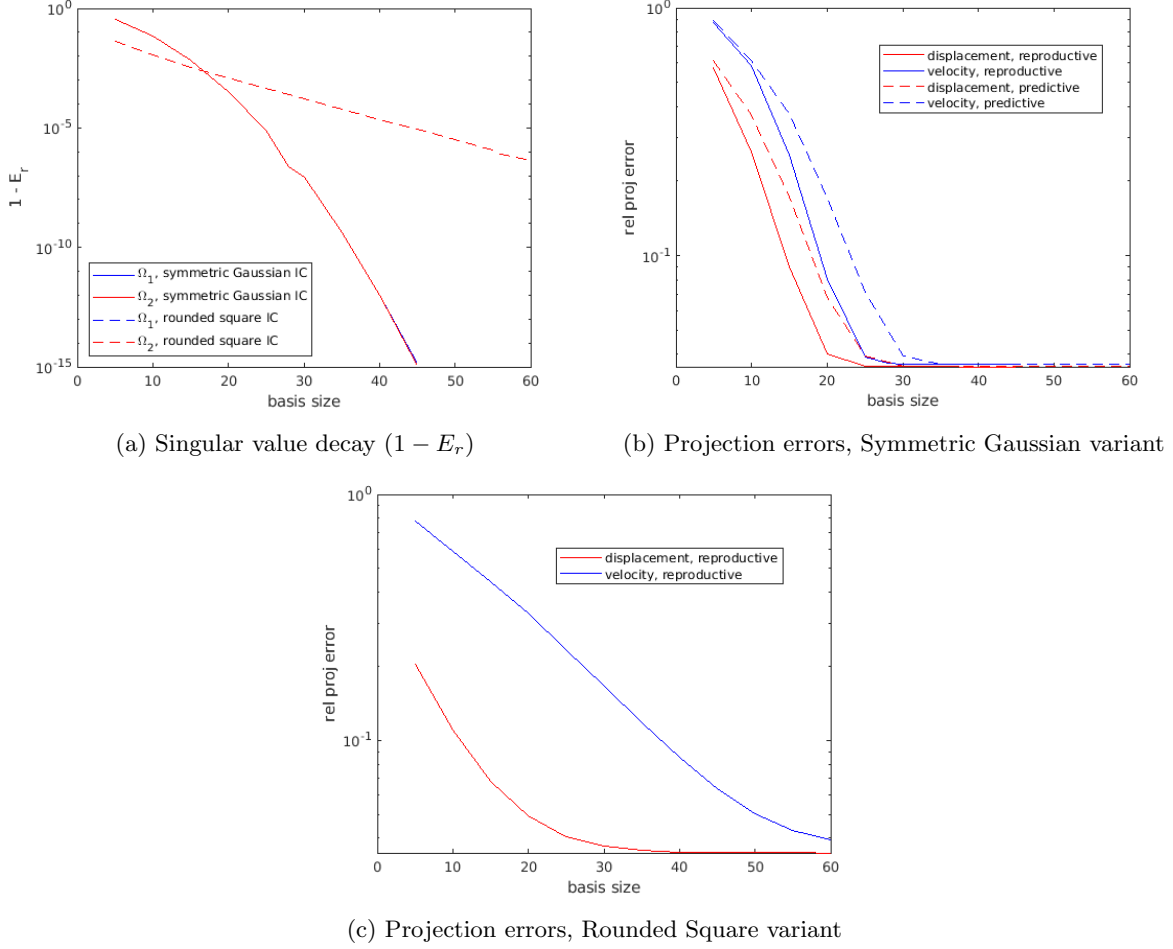
(a) Singular value decay $(1 - E_r)$ (b) Projection errors, Symmetric Gaussian variant

(c) Projection errors, Rounded Square variant

Figure 3: 1D linear elastic wave propagation problem: singular value decay (a) and projection errors calculated in Ω_2 for the Symmetric Gaussian (b) and Rounded Square (c) problem variant, as functions of the POD basis size.

purposes, relative errors achieved by various analogous FOM-FOM couplings are indicated with dashed horizontal lines. It is interesting to observe that, for basis sizes larger than approximately 25 modes, all FOM-OpInf and OpInf-OpInf couplings deliver solutions that are actually *more* accurate than analogous FOM-FOM couplings. We attribute this result to the fact that the couplings involving OpInf models rely on shape functions that are derived from solutions of the underlying PDE being solved, whereas the FOM-FOM couplings rely on generic and problem-agnostic finite element shape functions. The use of specialized problem-specific shape functions, namely POD modes, seems to enable our FOM-OpInf and OpInf-OpInf models to achieve a sort of super-convergence. The plateauing of the errors for large enough basis sizes can be attributed to the closure error inherent in the OpInf process, which arrests convergence in the pre-asymptotic regime.

Additionally, the reader can observe by examining Figures 4(a) and (c) that the Explicit-Implicit FOM-OpInf coupling for which the same time-step is employed in each subdomain delivers the most accurate solution, whereas the Implicit-Implicit FOM-OpInf coupling is the least accurate. This trend is different from the trend observed for analogous FOM-FOM couplings, for which the Implicit-Implicit model is the clear "winner" in terms of accuracy. Moreover, employing different time-steps in the two subdomains does not lead to the same decline in accuracy for the FOM-OpInf coupled models as it does for analogous FOM-FOM coupled models. From Figures 4(b) and (d), it can be seen that all OpInf-OpInf coupled models, regardless of the time integration scheme or time step employed, deliver roughly the same accuracy for all basis sizes considered. This accuracy is comparable to that of all but one of the FOM-OpInf coupled models.

Having studied accuracy, we now turn our attention to convergence and performance. Figure 5 shows
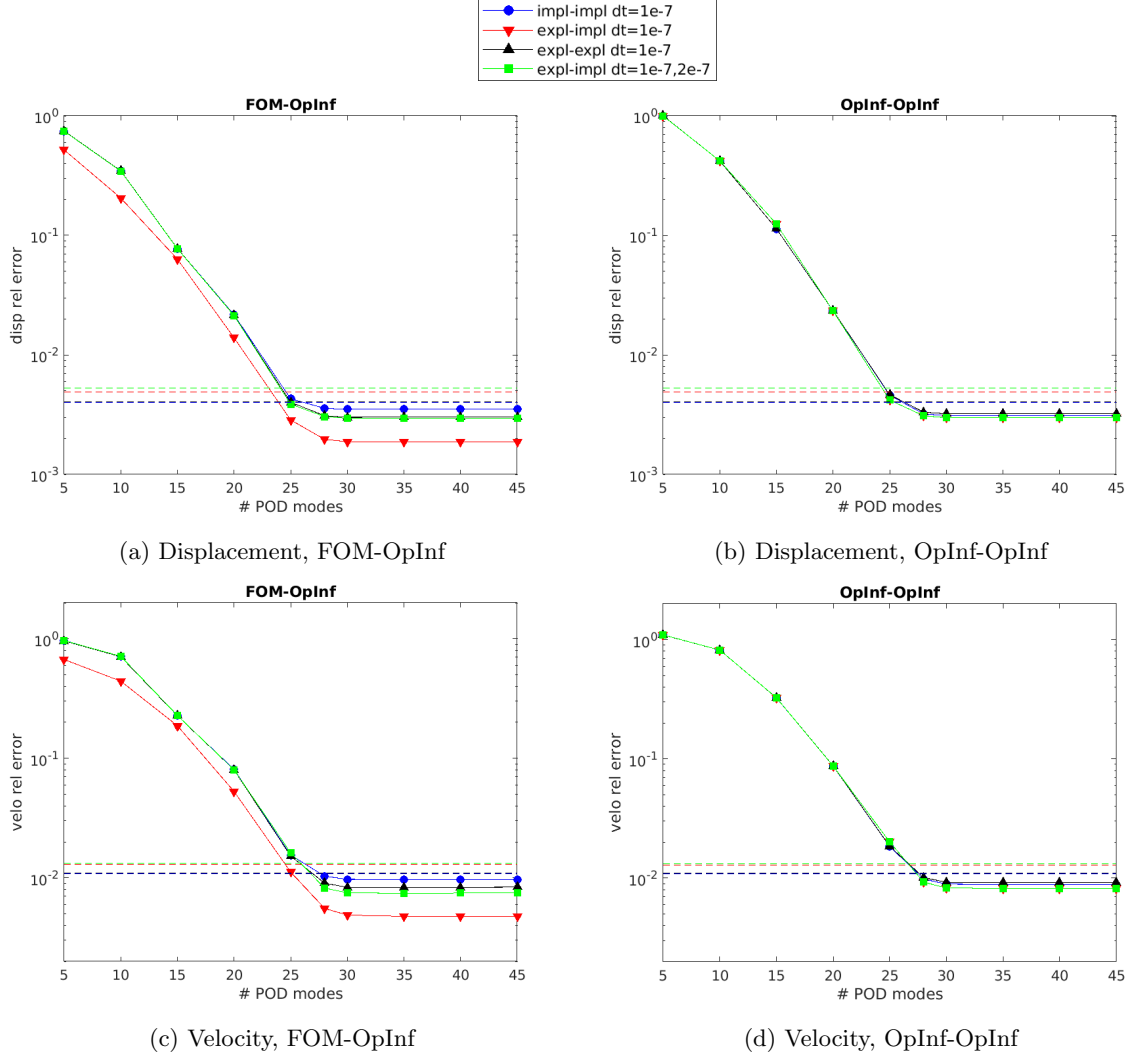
Figure 4: 1D linear elastic wave propagation problem, Symmetric Gaussian initial condition, reproductive regime: displacement (top row) and velocity (bottom row) relative errors with respect to the exact analytical solution for various FOM-OpInf (a) and OpInf-OpInf (b) O-SAM couplings. Dashed horizontal lines show relative errors for FOM-FOM O-SAM couplings with the colors designated in the legend.

the mean number of Schwarz iterations required to achieve convergence for a variety of FOM-OpInf and OpInf-OpInf coupled models, in comparison to analogous FOM-FOM couplings, indicated by the dashed horizontal lines. The reader can observe from Figure 5(a) that, for smaller basis sizes, more Schwarz iterations are required to converge the O-SAM method when performing FOM-OpInf couplings compared to analogous FOM-FOM couplings, but convergence of the former to the latter is observed with basis size refinement. The situation is different for the OpInf-OpInf couplings, however (Figure 5(b)): all OpInf-OpInf couplings converge in exactly two Schwarz iterations, compared to an average of ~ 2.3 Schwarz iterations for similar FOM-FOM couplings. Similar results were obtained for the other test cases considered in this paper. We believe there are two possible explanations for this behavior. First, since our OpInf models rely on POD modes to approximate the solution, these models give rise to solutions that are inherently smoother than their FEM analogues, which can aid convergence of the O-SAM coupling method. Convergence may also be accelerated due to the fact that the shape functions underlying the OpInf-OpInf couplings are problem-specific and data-driven, unlike the problem-agnostic and generic finite element shape functions underlying the FOM-FOM couplings.

Finally, we assess method performance. Figure 6 shows a Pareto plot for the various couplings considered, in which the speedup over a corresponding FOM-FOM coupling is plotted vs. the displacement relative errors. We do not show a Pareto plot for the velocity, as the conclusions in terms of relative
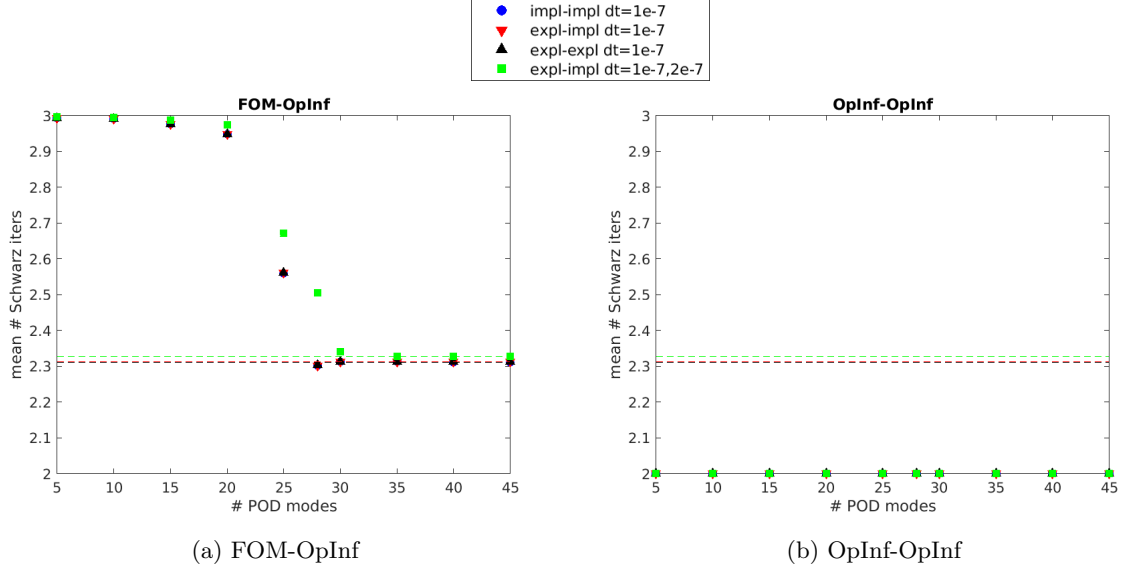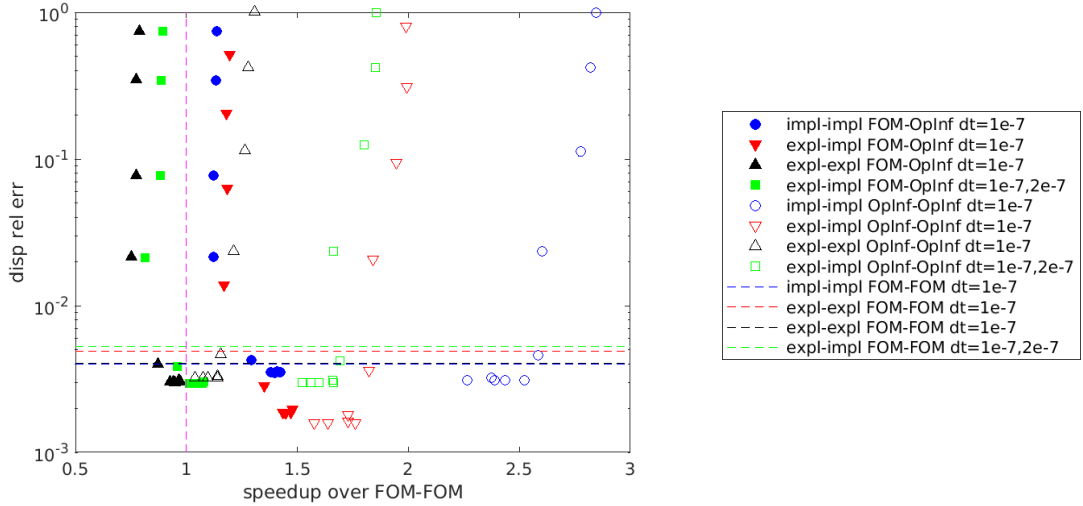
Figure 5: 1D linear elastic wave propagation problem, Symmetric Gaussian initial condition, reproductive regime: mean number of Schwarz iterations required to reach convergence for various FOM-OpInf (a) and OpInf-OpInf (b) O-SAM couplings. Dashed horizontal lines show the number of Schwarz iterations needed to reach convergence for FOM-FOM O-SAM couplings with the colors designated in the legend.

accuracy and efficiency of the various coupled models evaluated would be the same. A speed-up of unity is indicated by a vertical dashed magenta line, and the relative errors achieved by the four FOM-FOM couplings considered are indicated by horizontal dashed lines. It follows that all data points falling to the right of the vertical dashed line, and at/below the horizontal lines are competitive from an accuracy and efficiency standpoint. The reader can observe that the majority of the competitive models are OpInf-OpInf couplings. The largest speed-up achieved of $\sim 2.5\times$ is achieved by the Implicit-Implicit OpInf-OpInf coupled model. As we show below, more substantial speedups are possible for larger, three-dimensional benchmarks, in which larger DoF reductions are possible through the use of subdomain-local OpInf ROMs.

Figure 6: 1D linear elastic wave propagation problem, Symmetric Gaussian initial condition, reproductive regime: Pareto plot showing the speed-up over an analogous FOM-FOM coupling vs. displacement relative error for various FOM-OpInf (filled symbols) and OpInf-OpInf (unfilled symbols) couplings. Dashed horizontal lines indicate relative errors for corresponding FOM-FOM O-SAM couplings. Dashed magenta vertical line indicates a speedup of 1.

