

Constrained Optimization of Whitney Forms for Nonlinear Projection-Based Reduced Order Methods

Katherine Piper^{a,*}, Lorenz Biegler^b, Nathaniel Trask^c, Kaushik Dayal^a

^a*Department of Civil and Environmental Engineering, Carnegie Mellon University, Pittsburgh PA, USA*

^b*Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh PA, USA*

^c*Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia PA, USA*

Abstract

In this work, we present the NLP-ROM, a method for finding Machine-Learned Whitney forms for nonlinear model order reduction through constrained optimization. We first demonstrate the performance of the methods on the Poisson problem for a unit mesh, before demonstrating the applicability to free boundary linear elasticity on an unstructured mesh.

Keywords: Projection-Based ROMs, Whitney Forms, Structure-Preserving Machine Learning, Nonlinear Optimization

Nomenclature

$\mathbf{W} \in \mathbb{R}^{N \times M}$, $M \ll N$ - Convex Combination Tensor satisfying the POU property

M Degrees of Freedom in coarse mesh

N Degrees of Freedom in fine mesh

$\xi_i(x)$, $i = 1 \dots N$ - Finite Element basis functions

$u(x)$ - scalar solution

$f(x)$ - forcing function

$v(x)$ - test function associated with scalar solution $u(x)$

e - vector of ones

1. Introduction

Reduced order methods (ROMs) allow for accurate finite element modeling and simulation with minimal error to full scale models, often at a fraction of the cost. While a range of ROMs have been developed [1], recent literature has been split between two primary methods: Projection Based ROMs, and Structure Preserving Neural Networks (SPNNs). Projection-Based ROMs frequently use either proper Orthogonal Decomposition (POD)[2–7] or Singular Value Decomposition (SVD)[8–10] to create a reduced order basis, while SPNNs, use machine learning algorithms to define a Whitney Form basis using a PDE-constrained optimization [11–15]. Projection Based ROMs often lead to a 10-50% reduction in computational cost for full scale models[7], while SPNNs can result in a 95-99% reduction in computational cost[13], at the expense of a higher training cost. Collectively, these ROMs have a wide range of use cases, including multiscale modeling and inverse problems [1]. Both projection-based ROMs and SPNNs produce accurate solutions in limited data regimes.

In this work, we will demonstrate how determining a reduced-order finite element basis can alternatively be thought of as an optimal control problem. In machine learning (ML), a model is considered converged

*Corresponding author

Email addresses: kbpiper@andrew.cmu.edu (Katherine Piper), lb01@cmu.edu (Lorenz Biegler), ntrask@seas.upenn.edu (Nathaniel Trask), kaushik.dayal@cmu.edu (Kaushik Dayal)

when the gradient of the function is below a predetermined value for convergence. The exact performance and convergence conditions may vary slightly by optimizer [16]. In nonlinear programming (NLP), a model has converged when the first-order Karush–Kuhn–Tucker (KKT) conditions are satisfied within a tolerance [17]. The result is that NLPs provide solutions to optimal control problems which directly consider all constraints, while ML methods only consider the change in the gradient. For large systems, generalized reduced gradient (GRG) methods and primal-dual interior point methods (IPM) may be used. In GRG algorithms, such as CONOPT, a minimum is found by performing Newton’s method with an Armijo line search on the constraints until KKT conditions are satisfied [18] In IPM algorithms, such as IPOPT, the optimization is instead performed over the Lagrangian[19, 20], improving solver efficiency and speed while relying on satisfying KKT conditions for convergence[21].

Inverse problems are often computationally expensive, and have been approached using a range methods. Novel ideas, including machine-learning informed computer vision algorithms[22], complex resolution refinement schemes [23], introduction of Fourier neural operators [24], and several other methods are actively being investigated for efficiently solving inverse problems. Standard Projection-Based ROMs have also shown promise in solving inverse problems [25]. In this work, we aim to further develop the methods used for inverse problem solving with Projection-Based ROMs to instead use a Whitney form basis, enabling further improved inverse solving methods.