5.1.2 Scenario 2: reproductive Rounded Square

Although the takeaways are largely similar, we present, for completeness the reproductive Rounded Square variant of the 1D linear elastic wave propagation problem (Scenario 2), the solution to which is more difficult to represent using ROMs due to the presence of sharp gradients. We focus our discussion on the insights that are different from those discussed earlier in the context of Scenario 1. As can be seen from Figure 7, the Implicit-Implicit FOM-OpInf and OpInf-OpInf coupled models deliver solutions that are noticeably more accurate than their FOM-FOM analogs. This is true not only for the FOM-OpInf couplings but also the OpInf-OpInf couplings, unlike before. The reader can observe by examining Figure 8(a) that the O-SAM convergence behavior is also slightly different for the FOM-OpInf models applied to Scenario 2. Whereas, for Scenario 1, the average number of Schwarz iterations required for O-SAM to converge when coupling FOM-OpInf models converged to the FOM-FOM value with basis refinement, all FOM-OpInf couplings require more Schwarz iterations to converge for Scenario 2. As expected, fewer FOM-OpInf and OpInf-OpInf coupled models are competitive from a combined accuracy and efficiency perspective for the Rounded Square variant of this problem (Figure 9). This can be attributed to the problem being more difficult and requiring more modes to represent the solution, as inferred previously from Figure 3. Finally, in Figure 10, we plot the displacement and velocity solutions for the 45 mode mixed time-step Implicit-Explicit OpInf-OpInf coupled models at several times. While some oscillations can be seen in the velocity solutions, no coupling artifacts or propagation of these oscillations across subdomain boundaries are observed.

5.1.3 Scenario 3: predictive Symmetric Gaussian

Finally, we present results for Scenario 3, the predictive Symmetric Gaussian problem, in which the OpInf ROMs being coupled are trained on data from the Rounded Square variant of this problem and used to predict the solution with the Symmetric Gaussian initial condition. Remarkably, the key results/takeaways from Figures 11–13 are largely the same as those for the Scenario 1 reproductive version of this problem. Figure 14 shows the displacement and velocity solutions produced by our mixed time-step Implicit-Explicit OpInf-OpInf couplings with $r_1 = r_2 = 35$ POD modes at several times. The reader can observe that both solutions are smooth and artifact-free for all times despite this being a predictive problem. The same result is generally *not* observed for predictive monolithic ROMs.

5.2 3D nonlinear hyperelastic bolted joint problem

The aim of the second test case, the 3D bolted joint problem, is to illustrate the plug-and-play nature of the proposed coupling approach on a 3D realistic production-like geometry consisting of large-scale components joined together using small-scale fasteners (Figure 15). Such geometries are notoriously difficult to mesh due to their multiscale nature, and have the potential to benefit greatly from SAM: if one is interested in changing the shape of the geometry and/or including more detail in the bolts (e.g., by adding threading), one can create these alternate geometries/meshes offline and use SAM to seamlessly "glue" them together without having to remesh the entire system conformally.

The specific geometry considered here consists of three pieces: a bottom plate, a top component, and four bolts which join together the plate and the component. The square base of the joint is 127 mm × 127 mm, and the height of the joint is also 127 mm. We prescribe a a nonlinear hyperelastic Saint Venant–Kirchhoff material model [22] with material properties corresponding to those of steel, namely a Young's modulus of 200×10^9 Pa, a Poisson ratio of $\nu = 0.3$ and a density of $\rho = 7,800$ kg/m^3. It can be shown that the Saint-Venant Kirchhoff material model gives rise to PDEs with cubic nonlinearities; the reader is referred to see Appendix A.2 for details. Hence, in our numerical assessment of various FOM-ROM and ROM-ROM couplings, we will focus our attention on COpInf ROMs, which should be capable of representing the nonlinearities in the governing PDEs.

When applying SAM to the 3D bolted joint problem, we consider the natural domain decomposition of the geometry into two subdomains: one containing the four bolts, termed the "bolts," and one containing the bottom plate and top component, termed the "parts." Since the focus of this paper is O-SAM, which requires a non-empty overlap region between the subdomains being coupled, the subdomain containing the bolts also includes a small circular region containing a piece of the bottom plate and top component, as shown in Figures 15(c)–(d). In typical analyses, it is critical to resolve the bolts part of the geometry, so as to accurately characterize strain localization within the bolts and correctly predict possible fastener failure. Toward this effect, we discretize the subdomain containing the bolts, known herein as Ω_b, with a fine ten-node tetrahedral mesh containing 28,508 elements and 44,236 nodes, as shown in Figure 15(c). To
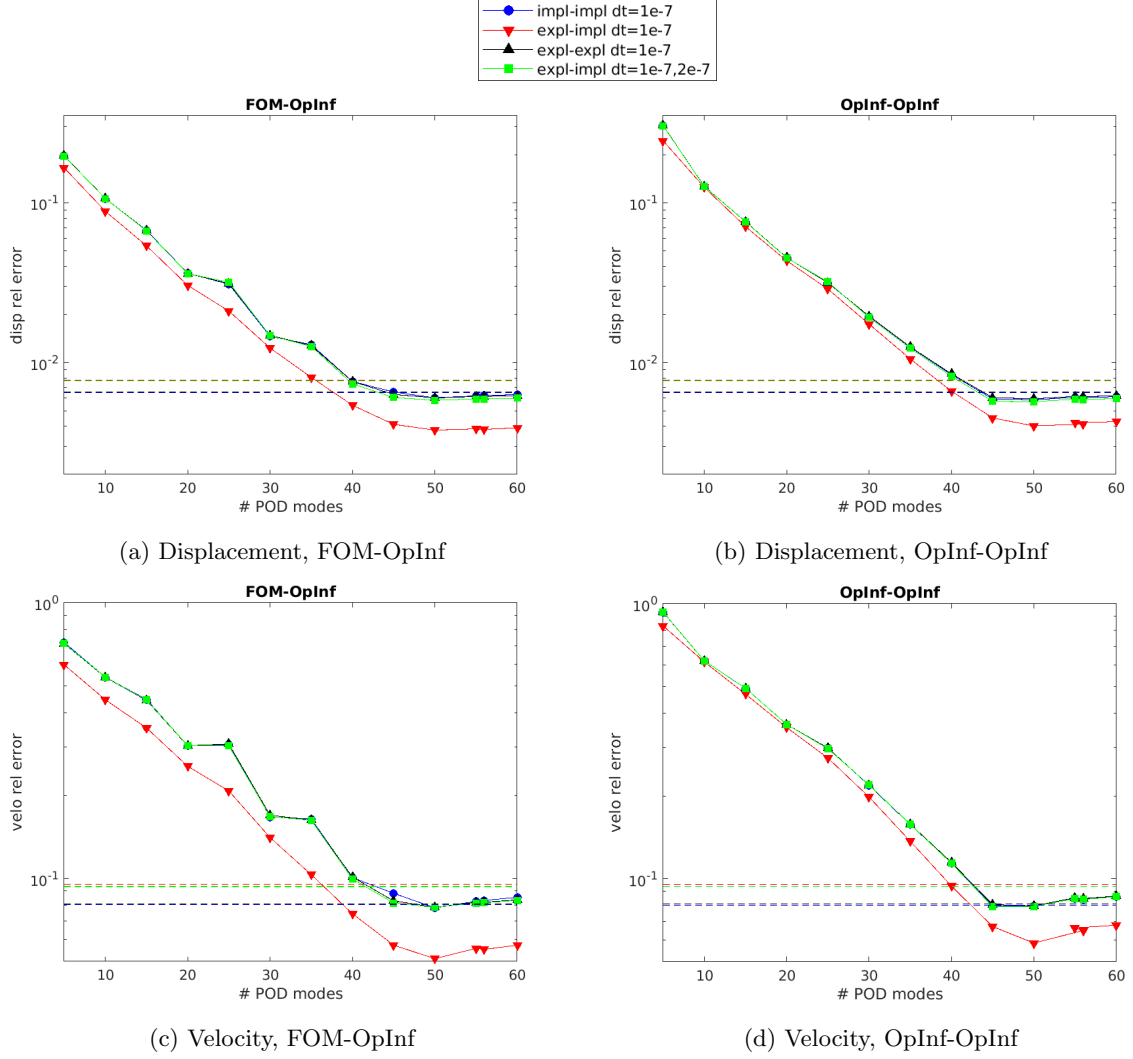
Figure 7: 1D linear elastic wave propagation problem, Rounded Square initial condition, reproductive regime: displacement (top row) and velocity (bottom row) relative errors with respect to the exact analytical solution for various FOM-OpInf (a) and OpInf-OpInf (b) O-SAM couplings. Dashed horizontal lines show relative errors for FOM-FOM O-SAM couplings with the colors designated in the legend.

demonstrate SAM's ability to couple different element types, mesh resolutions and models, we discretize the parts domain, known herein as Ω_p, with a coarse eight-node tetrahedral mesh containing 22,027 elements and 29,327 nodes (Figure 15(b)). Since the accurate calculation of the solution within the bolts is of utmost importance, we are primarily interested in FOM-COpInf couplings, in which a high-fidelity model in Ω_b is coupled to an COpInf ROM in Ω_p; however, for the sake of completeness, we also study herein COpInf-COpInf couplings in which an COpInf ROM is assigned to both subdomains. The problem is initialized with a **0** initial condition for both the displacement and velocity fields.

Remark 8. *As mentioned earlier, the bolted joint problem is an example of a test case where the monolithic governing geometry (Figure 15(a)) is extremely difficult to mesh. While we were able to generate a monolithic tetrahedral mesh of the full geometry, we were unsuccessful in producing a correct solution on this mesh using the classical Galerkin FEM. Running this problem with four-node tetrahedral finite elements yields an under-resolved model that produces non-physical stress patterns, whereas a ten-node tetrahedral mesh is too large and forces a very small time-step, making runs impractical.*
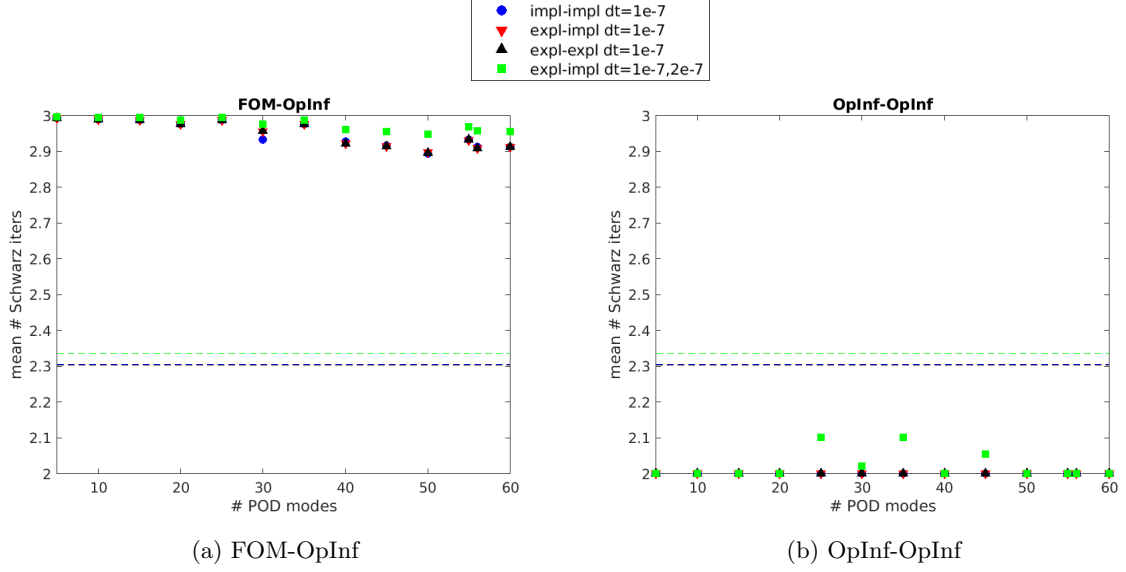
23

Figure 8: 1D linear elastic wave propagation problem, Rounded Square initial condition, reproductive regime: mean number of Schwarz iterations required to reach convergence for various FOM-OpInf (a) and OpInf-OpInf (b) O-SAM couplings. Dashed horizontal lines show the number of Schwarz iterations needed to reach convergence for FOM-FOM O-SAM couplings with the colors designated in the legend.
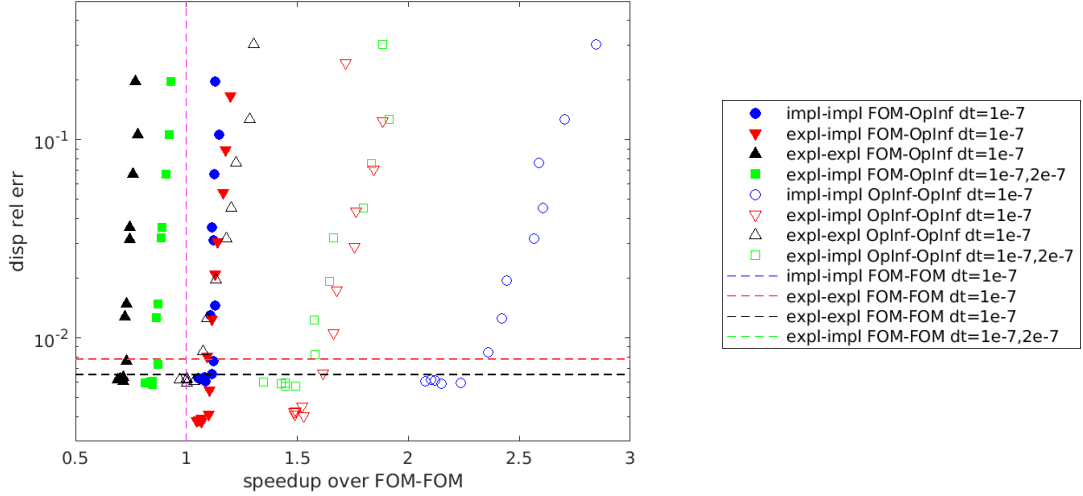

Figure 9: 1D linear elastic wave propagation problem, Rounded Square initial condition, reproductive regime: Pareto plot showing the speed-up over an analogous FOM-FOM coupling vs. the displacement relative errors for various FOM-OpInf (filled symbols) and OpInf-OpInf (unfilled symbols) couplings. Dashed horizontal lines indicate relative errors for corresponding FOM-FOM O-SAM couplings. Dashed magenta vertical line indicates a speedup of 1.

5.2.1 Reproductive variant of the bolted joint problem

As an initial verification, we first study our mixed FOM-OpInf and OpInf-OpInf O-SAM-based couplings on a reproductive version of the bolted joint problem. In this instance of the problem, a zero displacment boundary condition is prescribed on the bottom boundary of Ω_p, while the following time-dependent boundary condition is applied on the top boundary of Ω_p:

$$\boldsymbol{u}(\boldsymbol{x}, t) = \begin{pmatrix} 2 - 2\cos(500\pi t), & 0, & 2 - 2\cos(500\pi t) \end{pmatrix}^{\mathsf{T}}, \tag{32}$$
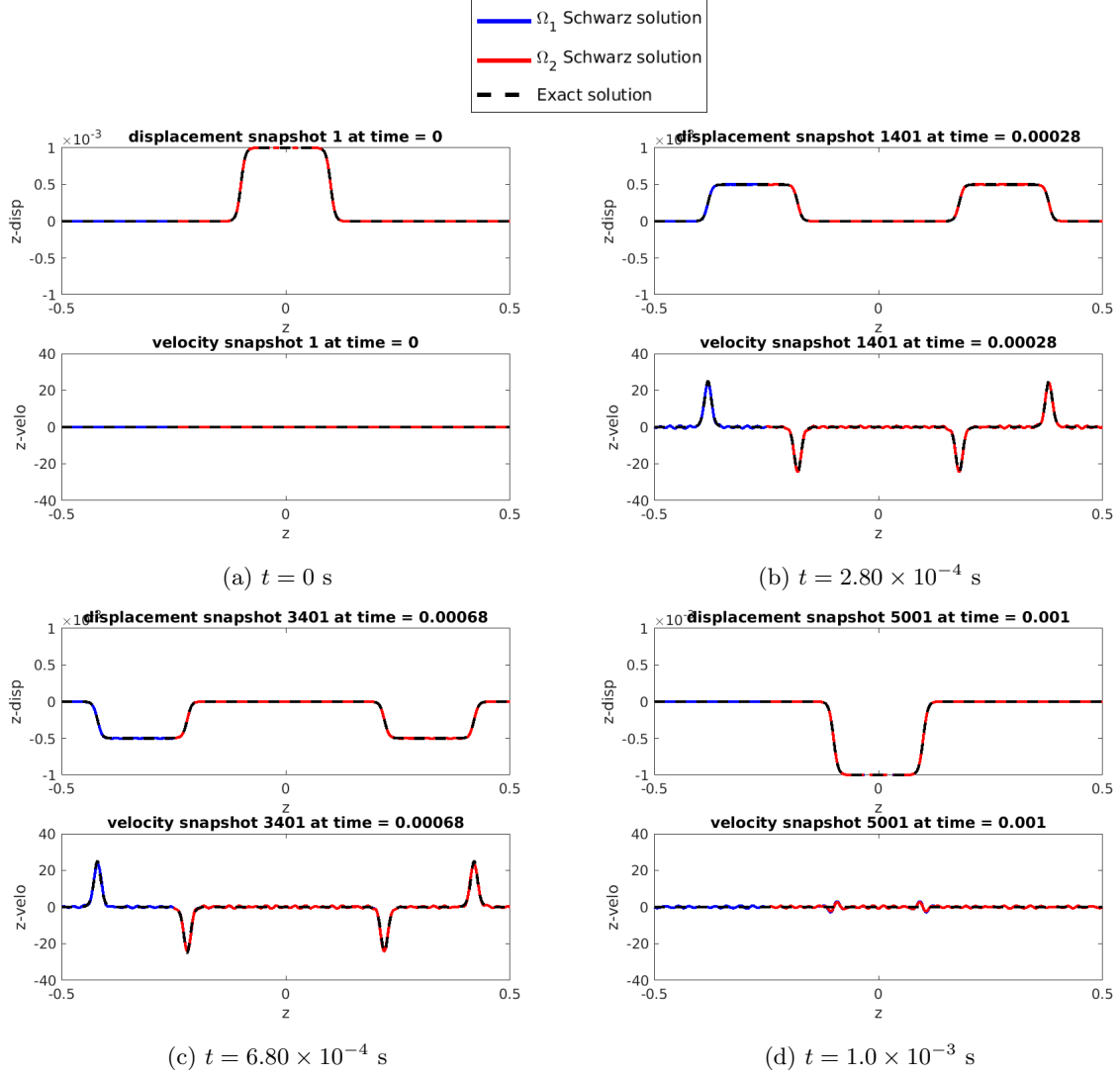
Figure 10: 1D linear elastic wave propagation problem, Rounded Square initial condition, reproductive regime: displacement and velocity computed using our coupled OpInf-OpInf models with $r_1 = r_2 = 45$ and an explicit-implicit scheme with $\Delta t_1 = 1.0 \times 10^{-7}$ s and $\Delta t_2 = 2.0 \times 10^{-7}$ s (blue, red), compared with the exact analytical solution (black). Solutions in Ω_1 are shown in blue, whereas solutions in Ω_2 are shown in red.

for $t \in [0, 5.8 \times 10^{-5}]$ s. We set as an initial condition a zero initial displacement and velocity in both subdomains. The problem is advanced forward in time using an implicit Newmark-β scheme with parameters $\beta = 0.25$ and $\gamma = 0.5$, and a constant time-step of 1.0×10^{-6} s, giving rise to a total of 58 snapshots of the solution. While our SAM-based methodology enables the coupling of disparate time-integrators with different time-steps, as shown earlier in Section 5.1, we do not study such couplings in this section, as it is disadvantageous to use an explicit scheme for the bolted joint problem due to a highly-restrictive CFL condition, which gives rise to a stable time-step of $\mathcal{O}(1 \times 10^{-9})$. Time-steps this small would not be used in production analyses involving this problem. When applying O-SAM in `Norma.jl` [27], we utilize a relative tolerance of $\delta_{\mathrm{rel}} = 1.0 \times 10^{-6}$ and an absolute tolerance of $\delta_{\mathrm{abs}} = 1.0 \times 10^{-4}$ for the Schwarz convergence criteria (22). For the Newton-based nonlinear solver and underlying iterative linear solver, we employ relative and absolute tolerances of 1.0×10^{-7} and 1.0×10^{-5}, respectively. To generate training data for our OpInf models, we run a coupled O-SAM simulation in which a subdomain-local FOM in Ω_b is coupled to a subdomain-local FOM in Ω_p. Running the FOM-FOM coupled model to completion takes 52m 49.5s on the `Rigel` cluster, and requires an average number of 3.21 Schwarz iterations to converge, as reported in Table 3.
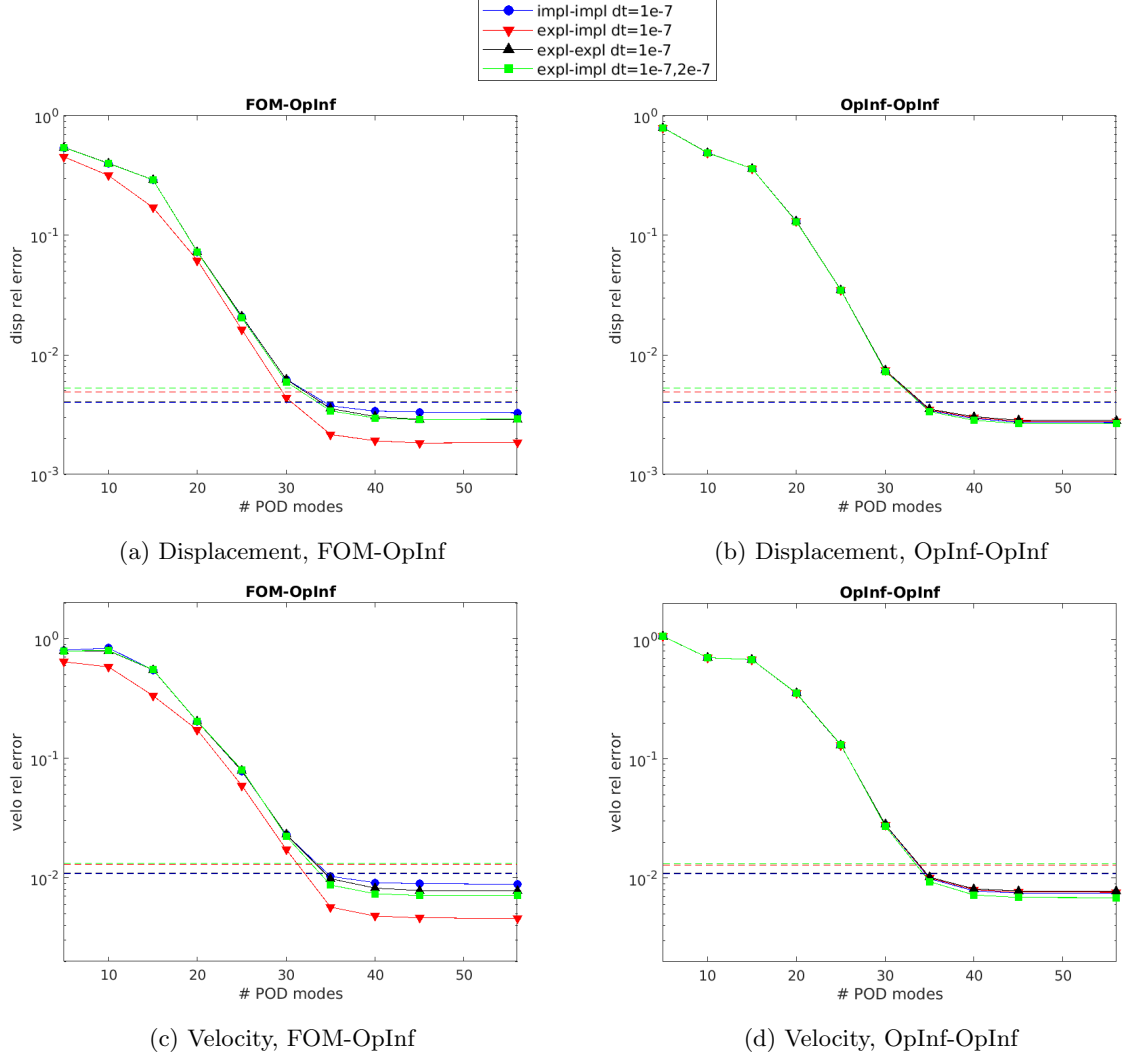
Figure 11: 1D linear elastic wave propagation problem, Symmetric Gaussian initial condition, predictive regime: displacement (top row) and velocity (bottom row) relative errors with respect to the exact analytical solution for various FOM-OpInf (a) and OpInf-OpInf (b) O-SAM couplings. Dashed horizontal lines show relative errors for FOM-FOM O-SAM couplings with the colors designated in the legend.

In order to build our subdomain-local COpInf models, we first apply the POD algorithm to our dataset consisting of 58 snapshots to create a reduced basis $\mathbf{\Phi}_r$. The singular value decay for POD bases of varying sizes calculated for the subdomain interiors, Ω_p and Ω_b, is plotted in Figure 16. It is interesting to observe that the singular value decay is slightly slower in Ω_p than in Ω_b for the reproductive variant of this problem considered in the present subsection. This is likely due to the fact that Ω_p is significantly larger than Ω_b. A POD basis of just 10 modes captures 99.9999% of the snapshot energy in the interior of both subdomains as well as on the Schwarz boundary, while a basis having just 1 mode captures this same energy percentage on the remaining Dirichlet boundaries.

We begin by studying the convergence of the O-SAM algorithm when coupling a subdomain-local COpInf ROM in Ω_b with a subdomain-local FOM in Ω_p with respect to the basis size in Ω_b, denoted by r_b. We consider interior basis sizes that range from 5 to 30 modes for the Ω_b COpInf model, while fixing the boundary basis sizes to their 99.9999% energy values. We fix the regularization parameters in the Ω_b and Ω_p subdomains, denoted by λ_b and λ_p, respectively, to their "optimal" values of 10^{-4} or 10^{-5}, as calculated by our estimation algorithm described in Section 4.2.4. These values are reported in Table 3. Curiously, for the FOM-COpInf couplings, increasing the basis size often leads to a decrease in the CPU time. This can be attributed to the fact that fewer Schwarz iterations are required for convergence.

Figure 17(a) plots the relative error in the displacement magnitude, the velocity magnitude and the
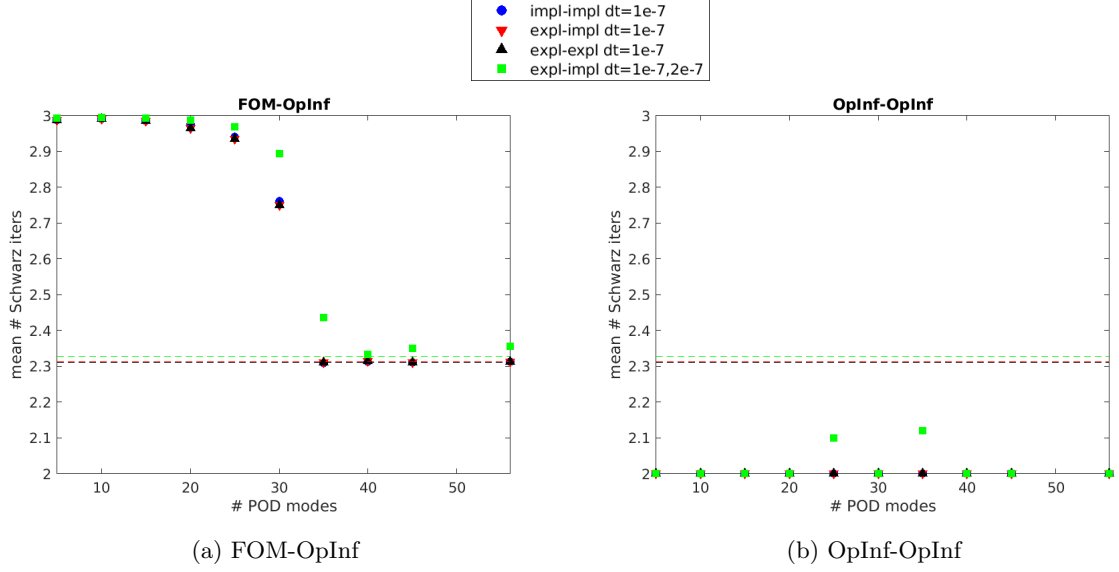
Figure 12: 1D linear elastic wave propagation problem, Symmetric Gaussian initial condition, predictive regime: mean number of Schwarz iterations required to reach convergence for various FOM-OpInf (a) and OpInf-OpInf (b) O-SAM couplings. Dashed horizontal lines show the number of Schwarz iterations needed to reach convergence for FOM-FOM O-SAM couplings with the colors designated in the legend.
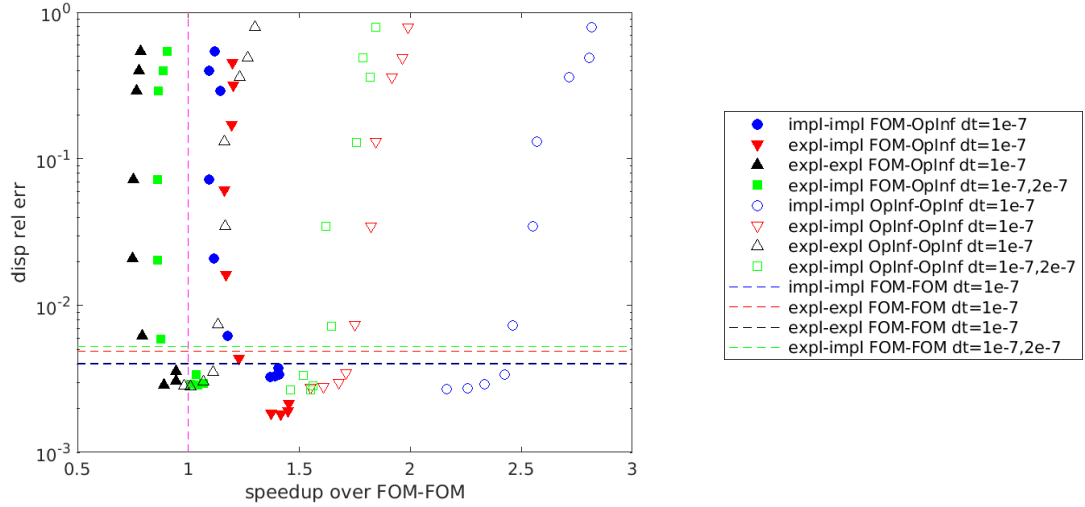

Figure 13: 1D linear elastic wave propagation problem, Symmetric Gaussian initial condition, predictive regime: Pareto plot showing the speed-up over an analogous FOM-FOM coupling vs. displacement relative errors for various FOM-OpInf (filled symbols) and OpInf-OpInf (unfilled symbols) couplings. Dashed horizontal lines indicate relative errors for corresponding FOM-FOM O-SAM couplings. Dashed magenta vertical line indicates a speedup of 1.