This work is divided into three sections. We first discuss the development of NLP-based Reduced Order Methods (NLP-ROMs), demonstrating how a finite element (FEM) basis can be reduced using Whitney forms. In doing so, we demonstrate the numerical equivalence between a Whitney form basis and Galerkin-discretized projection-based ROMs. We then begin by applying our ROM to the Poisson equation on the unit square, to demonstrate the equivalence between our Whitney forms and Whitney forms determined through optimization using an SPNN. Next, we demonstrate the applicability of our NLP-ROM to free boundary linear elasticity problems, providing a model reduction order of over 95% while maintaining low error.

2. Problem Setup

In this section, we demonstrate the discretization used in developing our nonlinear basis optimization algorithm, and demonstrate, by following the discretization methods from traditional Projection-Based ROMs, we develop the convergence conditions are numerically equivalent to the minimum eventually reached through gradient descent algorithms.

2.A. Finite Element Notation

Finite Element Methods frequently use slightly different notations across fields. We present our notation for a Galerkin discretization of the Poisson Equation using first order Lagrangian basis functions as follows:

The standard Poisson equation $\Delta u = -f$ with Dirichlet boundary conditions gives the weak form

$$\int_{\Omega} \nabla v \cdot \nabla u = - \int_{\Omega} v f, \text{ using a test function } v. \quad (1)$$

Using a Galerkin discretization, with $u = \sum_j u_j \xi_j, v = \xi_i$, for first-order Lagrange basis functions ξ_i , and setting boundary terms to zero, we find:

$$\sum_j u_j \int_{\Omega} \nabla \xi_i \cdot \nabla \xi_j = \int_{\Omega} \xi_i f =: f_i. \quad (2)$$

By defining a stiffness matrix $K_{ij} = \int_{\Omega} \nabla \xi_i \cdot \nabla \xi_j$, $K \in \mathbb{R}^{N \times N}$, we develop the linear system

$$K_{ij} u_j = f_i. \quad (3)$$

2.B. Reduced Basis Notation

Introducing a convex combination order-reduction tensor $\mathbf{W}_{ij} \in \mathbb{R}^{N \times M}$, we redefine the discretization

$$u = \sum_{ij} \hat{u}_i \mathbf{W}_{ij}^T \xi_j, \quad v_k = \sum_l \mathbf{W}_{kl}^T \xi_l. \quad (4)$$

We note this convex combination tensor serves the same function as a projection matrix in decomposition-based reduced order methods, including POD and SVD. Unlike POD or SVD, however, we choose to define \mathbf{W}_{ij} as the nonunique collection of Whitney 0-forms, characterized by

$$\mathbf{W}_{ij} \geq 0, \quad \sum_j \mathbf{W}_{ij} = 1 \quad (5)$$

We chose to provide an order reduction tensor in terms of Whitney forms for their structure-preserving properties and demonstrated performance of Whitney forms in machine learning applications. Higher-order Whitney forms, which parameterize fluxes, can be derived through application of the exterior derivative. For further details on these derivations and use cases, we refer the reader to Trask 2022[12]. For the remainder of this section, we exclusively refer to the collection of 0-forms $:= \mathbf{W}$. In this work, the critical property of Whitney forms is that they form a Partition of Unity (POU). The set of all POU is closed under convex combinations:

$$\text{if } \sum_i \mathbf{W}_{ij} = 1 \text{ and } \sum_j \xi_j = 1 \text{ then } \sum_i \sum_j \mathbf{W}_{ij} \xi_j = 1, \quad (6)$$

By this property, the projection $\mathbf{W}_{ij} \xi_j$ created using Whitney forms remains a valid finite element basis, guaranteeing solution adherence to the discretized governing law. While prior work has found success by partitioning \mathbf{W} into boundary and interior elements, \mathbf{W}_{bdry} and \mathbf{W}_{int} [12, 13], we find this unnecessary within our finite element framework. Applying this projection to the weak form from Equation 3, we derive the new linear equation

$$\mathbf{W}_{ij}^T K_{jl} \mathbf{W}_{lk} \hat{u}_k = \mathbf{W}_{im}^T f_m := \hat{f}_i \quad (7)$$

This equation is trilinear in \mathbf{W} , which significantly increases the computation cost. By introducing a linearization factor $B := \mathbf{W}_{ij}^T K_{jl} \mathbf{W}_{lk}$, we remove all 3rd-order terms, yielding the final optimization:

$$\min_{\mathbf{W}, \hat{u}, B} \|\mathbf{W} \hat{u} - u\|^2, \quad (8)$$

Using this nonlinear program, we aim to determine an optimal \mathbf{W} which minimizes the error $\|\mathbf{W} \hat{u} - u\|^2$, where u is determined from data.

3. Optimization

In this section, we derive the KKT conditions of the NLP. We reiterate the optimization statement:

$$\min_{\mathbf{W}, \hat{u}, B} \|\mathbf{W} \hat{u} - u\|^2, \quad (9)$$

subject to four constraints:

$$B \hat{u} = \hat{f}, B = \mathbf{W}_{ij}^T K_{jl} \mathbf{W}_{lk}, \mathbf{W} > 0, \text{ and } \mathbf{W}_{ij} e_j = e_i \quad (10)$$

The Lagrangian is given by:

$$\mathcal{L}(\mathbf{W}, \hat{u}, B, \lambda_1, \lambda_2, \lambda_3, \mu) = \|\mathbf{W}\hat{u} - u\|^2 + \lambda_1^T (B\hat{u} - \hat{f}) + \text{tr}(\Lambda_2^T (B - \mathbf{W}^T K \mathbf{W})) + \lambda_3^T (\mathbf{W}\mathbf{e} - \mathbf{e}) - \mu^T \mathbf{W} \quad (11)$$

where λ_i is used for equality constraints, and μ is used for the inequality constraint. This Lagrangian is subject to the primal constraints:

$$B\hat{u} = \hat{f} \quad (12a)$$

$$B = \mathbf{W}_{ij}^T K_{jl} \mathbf{W}_{lk} \quad (12b)$$

$$\mathbf{W}_{ij} \mathbf{e}_j = \mathbf{e}_i \quad (12c)$$

$$\mathbf{W}_{ij} \geq 0 \quad \forall \quad i, j \quad (12d)$$

The dual constraint and complimentary slackness:

$$\mu \geq 0 \quad (13a)$$

$$\mu_{ij} \mathbf{W}_{ij} = 0 \quad (13b)$$

And the stationary conditions:

$$\frac{\partial \mathcal{L}}{\partial \hat{u}} = 0 : \quad 2\mathbf{W}^T (\mathbf{W}\hat{u} - u) + B^T \lambda_1 = 0 \quad (14a)$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}} = 0 : \quad 2(\mathbf{W}\hat{u} - u)\hat{u}^T - 2K\mathbf{W}\Lambda_2 + \lambda_3 \mathbf{e}^T - \mu = 0 \quad (14b)$$

$$\frac{\partial \mathcal{L}}{\partial B} = 0 : \quad \lambda_1 \hat{u}^T + \lambda_2 = 0 \quad (14c)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_1} = 0 : \quad B\hat{u} = \hat{f} \quad (14d)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_2} = 0 : \quad B = \mathbf{W}^T K \mathbf{W} \quad (14e)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_3} = 0 : \quad \mathbf{W}\mathbf{e} = \mathbf{e} \quad (14f)$$

This nonlinear program is formulated as a quadratically constrained quadratic program (QCQP). \mathbf{W} and \hat{u} are quadratic both in the constraints and in the objective functions. QCQPs are a numerically favorable form for several NLP solvers. While this particular QCQP is nonconvex, the numerical structure remains favorable for quickly finding a local minimum within specified tolerance.