average (over all integration points) von Mises stress (σ_{vm}) [5] in Ω_b and Ω_p as a function of r_b. The reader can observe convergence with respect to the basis size, although this convergence plateaus for $r_b > 20$. The accuracy and convergence of the von Mises stress follows closely that of the velocity, most likely due to the highly dynamic nature of the bolted joint problem. For all three fields of interest, it is possible to achieve relative errors as low as $\mathcal{O}(0.1\%)$. This is well below the typical acceptable relative error for most engineering analyses, which generally falls in the range of 1-5%. It is interesting to remark that the displacement solution in the ROM subdomain, Ω_p, is more accurate for all values of r_p than the solution in the FOM solution in Ω_b. This suggests that the solution in Ω_p is easier to represent than the solution in Ω_b.
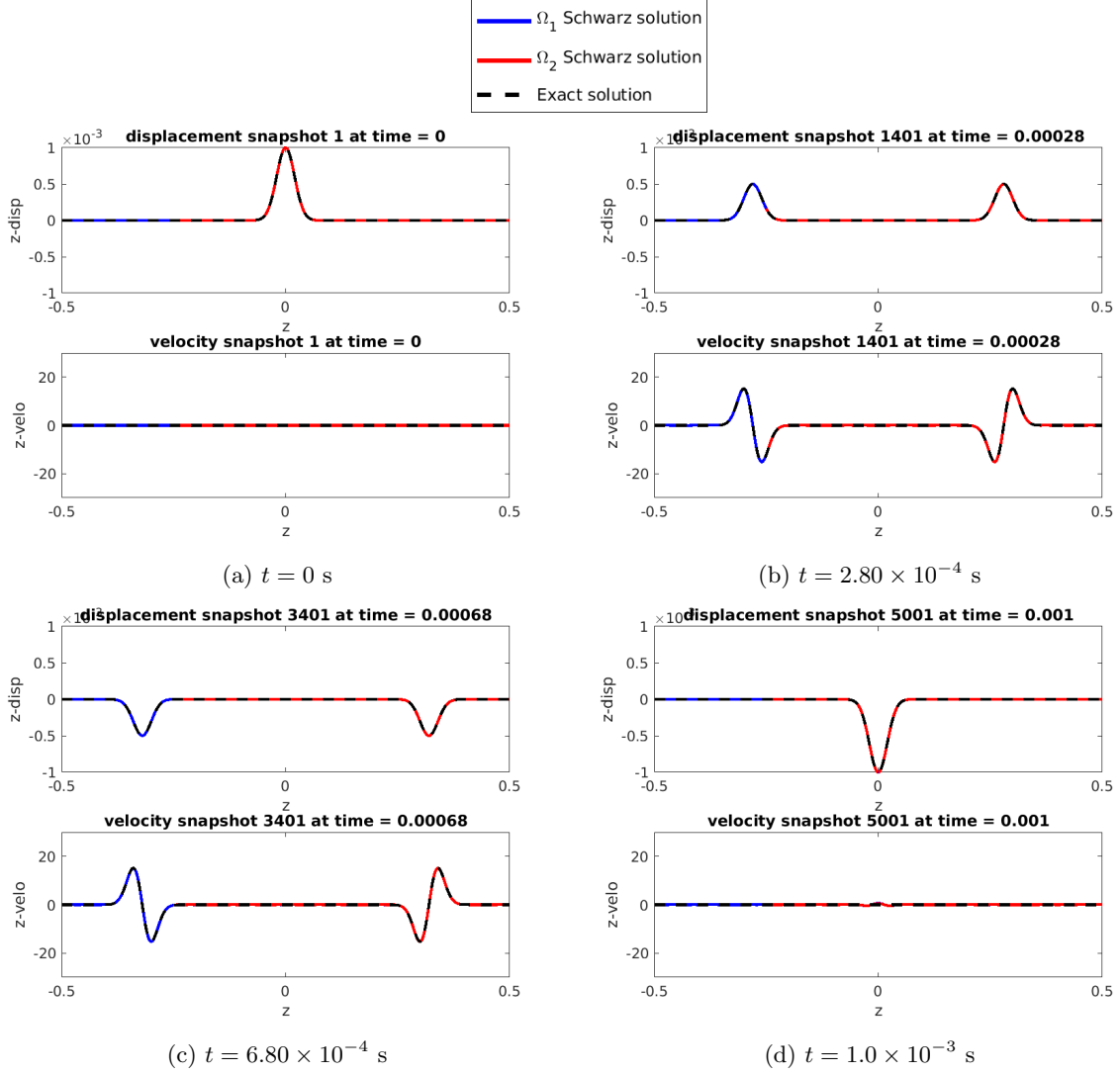
27

Figure 14: 1D linear elastic wave propagation problem, Symmetric Gaussian initial condition, predictive regime: displacement and velocity computed using our coupled OpInf-OpInf models with $r_1 = r_2 = 35$ and an explicit-implicit scheme with $\Delta t_1 = 1.0 \times 10^{-7}$ s and $\Delta t_2 = 2.0 \times 10^{-7}$ s (blue, red), compared with the exact analytical solution (black). Solutions in Ω_1 are shown in blue, whereas solutions in Ω_2 are shown in red.

Next, we compare some performance metrics for our FOM-COpInf couplings, namely the CPU times and the mean/max numbers of Schwarz iterations required for convergence for the FOM-COpInf coupled models compared to the FOM-FOM coupled models. These data are reported in Table 3. The reader can observe that the FOM-COpInf couplings require fewer Schwarz iterations to converge than the analogous FOM-FOM coupled model, as seen earlier in the context of the 1D linear elastic wave propagation problem (Section 5.1). While the mean number of Schwarz iterations does decrease as the basis size r_b is increased, this decrease is not strictly monotonic. For the FOM-COpInf couplings considered, speedups of up to $1.80\times$ are observed with respect to the corresponding FOM-FOM coupling. Curiously, employing a larger COpInf ROM in Ω_p does not necessarily lead to a larger CPU time. This is because the total CPU time is dominated by the number of Schwarz iterations and increasing the ROM basis size can actually decrease the number of Schwarz iterations required for convergence (Table 3). Finally, it is interesting to observe that O-SAM in general converges in a very small number of iterations (between 2-5) despite the overlap region being quite small (see Figure 15(c)). This SAM convergence behavior is similar to what we observed earlier on a similar problem solved within the production `SIERRA/SM` code base [28]; the interested reader is referred to [38] for more details.

(a) Full geometry

(b) Ω_p

(c) Ω_b

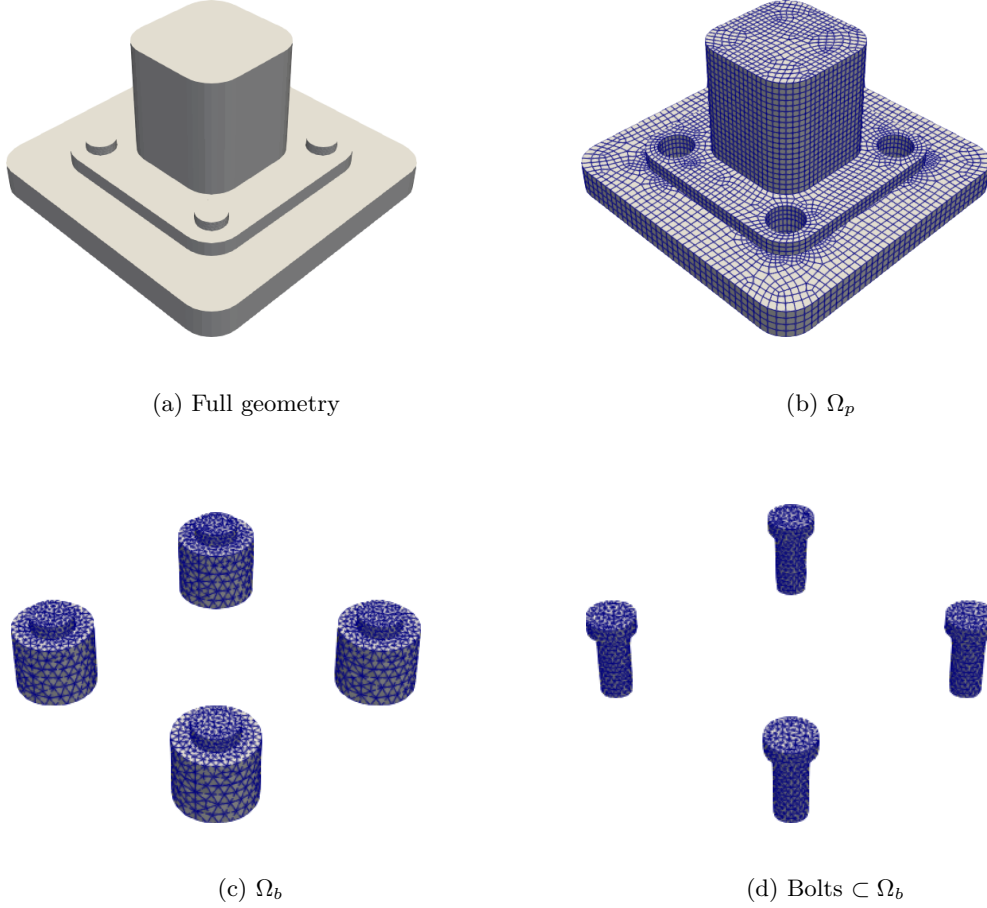(d) Bolts $\subset \Omega_b$

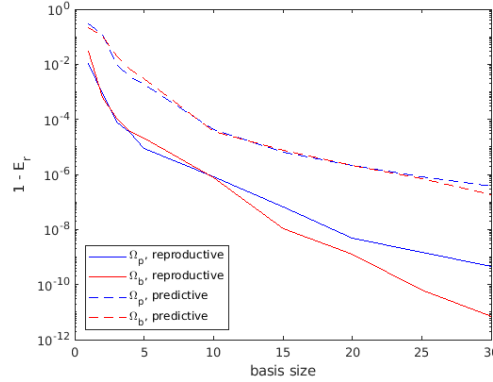Figure 15: 3D nonlinear hyperelastic bolted joint problem: geometry and meshes.

Figure 16: 3D nonlinear hyperelastic bolted joint problem: POD singular value decay $(1 - E_r)$ in Ω_p and Ω_b.

After assessing some FOM-COpInf couplings for the bolted joint problem, we now turn our attention to COpInf-COpInf couplings, in which a pre-trained COpInf model in Ω_p is coupled to a pre-trained COpInf model in Ω_b. We expect these couplings to give less accurate solutions, especially in the small-scale domain, Ω_b, which contains fine scale features that a data-driven model may be incapable of representing. Figure 17(b) shows relative errors in the displacement magnitude, velocity magnitude and the average von Mises stress in each of the two subdomains for COpInf-COpInf couplings having the same number of POD modes in each subdomain, denoted by $r = r_b = r_p$, where r_p is the number of modes retained in Ω_p. The value of r is varied between 5 and 30, as before. For the COpInf-COpInf case, convergence with basis size is no longer observed, as the smaller ROMs for which $r < 20$ are the most accurate (Figure 17(b)). The von Mises stresses have a higher relative error of $\mathcal{O}(10\%)$. The reader can observe from Table
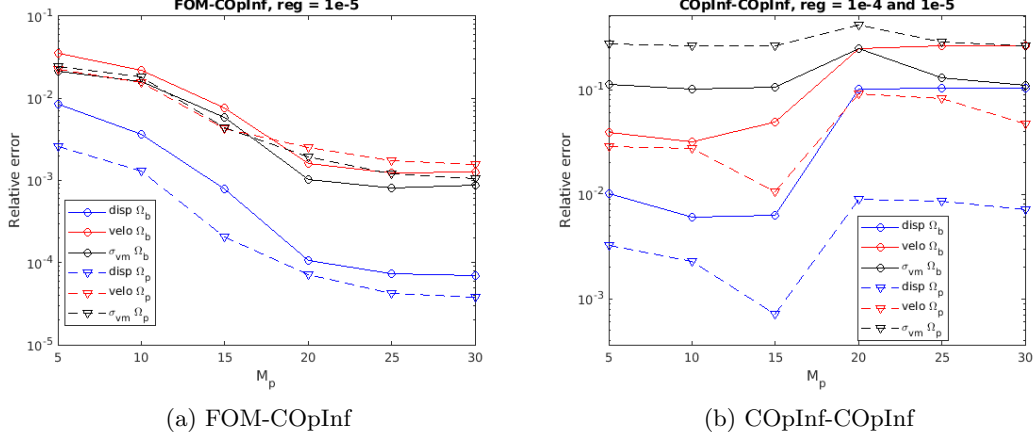
29

(a) FOM-COpInf (b) COpInf-COpInf

Figure 17: 3D nonlinear hyperelastic bolted joint problem, reproductive regime: relative errors in displacement magnitude, velocity magnitude and average von Mises stress (σ_{vm}) for various FOM-COpInf and COpInf-COpInf couplings as a function of the reduced basis size.

3 that greater CPU savings of up to $9.84\times$ are possible when utilizing COpInf-COpInf coupled models. Like for the FOM-COpInf couplings, fewer Schwarz iterations are needed to reach convergence for the COpInf-COpInf coupled models than for the FOM-FOM coupled model. Also, as before, employing a larger COpInf ROM does necessarily lead a a larger CPU time.

We conclude our discussion of the reproductive results by noting that it may be possible to obtain more accurate COpInf ROMs by further fine-tuning the regularization parameters used in the COpInf optimization problem. It is well-known that, for quadratic and cubic OpInf ROMs [33, 45], it is best to employ different regularization parameters for different operators during the operator learning problem (see Remark 4). Doing this is beyond the scope of the present work, which is focused on demonstrating SAM's ability to couple together a variety of different subdomain-local models including OpInf ROMs, rather than optimizing the individual OpInf models being coupled.

Table 3: 3D nonlinear hyperelastic bolted joint problem, reproductive regime: performance metrics for various O-SAM-based couplings. The FOM-COpInf and COpInf-COpInf couplings requiring the lowest CPU time and the smallest number of Schwarz iterations are highlighted in green.

	r_b	r_p	λ_b	λ_p	CPU time	Mean/max # Schwarz iterations
FOM-FOM	—	—	—	—	52m 49.5s	3.21/5
FOM-COpInf	—	5	—	1.0×10^{-5}	32m 58.1s	2.62/3
	—	10	—	1.0×10^{-5}	31m 24.6s	2.48/3
	—	15	—	1.0×10^{-5}	31m 15.8s	2.45/3
	—	20	—	1.0×10^{-5}	31m 11.3s	2.47/3
	—	25	—	1.0×10^{-5}	30m 29.3s	2.40/3
	—	30	—	1.0×10^{-5}	29m 21.9s	2.29/3
COpInf-COpInf	5	5	1.0×10^{-5}	1.0×10^{-5}	7m 19.0s	2.24/3
	10	10	1.0×10^{-5}	1.0×10^{-5}	5m 21.9s	2.22/3
	15	15	1.0×10^{-4}	1.0×10^{-4}	5m 40.1s	2.05/3
	20	20	1.0×10^{-5}	1.0×10^{-5}	6m 41.8s	2.03/3
	25	25	1.0×10^{-5}	1.0×10^{-5}	7m 18.1s	2.05/3
	30	30	1.0×10^{-5}	1.0×10^{-5}	6m 14.7s	2.05/3

5.2.2 Predictive variant of the bolted joint problem

Having performed an initial assessment/verification on a reproductive variant of the 3D nonlinear hyperelastic bolted joint problem, we now evaluate our coupled hybrid models in the predictive regime. To do so, consider the same problem formulation as above but with a more general time-dependent

boundary condition applied on the top boundary of Ω_p having the form

$$\boldsymbol{u}(\boldsymbol{x}, t) = \big(\ a_1[1 - \cos(500\pi t)], \quad a_2[1 - \cos(500\pi t)], \quad a_3[1 - \cos(500\pi t)] \ \big)^{\mathsf{T}}, \tag{33}$$

for $a_1, a_2, a_3 \in \mathbb{R}$. In the predictive version of the bolted joint problem, we collect training data by solving the problem from time 0 to time 5.8×10^{-5} for three sets of parameters: $(a_1, a_2, a_3) = (2, 0, 0)$, $(a_1, a_2, a_3) = (0, 0, 2)$ and $(a_1, a_2, a_3) = (2, 2, 0)$. This generates a total of 176 snapshots, from which we construct different size OpInf ROMs in Ω_b and Ω_p. After coupling these subdomain-local OpInf ROMs to each other and to subdomain-local FOMs, we predict the solution to the problem with $(a_1, a_2, a_3) = (2, 0, 2)$, as before. Effectively, we are generating training data by pushing the joint geometry in the positive x direction, the positive z direction, and at a 45 degree angle in the $x - y$ direction; we then predict the solution to the problem when the joint is pushed at a 45 degree angle in the $x - z$ direction. We consider in our study reduced bases consisting of between 10 and 24 POD modes. For the predictive bolted joint problem, $r_b = r_p = 15$ modes are needed to capture 99.999% of the snapshot energy in Ω_b and Ω_p, whereas $r_b = r_p = 24$ modes are needed to capture 99.9999% of the snapshot energy in Ω_b and Ω_p. On the Schwarz boundary of Ω_b, 23 POD modes capture 99.9999% of the snapshot energy, compared to 24 POD modes for the Schwarz boundary of Ω_p. As for the reproductive version of this problem, a single POD mode captures 99.9999% of the snapshot energy for the Dirichlet boundaries.

We report the relative errors in the displacement magnitude, velocity magnitude and von Mises stress solutions in each of the subdomains as a function of r_b and r_p; see Tables 4 and 5 for FOM-COpInf and COpInf-COpInf results, respectively. The "optimal" values of the regularization parameters in Ω_b and Ω_p, identified by our estimation algorithm (Section 4.2.4) were again 10^{-4} or 10^{-5}. For the FOM-COpInf couplings (Table 4), relative errors of $\mathcal{O}(0.1)$–$\mathcal{O}(1\%)$ and $\mathcal{O}(1\%)$ can be achieved for the displacement and velocity fields, respectively, and relative errors of 7–8% are possible for the von Mises stress σ_{vm}. The accuracy of the COpInf-COpInf coupled models is worse and degrades with basis refinement, though errors of 4.41%, 10.7% and 12.4% are achievable for the displacement, velocity and σ_{vm} fields in the bolts subdomain Ω_b with COpInf ROMs based on 10 POD modes. Figures 18 and 19 show the displacement magnitude in the full domain, and σ_{vm} in the bolts, respectively, for various O-SAM-based couplings. For the COpInf-COpInf couplings, we show the best and worst cases in subfigures (c) and (d), respectively. These correspond go $r = r_b = r_p = 10$ and $r = r_b = r_p = 20$, respectively. It can be seen that the large-scale deformation of the bolted joint is correct for all the models being assessed (Figure 18). Figure 19 shows that the maximum stress occurs in the rear fastener in the negative $x - z$ coordinate plane, parallel to the direction in which the forcing is applied, as expected. The COpInf-FOM σ_{vm} in the bolts (Figure 19(b)) matches the FOM-FOM σ_{vm} (Figure 19(a)) remarkably well. For the best COpInf-COpInf model with $r = r_b = r_p = 10$ (Figure 19(c)), the reader can observe that σ_{vm} is slightly overestimated in all of the bolts. The overestimation of σ_{vm} is more profound for the worst COpInf-COpInf models, having $r = r_b = r_p = 20$ (Figure 19(d)). While the error in σ_{vm} in the Ω_b subdomain for this coupling is close to 31%, which suggests that the magnitude of σ_{vm} is substantially off, it is important to remark that the locations of maximum stress are still correctly predicted by the model, as can be seen by examining Figure 19(d). Hence, even the less accurate COpInf-COpInf coupled models can be useful, as they can provide a conservative estimate of the bolt failure.