3.A. Algorithm Selection

A range of nonlinear optimization algorithms demonstrate good performance on QCQPs. Two feasible algorithms for large nonconvex QCQPs are generalized reduced gradient algorithms, and primal-dual interior point method (barrier) algorithms. We classify the performance of these algorithms on three factors. These factors are (1) optimizer wall time, (2) objective function value, and (3) physical relevance of subdomains for nonlinear problems. Consideration (3) is an especially desirable trait of Whitney forms from the SPNN literature[13], as it preserves low error under mesh refinement, without retraining. Capturing localized, domain-relevant features, such as high-frequency modes, generally requires further computational analysis and constraints [26]. Both algorithms are evaluated using the manufactured problem from Section 2, on a domain of the unit square. Example learned bases from each of these algorithms are provided in Figure 1.

3.A.1. Generalized Reduced Gradient (CONOPT algorithm)

CONOPT is a generalized reduced gradient (GRG) algorithm designed for large scale sparse nonlinear optimization problems [18]. In our tests, CONOPT performed poorly, with high wall time, relatively high objective functions, and solutions demonstrating little physical relevance over the domain.

3.A.2. Primal-Dual Interior Point Method (barrier) algorithms

Primal-dual interior point method (barrier) algorithms are designed for large scale highly nonlinear problems where second order conditions must be strictly enforced [21]. Of tested algorithms, IPOPT, an open source algorithm [27], outperformed both generalized reduced gradient methods and commercial barrier algorithm implementations. On manufactured problems, IPOPT, with default options and linear solver mumps, outperformed other optimizers in all metrics, including domain quality/physical relevance, optimizer wall time, and objective function value. For these reasons, we use the IPOPT algorithm for the remainder of this work.

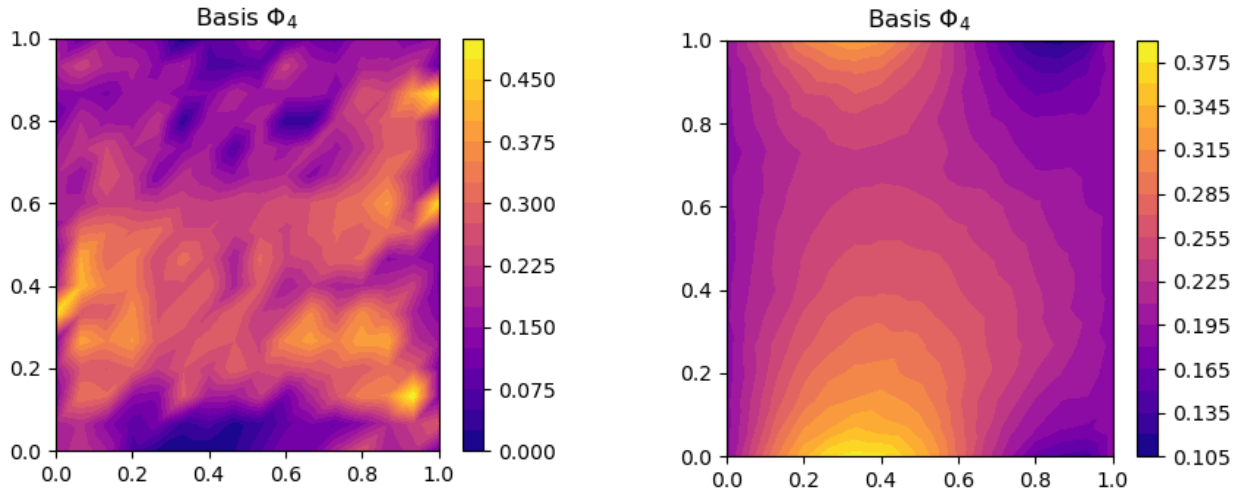


Figure 1: Sample Bases from tested optimizers, demonstrating the ability of IPOPT to capture locally-relevant subdomains

4. Results

In this section, we first develop a basic model of the Poisson equation $\Delta u = f$ on the unit square. We select this problem due to its structural relevance to a range of problems of interest, including linear elasticity, Darcy flow, magnetostatics, and the Helmholtz equation. We then demonstrate the feasibility of our method in learning a nonlinear reduced bases for a free-boundary linear elasticity problem on an unstructured mesh.

4.A. Poisson Problem

The Poisson equation on a unit mesh with Dirichlet boundary conditions at $x = 0$ and $x = 1$ and Neumann boundary conditions at $y = 0$ and $y = 1$ is discretized using the methods from Section 2, to obtain the form

$$\mathbf{W}_{ij}^T K_{jl} \mathbf{W}_{lk} \hat{u}_k = \mathbf{W}_{im}^T f_m := \hat{f}_i \quad (15)$$

We define a fine-space mesh of 15×15 elements, yielding 256 nodes. We compute training data over this fine-space mesh using FEniCSx, which returns the stiffness matrix, RHS vectors, and fine-mesh solution. We note data can be generated through a range of means beyond fine-mesh FEM, including through DIC and other experimental methods. To solve for the Whitney form projection matrix $\mathbf{W} \in \mathbb{R}^{256 \times 5}$, we follow the steps from Section 3, and solve the NLP using IPOPT 3.13.2 in pyomo, running with linear solver mumps.

4.A.1. Dirichlet Conditions

For Dirichlet boundary conditions, we optimize Whitney forms to minimize a basis for a combination of forcing functions. We generate training data of the form $\Delta u = f$ on a fine mesh for a range of forcing functions, designed to capture a range of possible behaviors, including low frequency forcing, polynomial forcing, exponential forcing, and higher frequency or mixed-mode forcing. The optimization algorithm takes inputs of stiffness matrix $K_i \in \mathbb{R}^{256 \times 256}$, $f_i \in \mathbb{R}^{256}$, and $u_{i,target} := K_i^{-1} f_i \in \mathbb{R}^{256}$, $i \in \{1, 2, 3, 4\}$, and returns $W_{opt} \in \mathbb{R}^{256 \times 5}$. Test solutions in the coarse mesh space are then computed by solving $W_{ij}^T K_{jl} W_{lk} \hat{u}_k = W_{im}^T f_m$, and projecting the coarse space solution, \hat{u}_k onto the fine space using $W \hat{u}_k := \underline{u}_{test}$. Forcing functions are selected to test a range of interpolation and extrapolation behavior, as well as various combinations of features from the training data. Error for these tests is then defined as $\varepsilon := \frac{1}{2} \|u_{fine} - \underline{u}_{test}\|^2$. We present a range of test functions and errors, as well as the approximate speedup in solver wall time compared to fine mesh FEM solutions.