Also reported in Tables 4 and 5 are CPU times for the various couplings. A modest speed-up of $1.65\times$ is achieved for the FOM-COpInf couplings, whereas a much greater speedup of $6.12\times$ is observed for the most accurate COpInf-COpInf coupling. As for the reproductive version of this problem, we observe that O-SAM converges in fewer Schwarz iterations for the FOM-COpInf and COpInf-COpInf couplings than for the FOM-FOM coupling, which contributes to the observed CPU-time reduction. It is likely that bigger cost reductions are possible when employing the cheaper QOpInf models, but a study involving QOpInf-FOM and QOpInf-QOpInf couplings for the bolted joint problem is beyond the scope of the present work.

5.3 3D nonlinear hyperelastic torsion problem

The next test case considers a nonlinear hyperelastic bar subjected to finite deformation by a high degree of torsion. Our bar geometry has dimensions 0.05 m $\times$ 0.05 m $\times$ 1.0 m, and is forced using the following initial conditions on the displacement and velocity, respectively:

$$\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{0}, \qquad \dot{\boldsymbol{u}}(\boldsymbol{x}, 0) = \big(\ -b_1 yz, \quad b_2 xz, \quad 0 \ \big)^{\mathsf{T}}. \tag{34}$$

In (34), $b_1, b_2 \in \mathbb{R}$ are the rotation rates. The boundary conditions are all homogeneous Neumann. We specify a nonlinear Neohookean-type material model with Young's modulus $E = 1.0 \times 10^9$ Pa, Poisson's

Table 4: 3D nonlinear hyperelastic bolted joint problem, predictive regime: relative errors and performance metrics for various FOM-COpInf O-SAM-based couplings. The best coupling in terms of a combination of the overall accuracy and efficiency is highlighted in green.

	Field	FOM-FOM	FOM-COpInf			
r_b	$-$	$-$	10	15	20	24
λ_b	$-$	$-$	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-4}
Ω_b rel errors	$\boldsymbol{u}$	$-$	4.51×10^{-2}	3.94×10^{-2}	3.75×10^{-2}	3.51×10^{-2}
	$\dot{\boldsymbol{u}}$	$-$	9.40×10^{-2}	5.61×10^{-2}	4.79×10^{-2}	4.25×10^{-2}
	σ_{vm}	$-$	9.29×10^{-2}	8.40×10^{-2}	8.14×10^{-2}	7.74×10^{-2}
Ω_p rel errors	$\boldsymbol{u}$	$-$	9.71×10^{-3}	7.26×10^{-3}	7.49×10^{-3}	6.92×10^{-3}
	$\dot{\boldsymbol{u}}$	$-$	3.97×10^{-2}	3.89×10^{-2}	2.32×10^{-2}	1.94×10^{-2}
	σ_{vm}	$-$	9.88×10^{-2}	8.48×10^{-2}	7.28×10^{-2}	7.17×10^{-2}
CPU time	$-$	52m 49.5s	33m 2.8s	35m 41.5s	31m 57.6s	32m 46.7s
Mean/max # Schwarz iterations	$-$	3.21/5	2.43/3	2.36/3	2.34/3	2.28/3

Table 5: 3D nonlinear hyperelastic bolted joint problem, predictive regime: relative errors and performance metrics for various COpInf-COpInf O-SAM-based couplings. The best coupling in terms of a combination of accuracy of the solution in Ω_b and efficiency is highlighted in green.

	Field	FOM-FOM	COpInf-COpInf			
$r = r_b = r_p$	$-$	$-$	10	15	20	24
λ_b	$-$	$-$	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-5}
λ_p	$-$	$-$	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-4}
Ω_b rel errors	$\boldsymbol{u}$	$-$	4.41×10^{-2}	9.48×10^{-2}	1.29×10^{-1}	4.51×10^{-2}
	$\dot{\boldsymbol{u}}$	$-$	1.07×10^{-1}	3.60×10^{-1}	5.27×10^{-1}	2.39×10^{-1}
	σ_{vm}	$-$	1.24×10^{-1}	2.38×10^{-1}	5.29×10^{-1}	1.72×10^{-1}
Ω_p rel errors	$\boldsymbol{u}$	$-$	9.78×10^{-3}	1.62×10^{-2}	1.25×10^{-2}	7.26×10^{-3}
	$\dot{\boldsymbol{u}}$	$-$	4.36×10^{-2}	9.35×10^{-2}	8.57×10^{-2}	3.89×10^{-2}
	σ_{vm}	$-$	3.02×10^{-1}	5.04×10^{-1}	5.96×10^{-1}	3.09×10^{-1}
CPU time	$-$	52m 49.5s	8m 37.7s	7m 41.2s	8m 48.4s	7m 35.1s
Mean/max # Schwarz iterations	$-$	3.21/5	2.43/3	2.47/3	2.36/3	2.29/3

ratio $\nu = 0.25$ and density $\rho = 1000$ kg/m^3. While these material properties are not realistic, they enable us to run this problem with an explicit time-stepper without having to resort to an extremely small time step to achieve CFL stability. For more details on the material model employed, the reader is referred to [39] and Appendix A.3. The problem is run from time 0 until time 2.0×10^{-3} s. We intentionally choose not to run this problem with a linear elastic material model, as the solution to such a problem is known to be nonphysical.

The main objective of the torsion problem is to demonstrate that O-SAM is capable of coupling two regions of the 3D bar having different mesh resolutions, element types, time integration schemes, time-steps and models. Toward this effect, we break up our geometry into two subdomains, $\Omega_1 = (-0.025, 0.025)^2 \times (-0.5, 0.25)$ and $\Omega_2 = (-0.025, 0.025)^2 \times (-0.25, 0.5)$, and discretize them with a fine four-node tetrahedral and coarse eight-node hexahedral mesh, as shown in Figure 20. The Ω_1 mesh consists of 7333 elements and 1787 nodes (Figure 20(a)), whereas the Ω_2 mesh has only 120 elements and 279 nodes (Figure 20(b)). Since the material model specified within the bar is nonlinear, a linear OpInf ROM will be incapable of capturing the problem dynamics. Although the Neohookean-type material model utilized gives rise to a set of PDEs with generic nonlinearities, we choose to approximate these nonlinearities with a quadratic OpInf model, denoted by QOpInf. We focus our attention on QOpInf-FOM couplings, in which a QOpInf ROM is prescribed in Ω_1 and a FOM is prescribed in Ω_2. Since the solution in Ω_1 is approximated by a relatively inexpensive ROM, we will advance this subdomain forward in time using an implicit Newmark-β stepper with parameters $\beta = 0.25$ and $\gamma = 0.5$ and time-step $\Delta t = 2.0 \times 10^{-6}$ s. For the FOM subdomain, Ω_2, we utilize an explicit Newmark-β scheme with $\gamma = 0.5$ and time-step $\Delta t = 1.0 \times 10^{-6}$ s. Within our SAM coupling algorithm, we specify a controller time-step of 2.0×10^{-6} s. We converge SAM to a relative tolerance of $\delta_{\text{rel}} = 1.0 \times 10^{-10}$ and an absolute tolerance

(a) FOM-FOM

(b) FOM-COpinf ($r_b = 24$)

(c) COpInf-COpInf ($r = r_b = r_p = 10$)

(d) COpInf-COpInf ($r = r_b = r_p = 20$)

displacement magnitude

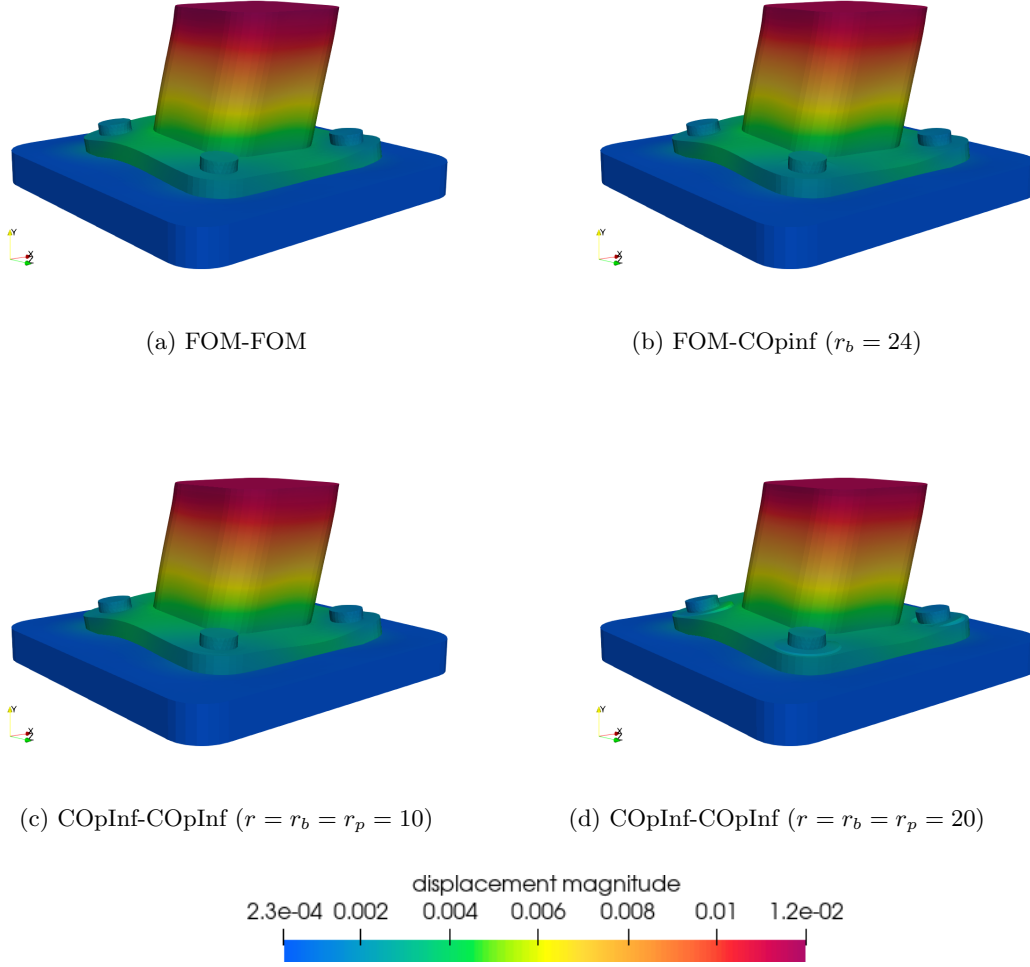2.3e-04 0.002   0.004   0.006   0.008   0.01   1.2e-02

Figure 18: 3D nonlinear hyperelastic bolted joint problem, predictive regime: plots of the displacement magnitude for a FOM-FOM O-SAM-based coupling (a) compared to a COpInf-FOM Schwarz coupling with $r_b = 24$ (b), a COpInf-COpInf Schwarz coupling with $r = r_b = r_p = 10$ (c), and a COpInf-COpInf Schwarz coupling with $r = r_b = r_p = 20$ (d).

of $\delta_{\mathrm{abs}} = 1.0 \times 10^{-6}$. For the implicit Newmark$-\beta$ runs, we employ the same relative and absolute tolerances for the Newton-based nonlinear solver and iterative linear solver.

As with the other test cases, we consider both a reproductive and a predictive variant of the torsion problem. For both problem variants, we wish to predict the solution for $b_1 = b_2 = 5500$ in (34). For the reproductive version, an O-SAM-based explicit-implicit FOM-FOM coupling is performed using these parameter values with snapshots saved every 2.0×10^{-5} s, yielding a total of 201 snapshots, from which a POD basis and QOpInf ROM is constructed. For the predictive version of the problem, the parameters b_1, b_2 are each sampled from the parameter set $\{500, 1000, 5000, 8000\}$, and a total of 16 O-SAM-based implicit FOM-FOM coupling runs are performed. As before, snapshots are saved every 2.0×10^{-5} s, this time yielding a total of 3216 snapshots, from which a QOpInf ROM is constructed in Ω_1. A singular value-based energy analysis reveals that $r_1 = 30$ and $r_1 = 27$ POD modes capture 99.999% of the snapshot energy in Ω_1 for the reproductive and predictive versions of the torsion problem, respectively; meanwhile, 3 and 6 POD modes capture 99.9999% of the snapshot energy on the Schwarz boundary for the reproductive and predictive versions of this problem, respectively. Our parameter sweep algorithm determined optimal regularization parameters of 1.0×10^{-3} and 1.0×10^{-11} respectively for these cases (see Section 4.2.4).

Table 6 reports the displacement and velocity magnitude errors in both subdomains for QOpInf-FOM
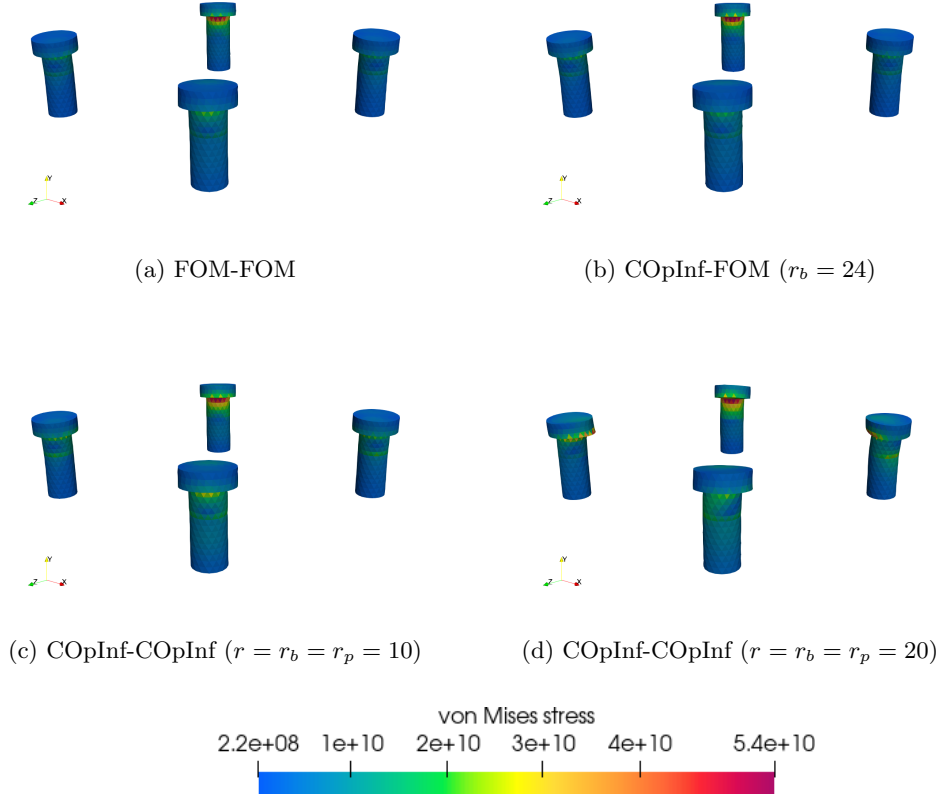
(a) FOM-FOM
(b) COpInf-FOM ($r_b = 24$)

(c) COpInf-COpInf ($r = r_b = r_p = 10$)
(d) COpInf-COpInf ($r = r_b = r_p = 20$)

von Mises stress

2.2e+08 1e+10 2e+10 3e+10 4e+10 5.4e+10

Figure 19: 3D nonlinear hyperelastic bolted joint problem, predictive regime: plots of the average von Mises stresses σ_{vm} for a FOM-FOM O-SAM-based coupling (a) compared to a COpInf-FOM Schwarz coupling with $r_b = 24$ (b), a COpInf-COpInf Schwarz coupling with $r = r_b = r_p = 10$ (c), and a COpInf-COpInf Schwarz coupling with $r = r_b = r_p = 20$ (d).