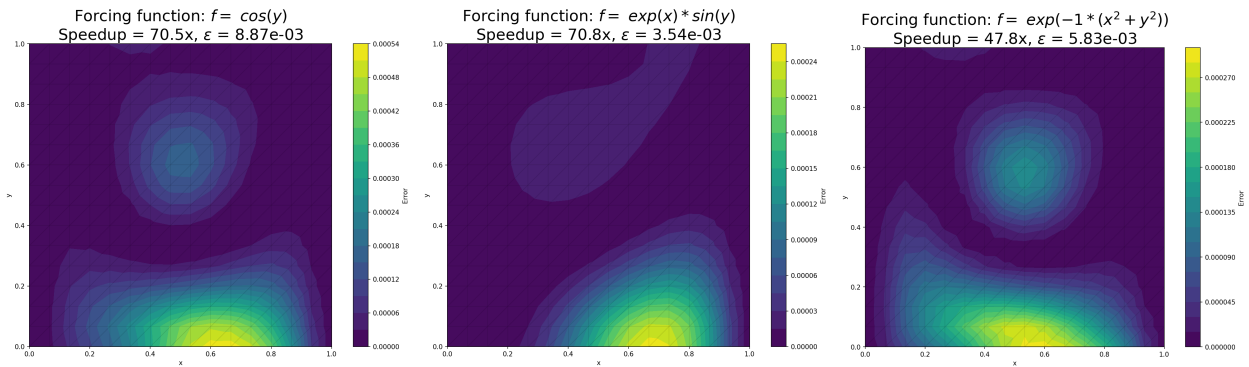


Figure 2: Coarse Solution Error for a Range of Test Functions with Zero Dirichlet Boundary Conditions

4.A.2. Non-zero Dirichlet Conditions

In cases where the boundary conditions are nonzero, the forcing terms contain information regarding the boundary conditions. In this section, we demonstrate that our nonlinear optimization still applies in these cases. We again define the domain to be the unit square, discretized into a 15×15 mesh, which yields a linear equation $Ku = f$, $K \in \mathbb{R}^{256 \times 256}$, $u, f \in \mathbb{R}^{256}$. We define Neumann boundary conditions on y , and Dirichlet conditions of $u = 0$ on $x = 0$, and $u = 1$ on $x = 1$.

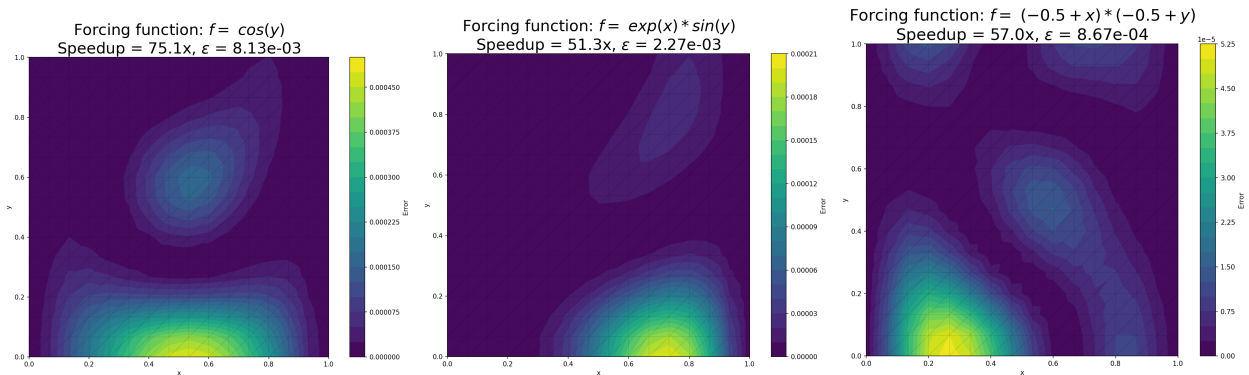


Figure 3: Coarse Solution Error for a Range of Test Functions with Nonzero Dirichlet Boundary Conditions

We also demonstrate the robustness of these methods to spatially varying boundary conditions, with Neumann boundary conditions on y , and Dirichlet conditions of $u = 0$ on $x = 0$, and $u = \sin y$ on $x = 1$.