(a) Ω_1

(b) Ω_2
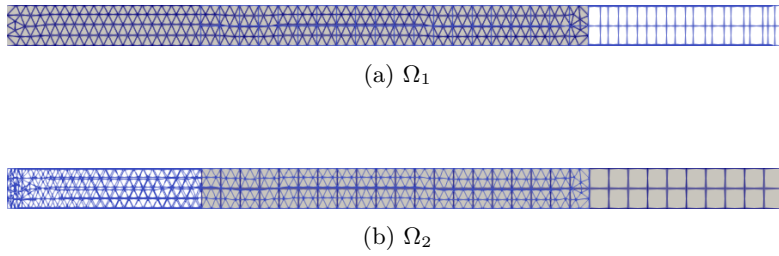
Figure 20: 3D nonlinear hyperelastic torsion problem: domain decomposition and meshes. Ω_1 and Ω_2 and their meshes are shown in gray in subfigures (a) and (b), respectively.

couplings in which the QOpInf ROM has either 30 or 27 POD modes, along with some performance metrics. The reader can observe that the QOpInf-FOM coupled models deliver accurate displacement solutions in both the reproductive and predictive regime, with maximum displacement magnitude errors of 0.267% and 4.32%, respectively, and maximum velocity magnitude errors of 3.56% and 14.9%, respectively As expected, the solutions are more accurate in the FOM subdomain, Ω_2, than in the ROM subdomain, Ω_1. Figures 21–22 show plots of the reproductive and predictive QOpInf-ROM solutions compared to their FOM-FOM analogs for the displacement and velocity fields at the final simulation time, 2.0×10^{-3} s. The coupled QOpInf-FOM models are capable of capturing the displacement solution remarkably well.

Finally, it is interesting to observe that the QOpInf-ROM couplings summarized in Table 6 require

fewer Schwarz iterations to achieve convergence than their FOM-FOM analogs, as previously observed for our other test cases. The reduced size of the QOpInf ROMs combined with this reduction in Schwarz iterations enables the coupled models to achieve speedups of up to 23.5×.

Table 6: 3D nonlinear hyperelastic torsion problem: performance metrics for various O-SAM-based couplings.

	Field	FOM-FOM	QOpInf-FOM reproductive	QOpInf-FOM predictive
r_1	−	−	30	27
λ_1	−	−	1.0×10^{-3}	1.0×10^{-1}
Ω_1 rel errors	$\boldsymbol{u}$	−	2.67×10^{-3}	4.32×10^{-2}
	$\dot{\boldsymbol{u}}$	−	3.56×10^{-2}	1.49×10^{-1}
Ω_2 rel errors	$\boldsymbol{u}$	−	1.13×10^{-3}	2.44×10^{-2}
	$\dot{\boldsymbol{u}}$	−	1.12×10^{-2}	9.52×10^{-2}
CPU time	−	39m 11.8s	1m 40.2s	1m 39.5s
Mean/max # Schwarz iterations	−	3.0/3	2.0/2	2.0/2

(a) FOM-FOM   (b) FOM-FOM

(c) OpInf-FOM (reproductive)   (d) OpInf-FOM (reproductive)

(e) OpInf-FOM (predictive)   (f) OpInf-FOM (predictive)

displacement magnitude
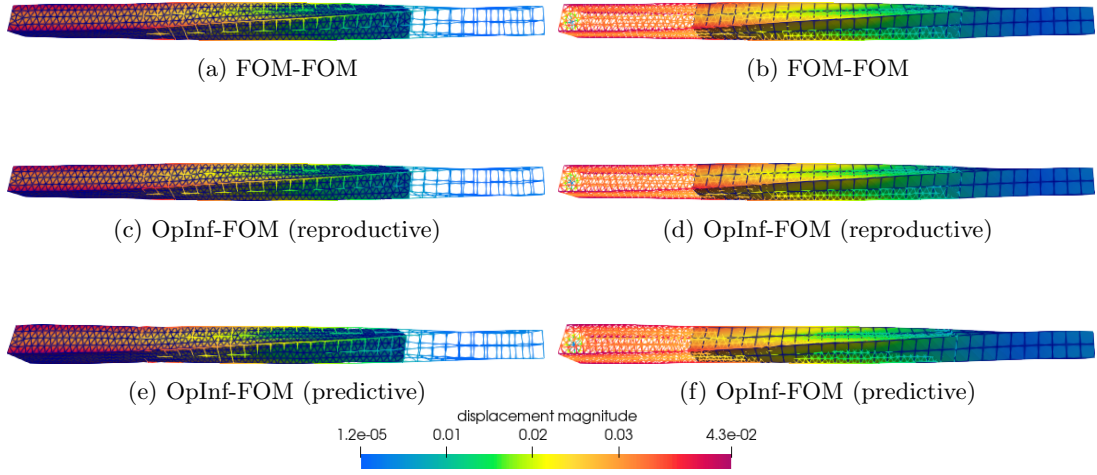1.2e-05   0.01   0.02   0.03   4.3e-02

Figure 21: 3D nonlinear hyperelastic torsion problem: computed displacement solution magnitudes in Ω_1 ((a), (c), (e)) and Ω_2 ((b), (d), (f)) at the final time 2.0×10^{-3} s for various O-SAM-based couplings.

5.4 3D nonlinear hyperelastic tension specimen problem

The final test case on which we evaluate our coupling method is the tension specimen problem. Consider a uniaxial cylindrical tensile specimen made of aluminum. In order to minimize computational cost, we perform a simulation of $\frac{1}{8}$ of the full tension specimen geometry (Figure 23(a)) and apply symmetry boundary conditions to obtain a solution consistent with a simulation on the full geometry. The height of the geometry on which the simulation is performed (Figure 23(a)) is 4.445 cm. The smaller gauge radius is 3.81 m whereas the larger grip radius is 6.35 mm. We decompose the geometry into two subdomains, one containing the gauge, denoted by Ω_1, and one containing the grip, denoted by Ω_2. The heights of Ω_1 and Ω_2 are 3.175 cm and 2.54 cm, respectively. In an effort to further assess O-SAM's ability to couple regions having different meshes and element types, we discretize Ω_1 with 17,496 ten-node tetrahedral elements, while discretizing Ω_2 with a coarser mesh having 10,800 eight-node hexahedral elements. The meshes contain 20,008 and 12,444 nodes, respectively. As for the torsion problem, we specify a nonlinear Neohookean-type material model, this time with Young's modulus $E = 70 \times 10^9$ Pa, Poisson's ratio $\nu = 0.36$ and density $\rho = 2700$ kg/m^3. Assuming our coordinate system is oriented such as "up" is in the positive y–dimension and that the lower left corner of the geometry (Figure 23(a)) is at the origin $(0, 0, 0)$, symmetry boundary conditions are specified for boundaries having $x = 0$, $z = 0$ and $y = 0$, which amount to setting homogeneous Dirichlet boundary conditions for the x, z and y components of the displacement
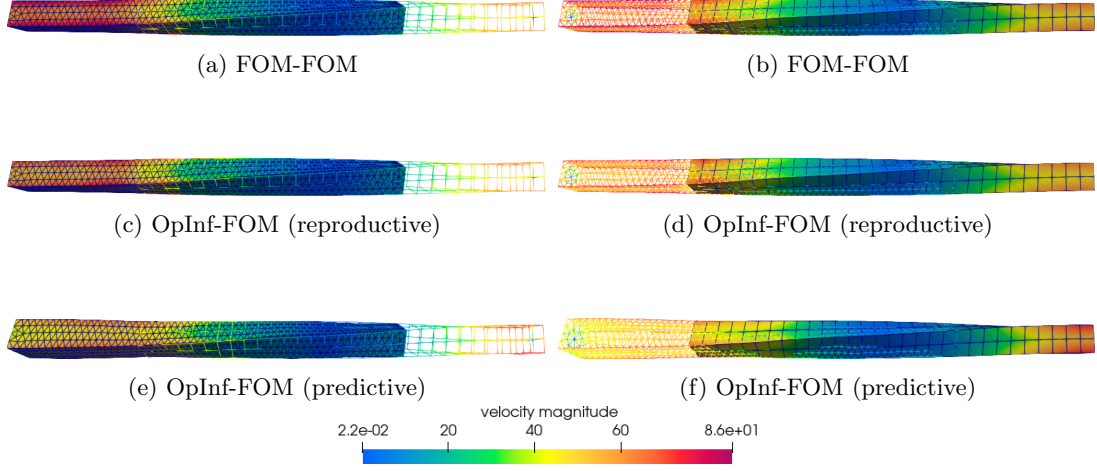
Figure 22: 3D nonlinear hyperelastic torsion problem: computed velocity solution magnitudes in Ω_1 ((a), (c), (e)) and Ω_2 ((b), (d), (f)) at the final time 2.0×10^{-3} s for various O-SAM-based couplings.

on these boundaries, respectively. The problem is forced by setting the following dynamic boundary condition at the positive y boundary:

$$\boldsymbol{u}(\boldsymbol{x}, t) = \begin{pmatrix} 0, & \frac{1}{2}\alpha(1 - \cos(\pi t)), & 0 \end{pmatrix}^{\mathsf{T}}, \tag{35}$$

for a specified scalar parameter $\alpha > 0$. The problem is initialized with zero initial conditions for the displacement and the velocity, and advanced in time using an implicit Newmark-β time-integrator having $\beta = 0.25$ and $\gamma = 0.5$ until time 1.0 s. In applying O-SAM, we specify relative and absolute Schwarz tolerances of $\delta_{\mathrm{rel}} = 1.0 \times 10^{-8}$ and $\delta_{\mathrm{abs}} = 1.0 \times 10^{-6}$, respectively. To prevent the Schwarz algorithm from taking too much CPU time, we cap the number of Schwarz iterations as 32. For the Newton-based nonlinear and iterative linear solvers, we employ relative tolerances of 1.0×10^{-7} and 1.0×10^{-5}, respectively. The problem is initialized with a $\mathbf{0}$ initial condition for both the displacement and velocity fields.

The objective herein is to construct FOM-QOpinf and QOpInf-QOpInf coupled models using O-SAM, and to evaluate them in the reproductive as well as the predictive regime. Our goal is to capture the solution corresponding to the initial condition (35) with $\alpha = 0.005$. For our FOM-QOpInf coupling, we assign the QOpInf ROM to the subdomain containing the holder, Ω_2, where less complex dynamics are expected. For the reproductive version of the problem, training data are generated by simulating the problem using a FOM-FOM O-SAM coupling with $\alpha = 0.005$ and a time-step of 1.0×10^{-2} s in both subdomains. For the predictive variant, training data are generated by performing FOM-FOM O-SAM couplings with $\alpha \in \{0.003, 0.006\}$. We used the resulting snapshot sets to build POD bases in Ω_1 and Ω_2, each capturing 99.9999% of the snapshot energy, which required a mere 2 modes in the subdomain interiors, 2 modes on the Schwarz boundary and 1 mode on the symmetry boundaries for both the reproductive and predictive cases. We utilized our regularization parameter optimization algorithms to determine the optimal values of λ_1 and λ_2 in the two subdomains; these are reported in Tables 7 and 8.

The main results of our analysis are reported in Tables 7 and 8 for the reproductive and predictive cases, respectively. The tables give relative errors in the displacement, velocity and von Mises stress fields, as well as the CPU times required to run the cases on `Rigel`, and and mean and maximum number of Schwarz iterations required to reach convergence. In addition to the classical iterative Schwarz algorithm advocated in this paper, termed "full SAM" in Tables 7 and 8, in which the Schwarz iteration is converged up to the specified δ_{rel} and δ_{abs} tolerances, we consider also a simplified version of the Schwarz algorithm, in which a single Schwarz iteration is performed in each time step. This second variant of O-SAM is roughly equivalent to the approach proposed in [16] by Farcas *et al.*, and is labeled "1-iter SAM" in the tables and figures that follow.

The reader can observe by examining Tables 7 and 8 that all FOM-QOpInf and QOpInf-QOpInf models evaluated deliver relative errors of $\mathcal{O}(10^{-4}) - \mathcal{O}(10^{-3})$ for the displacement and von Mises stress, and errors of $\mathcal{O}(10^{-3}) - \mathcal{O}(10^{-2})$ for the velocity when applying the full Schwarz algorithm. When applying the 1-iteration version of SAM, the errors are several orders of magnitude higher, indicating convergence to the wrong solution, a result corroborated by Figures 24(c) and 25(c), which show the displacement and
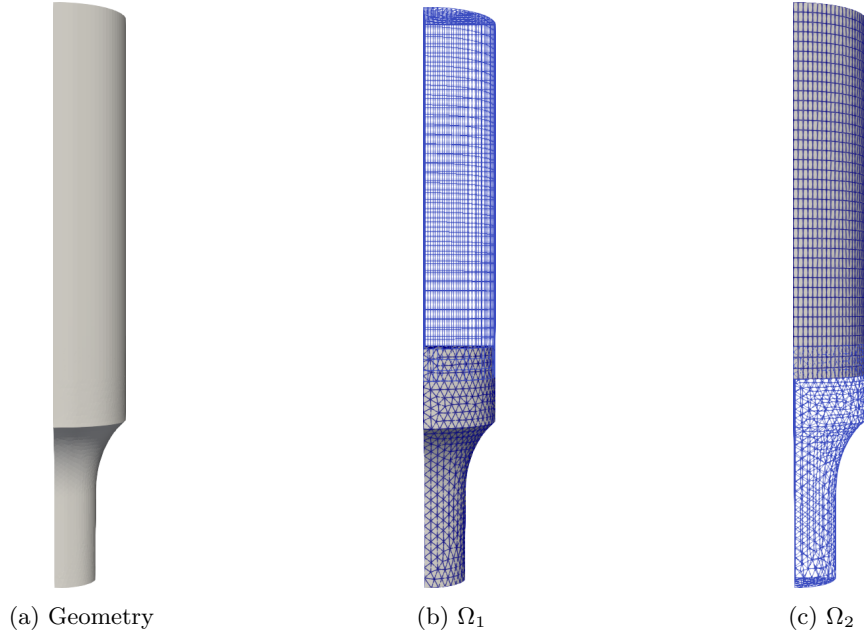
36

(a) Geometry   (b) Ω_1 (c) Ω_2

Figure 23: 3D nonlinear hyperelastic tension-specimen problem: geometry (a) and domain decomposition into Ω_1, discretized with a ten-node tetrahedral mesh (b), and Ω_1, discretized using an eight-node hexahedral mesh (c).

von Mises stress solutions at the final time 1.0 s for several such couplings. It is interesting to observe that the full Schwarz method requires significantly fewer Schwarz iterations to converge when applied to our FOM-QOpInf and QOpInf-QOpInf couplings than the corresponding FOM-FOM coupling: whereas the FOM-FOM coupling fails to achieve the specified Schwarz tolerances in the maximum number of Schwarz iterations allowed (32), the hybrid couplings involving QOpInf ROMs converge in a mere 7–10 Schwarz iterations. This enables O-SAM to achieve impressive speedups of up to $6.13\times$ and $106\times$ when performing FOM-QOpInf and QOpInf-QOpInf couplings, all while maintaining good accuracy with respect to the corresponding FOM-FOM coupled solution (Tables 7–8 and Figures 24–25(a),(b)). As discussed earlier in Section 5.1, we believe the reduction in the number of Schwarz iterations required for convergence when coupling in ROMs is due to the fact the ROM solutions tend to be smoother than their FOM analogs, making them easier to couple, together with the fact that the ROMs use problem-specific POD bases to represent the solution, instead of generic and problem-agnostic finite element shape functions.

Table 7: 3D neohookean tension specimen problem, reproductive regime: errors and performance metrics for various O-SAM-based couplings. The best coupling in terms of a combination of the overall accuracy and efficiency is highlighted in green. The errors highlighted in red are deemed unacceptable.

	Field	FOM-FOM full SAM	FOM-QOpInf full SAM	FOM-QOpInf 1-iter SAM	QOpInf-QOpInf full SAM	QOpInf-QOpInf 1-iter SAM
r_1	$-$	$-$	$-$	$-$	2	2
r_2	$-$	$-$	2	2	2	2
λ_1	$-$	$-$	$-$	$-$	1.0×10^{-6}	1.0×10^{-6}
λ_2	$-$	$-$	1.0×10^{-7}	1.0×10^{-7}	1.0×10^{-11}	1.0×10^{-11}
Ω_1 rel errors	$\boldsymbol{u}$	$-$	1.01×10^{-4}	1.88×10^{-2}	5.68×10^{-4}	2.46×10^{1}
	$\dot{\boldsymbol{u}}$	$-$	8.23×10^{-3}	3.94×10^{-2}	1.83×10^{-2}	4.49
	σ_{vm}	$-$	1.04×10^{-4}	1.93×10^{-2}	1.20×10^{-3}	3.17
Ω_2 rel errors	$\boldsymbol{u}$	$-$	6.57×10^{-5}	6.87×10^{-3}	2.38×10^{-4}	1.03×10^{-1}
	$\dot{\boldsymbol{u}}$	$-$	8.91×10^{-3}	4.54×10^{-2}	7.60×10^{-3}	1.89
	σ_{vm}	$-$	8.51×10^{-4}	1.22×10^{-1}	8.94×10^{-4}	4.33×10^{-1}
CPU time	$-$	8h 19m 29.5s	1h 34m 53.1s	12m 37.9s	5m 20.3s	1m 43.5
Mean/max # Schwarz iters	$-$	32.0/32	7.71/8	1.0/1	9.04/10	1.0/1

Table 8: 3D neohookean tension specimen problem, predictive regime: errors and performance metrics for various O-SAM-based couplings. The best coupling in terms of a combination of the overall accuracy and efficiency is highlighted in green. The errors highlighted in red are deemed unacceptable.

	Field	FOM-FOM full SAM	FOM-QOpInf full SAM	FOM-QOpInf 1-iter SAM	QOpInf-QOpInf full SAM	QOpInf-QOpInf 1-iter SAM
r_1	−	−	−	−	2	2
r_2	−	−	2	2	2	2
λ_1	−	−	−	−	1.0×10^{-11}	1.0×10^{-11}
λ_2	−	−	1.0×10^{-6}	1.0×10^{-6}	1.0×10^{-6}	1.0×10^{-6}
Ω_1 rel errors	$\boldsymbol{u}$	−	3.44×10^{-4}	5.62×10^{-2}	5.73×10^{-4}	2.65×10^{-2}
	$\dot{\boldsymbol{u}}$	−	1.72×10^{-2}	1.42	1.83×10^{-2}	1.24×10^{-1}
	σ_{vm}	−	3.41×10^{-4}	5.53×10^{-2}	8.53×10^{-4}	2.75×10^{-2}
Ω_2 rel errors	$\boldsymbol{u}$	−	2.50×10^{-4}	4.41×10^{-2}	5.78×10^{-4}	1.92×10^{-2}
	$\dot{\boldsymbol{u}}$	−	1.86×10^{-2}	1.27	1.96×10^{-2}	1.91×10^{-1}
	σ_{vm}	−	2.40×10^{-3}	2.04	6.00×10^{-3}	2.33×10^{-1}
CPU time	−	8h 19m 29.5s	1h 21m 25.9s	12m 43.5s	4m 42.1s	1m 49.3s
Mean/max # Schwarz iters	−	32.0/32	7.03/8	1.0/1	7.74/8	1.0/1

(a) FOM-FOM, full SAM   (b) QOpInf-QOpInf, full SAM   (c) QOpInf-QOpInf, 1-iter SAM
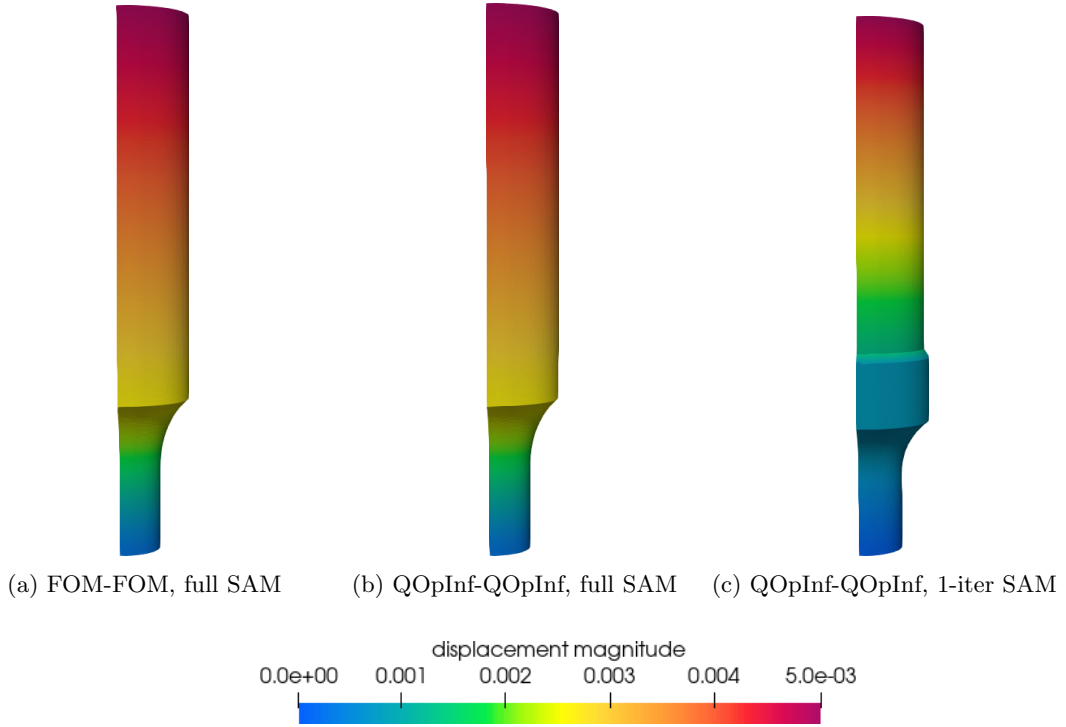
Figure 24: 3D nonlinear hyperelastic tension-specimen problem, reproductive regime: plots of the displacement magnitude at the final time for a FOM-FOM O-SAM-based coupling (a) compared to a QOpInf-QOpInf full Schwarz coupling (b) and a QOpInf-QOpInf Schwarz coupling with only 1 Schwarz iteration (c).

6 Conclusions

In this work, we have presented a hybrid DD-based approach for coupling subdomain-local high-fidelity FOMs with non-intrusive OpInf ROMs using O-SAM. This method addresses significant challenges faced by analysts using traditional high-fidelity simulation codes, reducing both the long runtime requirements and the extensive mesh generation challenges that analysts often face. Our approach mitigates these issues by enabling the seamless "gluing together" of arbitrary combinations of subdomain-local FOMs and non-intrusive OpInf ROMs in a plug-and-play fashion, thereby enhancing online efficiency, and paving the way for more flexible and efficient simulation workflows in engineering applications.
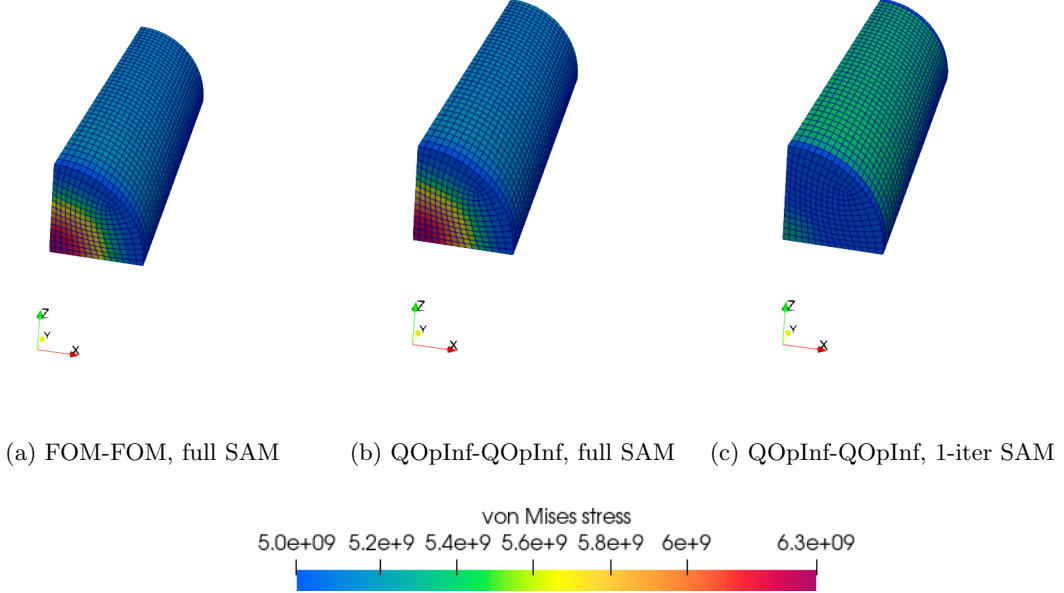
(a) FOM-FOM, full SAM   (b) QOpInf-QOpInf, full SAM   (c) QOpInf-QOpInf, 1-iter SAM

Figure 25: 3D nonlinear hyperelastic tension-specimen problem, predictive regime: plots of the average von Mises stress σ_{vm} in Ω_2 at the final time for a FOM-FOM O-SAM-based coupling (a) compared to a QOpInf-QOpInf full Schwarz coupling (b) and a QOpInf-QOpInf Schwarz coupling with only 1 Schwarz iteration (c).

Through a series of numerical experiments, we have demonstrated the efficacy of our approach on several complex 3D solid dynamics problems characterized by nonlinear behavior, implemented within the `Norma.jl` [27] Julia code. The results indicate that our SAM-based coupling framework not only enhances computational efficiency by a factor as high as $106\times$ with respect to a comparable FOM-FOM coupling performed via O-SAM, but also maintains high accuracy across disparate models, meshes, and time integration schemes. Notably, the extension of our minimally-intrusive SAM-based coupling strategy to non-intrusive OpInf ROMs significantly reduces the implementation burden associated with integrating data-driven models into existing mod/sim workflows. Additionally, we demonstrate that subdomain-local quadratic OpInf (QOpInf) ROMs, when coupled with each other and with (fully nonlinear) FOMs, can deliver highly accurate solutions even when applied to problems with more generic (non-polynomial) nonlinearities. We propose some innovative strategies to enhance the accuracy, robustness and efficiency of SAM when coupling subdomain-local OpInf ROMs, including a method for automatically optimizing subdomain-local OpInf regularization parameters and a mechanism for reducing the size of the learned boundary operator used in the Schwarz iteration process. While attention is focused on solid dynamics, our general O-SAM methodology can be applied to a wide range of PDEs.

The research summarized herein has informed several directions for future work. We are currently in the process of developing a non-overlapping version of SAM that is capable of creating hybrid OpInf-OpInf and OpInf-FOM couplings [47]. The non-overlapping variant of SAM, termed NO-SAM, has several advantages, namely that it is more flexible and can readily handle a wider range of problems than O-SAM, e.g., multi-material problems, multi-physics problems and problems with interfaces. Additionally, we are exploring mechanisms for accelerating SAM through various strategies such as optimized transmission conditions [47], Aitken [10] and Anderson [52] acceleration, and the introduction of additional parallelism in the form of additive Schwarz [17]. Towards enhancing the accuracy of SAM-based ROM-ROM and ROM-FOM-based couplings for highly nonlinear problems, e.g., problems with plasticity, we are extending SAM to work with non-intrusive structure-preserving NN-based OpInf ROMs [41] and other classes

of non-intrusive ROMs, such as kernel manifold ROMs [11, 12]. We are also developing an adaptive algorithm that will enable error indicator-driven online switching between ROMs and FOM in a way that manages both accuracy and efficiency. Towards improving SAM's usability, we are developing a workflow for auto-tuning various SAM inputs (e.g., interface location, subdomain count, overlap size) through multi-objective optimization (e.g., by simultaneously minimizing the CPU time and a stress recovery-based error indicator) using the `GPTune` library [56], which performs a Bayesian, gradient-free optimization of black-box models using interpretable Gaussian process (GP) surrogates. Finally, we are beginning the implementation of our OpInf-FOM O-SAM-based couplings within Sandia's production code, `SIERRA/SM` [28], towards making it accessible to analysts running a variety of mission-critical problems on complex geometries.

Appendix A. Constitutive models

A.1. Linear elastic material model

For a simple linear elastic material, the Helmholtz free energy density takes the form

$$A(\boldsymbol{F}) := \frac{1}{2}\lambda(\mathrm{tr}(\boldsymbol{\epsilon}))^2 + \mu\mathrm{tr}(\boldsymbol{\epsilon}^2), \tag{36}$$

where $\boldsymbol{\epsilon} := \frac{1}{2}(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^{\intercal})$ and $\nabla\boldsymbol{u} := \boldsymbol{F} - \boldsymbol{I}$, with $\boldsymbol{F} := \nabla\boldsymbol{\varphi}$ denoting the deformation gradient, $\boldsymbol{I}$ denoting the 3×3 identity matrix, and $\lambda, \mu > 0$ denoting the Lamé coefficients.

The governing PDEs now take the form (1), where

$$\boldsymbol{\sigma} := \boldsymbol{C} : \boldsymbol{\epsilon}, \tag{37}$$

where $\boldsymbol{C}$ is the fourth-order stiffness tensor.

A.2. Saint Venant–Kirchhoff material model

In addition to describing the mathematical formulation of the Saint Venant–Kirchhoff material model, we also provide a derivation demonstrating that this constitutive model gives rise to PDEs with cubic nonlinearities.

For the Saint Venant–Kirchhoff material model, it is well-known [22] that the Helmholtz free-energy density $A(\boldsymbol{F})$ implicit in the Piola-Kirchhoff stress tensor $\boldsymbol{P}$ takes the form

$$A(\boldsymbol{F}) = A(\boldsymbol{E}) = \frac{\lambda}{2}(\mathrm{tr}\boldsymbol{E})^2 + \mu\mathrm{tr}(\boldsymbol{E}^2), \tag{38}$$

where $\boldsymbol{E} := \frac{1}{2}\boldsymbol{F}^{\intercal}\boldsymbol{F} - \boldsymbol{I}$, with $\boldsymbol{F} := \boldsymbol{I} + \nabla\boldsymbol{u}$ denoting the deformation gradient, and where $\lambda, \mu > 0$ are the Lamé parameters. To derive the strong form of the dynamic solid mechanics PDEs, it is necessary to calculate the first Piola-Kirchhoff stress $\boldsymbol{P} := \frac{\partial A}{\partial \boldsymbol{F}}$. We first calculate the second Piola-Kirchhoff stress, $\boldsymbol{S} := \frac{\partial A}{\partial \boldsymbol{E}}$. The derivative of the first term in (38) is:

$$\frac{\partial(\mathrm{tr}\boldsymbol{E})^2}{\partial \boldsymbol{E}} = 2\mathrm{tr}\boldsymbol{E}\frac{\partial(\mathrm{tr}\boldsymbol{E})}{\partial \boldsymbol{E}} = 2\mathrm{tr}\boldsymbol{E}\frac{\partial(\boldsymbol{E} : \boldsymbol{I})}{\partial \boldsymbol{E}} = 2\mathrm{tr}(\boldsymbol{E})\boldsymbol{I}. \tag{39}$$

Since the second term in (38) is $E_{ij}E_{ij} = \boldsymbol{E} : \boldsymbol{E}$, its derivative with respect to $\boldsymbol{E}$ is $2\boldsymbol{E}$. It follows that

$$\boldsymbol{S} := \frac{\partial A}{\partial \boldsymbol{E}} = \lambda\mathrm{tr}(\boldsymbol{E})\boldsymbol{I} + 2\mu\boldsymbol{E}. \tag{40}$$

Remark that

$$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^{\intercal}\boldsymbol{F} - \boldsymbol{I}) = \frac{1}{2}\left[(\boldsymbol{I} + \nabla\boldsymbol{u})^{\intercal}(\boldsymbol{I} + \nabla\boldsymbol{u}) - \boldsymbol{I}\right] = \frac{1}{2}\left[\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^2 + (\nabla\boldsymbol{u})^{\intercal}\nabla\boldsymbol{u}\right]. \tag{41}$$

Then, (40) can be rewritten as

$$\boldsymbol{S} = \frac{1}{2}\lambda\left[2\mathrm{tr}(\nabla\boldsymbol{u}) + \mathrm{tr}((\nabla\boldsymbol{u})^{\intercal}\nabla\boldsymbol{u})\right]\boldsymbol{I} + \mu\left[\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^2 + (\nabla\boldsymbol{u})^{\intercal}\nabla\boldsymbol{u}\right]. \tag{42}$$

To calculate the first Piola-Kirchhoff stress $\boldsymbol{P}$, we compute

$$
\begin{aligned}
\boldsymbol{P} \;\; & := \frac{\partial A}{\partial \boldsymbol{F}} \\
& = \frac{\partial A}{\partial \boldsymbol{E}} \frac{\partial \boldsymbol{E}}{\partial \boldsymbol{F}} \\
& = \boldsymbol{F}\boldsymbol{S} \\
& = \tfrac{1}{2}\lambda \boldsymbol{F}\left[2\mathrm{tr}(\nabla\boldsymbol{u}) + \mathrm{tr}((\nabla\boldsymbol{u})^{\intercal}\nabla\boldsymbol{u})\right]\boldsymbol{I} + \mu\left[\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^{\intercal} + (\nabla\boldsymbol{u})^{\intercal}\nabla\boldsymbol{u}\right] \\
& = \tfrac{1}{2}\lambda(\boldsymbol{I}+\nabla\boldsymbol{u})\left[2\mathrm{tr}(\nabla\boldsymbol{u}) + \mathrm{tr}((\nabla\boldsymbol{u})^{\intercal}\nabla\boldsymbol{u})\right]\boldsymbol{I} + \mu(\boldsymbol{I}+\nabla\boldsymbol{u})\left[\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^{\intercal} + (\nabla\boldsymbol{u})^{\intercal}\nabla\boldsymbol{u}\right].
\end{aligned}
\tag{43}
$$

Now, the strong form of the PDE is (1), where $\boldsymbol{P}$ is given by (43). In particular, from (43), one can see that the governing PDE (1) has just cubic nonlinearities.

A.3. Neohookean material model

For the Neohookean material model employed herein, the Helmholtz free-energy density is decomposed into a volumetric and deviatoric component

$$
A(\boldsymbol{F}) := A^{\mathrm{vol}}(\boldsymbol{F}) + A^{\mathrm{dev}}(\boldsymbol{F}),
\tag{44}
$$

where

$$
A^{\mathrm{vol}}(\boldsymbol{F}) := \frac{1}{4}\kappa(\det(\boldsymbol{C}) - \log(\det(\boldsymbol{C})) - 1),
\tag{45}
$$

$$
A^{\mathrm{dev}}(\boldsymbol{F}) := \frac{1}{2}\mu((\det(\boldsymbol{C}))^{-1/3}\mathrm{tr}(\boldsymbol{C}) - 3),
\tag{46}
$$

where $\boldsymbol{C} := \boldsymbol{F}^{\intercal}\boldsymbol{F}$ and $\boldsymbol{F}$ denotes the deformation gradient, defined above. In (45)–(46), κ denotes the bulk modulus and μ is a Lamé parameter. From (44), one can shown that

$$
\boldsymbol{S} := \boldsymbol{S}^{\mathrm{vol}} + \boldsymbol{S}^{\mathrm{dev}},
\tag{47}
$$

where

$$
\boldsymbol{S}^{\mathrm{vol}} := \frac{1}{2}\kappa(\det\boldsymbol{C} - 1)\boldsymbol{C}^{-1},
\tag{48}
$$

$$
\boldsymbol{S}^{\mathrm{dev}} := \mu(\det(\boldsymbol{C}))^{-1/3}\left(\boldsymbol{I} - \frac{1}{2}\boldsymbol{C}^{-1}\mathrm{tr}(\boldsymbol{C})\right),
\tag{49}
$$

where $\boldsymbol{I}$ is the 3×3 identity matrix. Now, the first Piola-Kirchhoff stress takes the form $\boldsymbol{P} := \boldsymbol{F}\boldsymbol{S}$, as before.

Code availability and reproducibility

Acknowledgements

References

[1]   Joshua Barnett and Charbel Farhat. "Quadratic approximation manifold for mitigating the Kolmogorov barrier in nonlinear projection-based model order reduction". In: *Journal of Computational Physics* 464 (2022), p. 111348. DOI: 10.1016/j.jcp.2022.111348.

[2]   Joshua Barnett, Charbel Farhat, and Yvon Maday. "Neural-network-augmented projection-based model order reduction for mitigating the Kolmogorov barrier to reducibility". In: *Journal of Computational Physics* 492 (2023), p. 112420. DOI: 10.1016/j.jcp.2023.112420.

[3]   Joshua Barnett, Irina Tezaur, and Alejandro Mota. *The Schwarz alternating method for the seamless coupling of nonlinear reduced order models and full order models*. ArXiv pre-print. 2022. URL: https://arxiv.org/abs/2210.12551.

[4]   Michel Bergmann et al. "A zonal Galerkin-free POD model for incompressible flows". In: *Journal of Computational Physics* 352 (2018), pp. 301–325. DOI: 10.1016/j.jcp.2017.10.001.

[5]   A. P. Boresi and R. J. Schmidt. *Advanced Mechanics of Materials*. John Wiley & Sons, 2013.

[6]   M. Buffoni, H. Telib, and A. Iollo. "Iterative methods for model reduction by domain decomposition". In: *Computers & Fluids* 38.6 (2009), pp. 1160–1167. ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2008.11.008. URL: https://www.sciencedirect.com/science/article/pii/S0045793008002259.

[7]   Youngsoo Choi et al. *Defining Foundation Models for Computational Science: A Call for Clarity and Rigor*. 2025. arXiv: 2505.22904 [cs.LG]. URL: https://arxiv.org/abs/2505.22904.

[8]   Seung Whan Chung et al. "Train small, model big: Scalable physics simulators via reduced order modeling and domain decomposition". In: *Computer Methods in Applied Mechanics and Engineering* 427 (2024), p. 117041. DOI: 10.1016/j.cma.2024.117041.

[9]   Davide Cinquegrana, Raffaele Donelli, and Antonio Viviani. "A hybrid method based on POD and domain decomposition to compute the 2-D aerodynamic flow field - incompressible validation". In: *XX Congresso dell'Associazione Italiana di Meccanica Teorica e Applicata, Bologna, ITA*. 2011, pp. 1–10.

[10] Simone Deparis, Marco Discacciati, and Alfio Quarteroni. "A domain decomposition framework for fluid-structure interaction problems". In: *Computational Fluid Dynamics 2004: Proceedings of the Third International Conference on Computational Fluid Dynamics, ICCFD3, Toronto, 12–16 July 2004*. Springer. 2006, pp. 41–58.

[11] Alejandro N Diaz et al. *Interpretable and flexible non-intrusive reduced-order models using reproducing kernel Hilbert spaces*. 2025. arXiv: 2506.10224 [cs.CE]. URL: https://arxiv.org/abs/2506.10224.

[12] Alejandro N. Diaz et al. *Kernel manifolds: nonlinear-augmentation dimensionality reduction using reproducing kernel Hilbert spaces*. 2025. arXiv: 2509.00224 [cs.CE]. URL: https://arxiv.org/abs/2509.00224.

[13] Marco Discacciati, Ben J. Evans, and Matteo Giacomini. "An overlapping domain decomposition method for the solution of parametric elliptic problems via proper generalized decomposition". In: *Computer Methods in Applied Mechanics and Engineering* 418 (2024), p. 116484. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2023.116484. URL: https://www.sciencedirect.com/science/article/pii/S0045782523006084.

[14] Marco Discacciati, Ben J. Evans, and Matteo Giacomini. *PGD-based local surrogate models via overlapping domain decomposition: a computational comparison*. 2025. arXiv: 2508.01313 [math.NA]. URL: https://arxiv.org/abs/2508.01313.

[15] Niccolò Discacciati and Jan S. Hesthaven. "Model reduction of coupled systems based on non-intrusive approximations of the boundary response maps". In: *Computer Methods in Applied Mechanics and Engineering* 420 (2024), p. 116770. DOI: 10.1016/j.cma.2024.116770.

[16] Ionut-Gabriel Farcas et al. "Domain Decomposition for Data-Driven Reduced Modeling of Large-Scale Systems". In: *AIAA Journal* 0.0 (2024), pp. 1–16. DOI: 10.2514/1.J063715.

[17] Martin J. Gander. "Schwarz methods over the course of time". In: *Electronic Transactions on Numerical Analysis* 31 (2008), pp. 228–255.

[18]   Rudy Geelen, Stephen Wright, and Karen Willcox. "Operator inference for non-intrusive model reduction with quadratic manifolds". In: *Computer Methods in Applied Mechanics and Engineering* 403 (2023), p. 115717. DOI: 10.1016/j.cma.2022.115717.

[19]   Leonidas Gkimisis et al. "Non-intrusive reduced-order modeling for dynamical systems with spatially localized features". In: *Computer Methods in Applied Mechanics and Engineering* 444 (2025), p. 118115. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2025.118115. URL: https://www.sciencedirect.com/science/article/pii/S0045782525003871.

[20]   Anthony Gruber and Irina Tezaur. "Variationally Consistent Hamiltonian Model Reduction". In: *SIAM Journal on Applied Dynamical Systems* 24.1 (2025), pp. 376–414.

[21]   P. Holmes, J.L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 1996.

[22]   Gerhard A. Holzapfel. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. English. Wiley, 2000, p. 455. ISBN: 9780471823049.

[23]   K. Kunisch and S. Volkwein. "Galerkin proper orthogonal decomposition methods for parabolic problems". In: *Numerische Mathematik* 90.1 (2001), pp. 117–148. DOI: 10.1007/s002110100282.

[24]   Sandia National Laboratories. *Automating complex 3D modeling*. Accessed: 2025-10-31. 2020. URL: https://www.sandia.gov/labnews/2020/04/23/vorocrust/.

[25]   Sandia National Laboratories. *CUBIT Mesher*. Accessed: 2025-10-23. 2023. URL: https://cubit.sandia.gov/.

[26]   Sandia National Laboratories. *norma-opinf*. Accessed: 2025-10-23. 2023. URL: https://github.com/sandialabs/norma-opinf.

[27]   Sandia National Laboratories. *Norma.jl: A Julia testbed for coupling and multiphysics*. https://github.com/sandialabs/Norma.jl. Commit on GitHub: accessed on September 2, 2025. 2025.

[28]   Sandia National Laboratories. *SIERRA/SM: A Multi-Physics Simulation Framework*. Accessed: 2025-11-07. 2023. URL: https://www.sandia.gov/sierra/.

[29]   Kookjin Lee and Kevin T. Carlberg. "Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders". In: *Journal of Computational Physics* 404 (2020), p. 108973. DOI: 10.1016/j.jcp.2019.108973.

[30]   Ke Li et al. "D3M: A Deep Domain Decomposition Method for Partial Differential Equations". In: *IEEE Access* 8 (2020), pp. 5283–5294. DOI: 10.1109/ACCESS.2019.2957200.

[31]   Wuyang Li, Xueshuang Xiang, and Yingxiang Xu. "Deep Domain Decomposition Method: Elliptic Problems". In: *Proceedings of Machine Learning Research* 107 (2020), pp. 269–286.

[32]   Luis Mandl et al. "Separable physics-informed DeepONet: Breaking the curse of dimensionality in physics-informed machine learning". In: *Computer Methods in Applied Mechanics and Engineering* 434 (2025), p. 117586. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2024.117586. URL: https://www.sciencedirect.com/science/article/pii/S0045782524008405.

[33]   Shane A. McQuarrie, Cheng Huang, and Karen E. Willcox. "Data-driven reduced-order models via regularised Operator Inference for a single-injector combustion process". In: *Journal of the Royal Society of New Zealand* 51.2 (2021), pp. 194–211. DOI: 10.1080/03036758.2020.1863237. eprint: https://doi.org/10.1080/03036758.2020.1863237. URL: https://doi.org/10.1080/03036758.2020.1863237.

[34]   Ian Moore et al. *Domain Decomposition-Based Coupling of High-Fidelity Finite Element and Reduced Order Operator Inference Models Using the Schwarz Alternating Method*. 2025. arXiv: 2510.05350 [math.NA]. URL: https://arxiv.org/abs/2510.05350.

[35]   Ian Moore et al. *Domain Decomposition-based coupling of Operator Inference reduced order models via the Schwarz alternating method*. 2024. arXiv: 2409.01433.

[36]   A. Mota, W. Klug, M. Ortiz, et al. "Finite-element simulation of firearm injury to the human cranium". In: *Computational Mechanics* 31 (2003), pp. 115–121. DOI: 10.1007/s00466-002-0398-8.

[37]   Alejandro Mota, Irina Tezaur, and Coleman Alleman. "The Schwarz alternating method in solid mechanics". In: *Computer Methods in Applied Mechanics and Engineering* 319 (2017), pp. 19–51. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2017.02.006. URL: https://www.sciencedirect.com/science/article/pii/S0045782516310982.

[38] Alejandro Mota, Irina Tezaur, and Gregory Phlipot. "The Schwarz alternating method for dynamic solid mechanics". In: *International Journal for Numerical Methods in Engineering* (2022), pp. 1–36. DOI: 10.1002/nme.6982.

[39] Alejandro Mota and Jonathan A. Zimmerman. "A variational, finite-deformation constitutive model for piezoelectric materials". In: *International Journal for Numerical Methods in Engineering* 85.6 (2011), pp. 752–767. DOI: https://doi.org/10.1002/nme.2993. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.2993. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2993.

[40] Alejandro Mota et al. "A Fundamentally New Coupled Approach to Contact Mechanics via the Dirichlet-Neumann Schwarz Alternating Method". In: *International Journal for Numerical Methods in Engineering* 126.9 (2025), e70039. DOI: https://doi.org/10.1002/nme.70039. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.70039. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.70039.

[41] E. Parish, A. Gruber, and I. Tezaur. *Non-intrusive reduced-order modeling of dynamical systems using structure-preserving composable neural networks*. in preparation. 2025.

[42] E. Parish, P. Lindsay, T. Shelton, et al. "Embedded symmetric positive semi-definite machine-learned elements for reduced-order modeling in finite-element simulations with application to threaded fasteners". In: *Computational Mechanics* 74 (2024), pp. 1357–1381. DOI: 10.1007/s00466-024-02481-5.

[43] Benjamin Peherstorfer. "Model Reduction for Transport-Dominated Problems via Online Adaptive Bases and Adaptive Sampling". In: *SIAM Journal on Scientific Computing* 42.5 (2020), A2803–A2836. DOI: 10.1137/19M1257275.

[44] Benjamin Peherstorfer and Karen Willcox. "Data-driven operator inference for nonintrusive projection-based model reduction". In: *Computer Methods in Applied Mechanics and Engineering* 306 (2016), pp. 196–215.

[45] Elizabeth Qian, Ionuţ-Gabriel Farcaş, and Karen Willcox. "Reduced Operator Inference for Nonlinear Partial Differential Equations". In: *SIAM Journal on Scientific Computing* 44.4 (2022), A1934–A1959. DOI: 10.1137/21M1393972. URL: https://doi.org/10.1137/21M1393972.

[46] M. Raissi, P. Perdikaris, and G.E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational Physics* 378 (2019), pp. 686–707. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.10.045. URL: https://www.sciencedirect.com/science/article/pii/S0021999118307125.

[47] Cameron Rodriguez et al. *Transmission Conditions for the Non-Overlapping Schwarz Coupling of Full Order and Operator Inference Models*. 2025. arXiv: 2509.12228 [math.NA]. URL: https://arxiv.org/abs/2509.12228.

[48] Henrik Rosenberger, Benjamin Sanderse, and Giovanni Stabile. *Exact operator inference with minimal data*. 2025. arXiv: 2506.01244 [math.NA]. URL: https://arxiv.org/abs/2506.01244.

[49] Hermann Amandus Schwarz. *Ueber einen Grenzübergang durch alternirendes Verfahren*. Zürcher u. Furrer, 1870.

[50] L. Sirovich. "Turbulence and the dynamics of coherent structures, part III: dynamics and scaling". In: *Q. Appl. Math.* 45.3 (1987), pp. 583–590.

[51] Will Snyder, Irina Tezaur, and Christopher Wentland. *Domain decomposition-based coupling of physics-informed neural networks via the Schwarz alternating method*. 2023. arXiv: 2311.00224 [math.NA].

[52] Homer F. Walker and Peng Ni. "Anderson Acceleration for Fixed-Point Iterations". In: *SIAM Journal on Numerical Analysis* 49.4 (2011), pp. 1715–1735. DOI: 10.1137/10078356X. eprint: https://doi.org/10.1137/10078356X. URL: https://doi.org/10.1137/10078356X.

[53] Hengjie Wang et al. "Mosaic flows: A transferable deep learning framework for solving PDEs on unseen domains". In: *Computer Methods in Applied Mechanics and Engineering* 389 (2022), p. 114424. DOI: 10.1016/j.cma.2021.114424.

[54] Wei Wang et al. *Time Marching Neural Operator FE Coupling: AI Accelerated Physics Modeling*. 2025. arXiv: 2504.11383 [cs.LG]. URL: https://arxiv.org/abs/2504.11383.

[55] Christopher R. Wentland et al. *The role of interface boundary conditions and sampling strategies for Schwarz-based coupling of projection-based reduced order models.* ArXiv pre-print. 2024. URL: https://arxiv.org/abs/2410.04668.

[56] Zhiqiang Yuan et al. "GPTune: A Framework for Automated Hyperparameter Tuning of Machine Learning Models". In: *Proceedings of the 2021 IEEE International Conference on Big Data (Big Data)*. IEEE, 2021, pp. 1234–1242. DOI: 10.1109/BigData52589.2021.9671740.