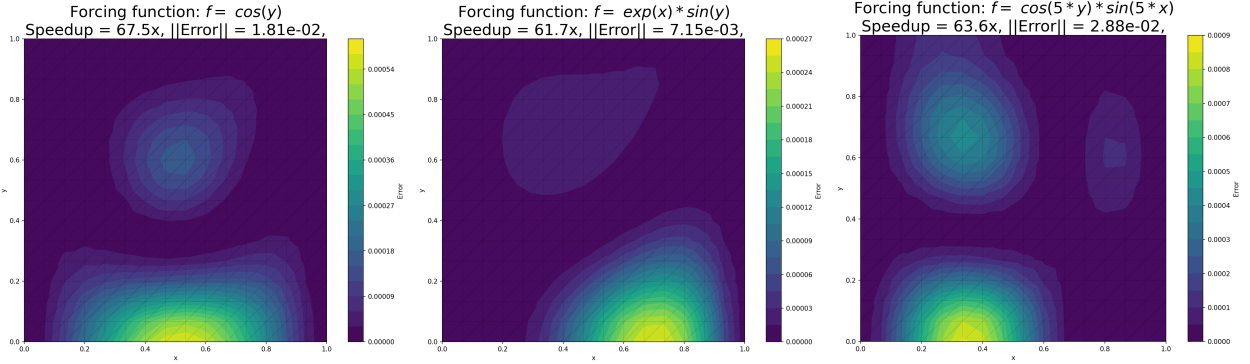


Figure 4: Coarse Solution Error for a Range of Test Functions with Spatially Varying Dirichlet Boundary Conditions

5. Linear Elasticity

In this section, we demonstrate the ability to capture relevant subdomains in free-boundary linear elasticity. For linear elasticity with $EI = 1$, the model reduces to the Poisson problem. Two changes are made to the previously designated algorithm for large systems. In this section we use IPOPT with HSL solver MA97, MC64 scaling, 10^{-20} pivot tolerance, 10^{-8} convergence tolerance, and 10^{-4} acceptable tolerance. Additionally, we initialize from a feasible point, and implement a bound push and bound frac of 10^{-3} . For larger problems, we construct the Whitney form basis using a greedy algorithm 1. This algorithm ensures that,

Algorithm 1 Learning sequence for \mathbf{W}

Require: Training scenarios $\{K^\alpha u_{data}^\alpha = f^\alpha\}_{\alpha=1}^{n_k}$, convergence tolerance $\varepsilon > 0$

- 1: Initialize $\mathbf{W} \in \mathbb{R}^{1 \times n_i}$ with $[\mathbf{W}^{(1)}]_{1,:} = \mathbf{1}$, set $n_w \leftarrow 1$
- 2: **for** $k = 1, 2, \dots, n_k$ **do** ▷ Outer loop: grow scenario set
- 3: $\mathcal{L} \leftarrow \infty$
- 4: **while** $\mathcal{L} > \varepsilon$ **do** ▷ Inner loop: grow basis dimension
- 5: $n_w \leftarrow n_w + 1$, expand \mathbf{W} to $\mathbb{R}^{n_w \times n_i}$
- 6: Solve the NLP:

$$\min_{\mathbf{W}, \hat{u}, B} \left\| \sum_{\alpha} \mathbf{W} \hat{u}^{\alpha} - u \right\|^2$$

subject to:

$$\begin{aligned} B^{\alpha} \hat{u}^{\alpha} &= \hat{f}, \\ B^{\alpha} &= \mathbf{W}^T K^{\alpha} \mathbf{W}, \\ \mathbf{W}_{ij} &> 0 \quad \forall \quad i, j, \\ \mathbf{W}_{ij} \mathbf{e}_j &= \mathbf{e}_i \end{aligned}$$

$$7: \quad \mathcal{L} \leftarrow \left\| \sum_{\alpha} \mathbf{W} \hat{u}^{\alpha} - u^{\alpha} \right\|^2$$

8: **end while**

9: **end for**

Ensure: \mathbf{W}

after the first iteration, each future iteration of the NLP is initialized near a "good" solution, resulting in faster convergence, often within 20 iterations for later subproblems. Additionally, through the greedy construction, this algorithm learns the minimum number of dominant modes for a nonlinear free boundary problem, giving meaningful physical data about the system.

5.A. Results

We use a free boundary elasticity problem on an unstructured mesh of an arch bridge to demonstrate the ability of our learning methods to capture relevant domains for nonlinear problems. We define Dirichlet boundary conditions of zero on the edges of the mesh, and apply a point load some distance d from the center of the mesh. We aim to learn a reduced space operator which is independent of the location of the applied load. The mesh and boundary conditions are shown in Figure 5. After training, we obtain an order reduction

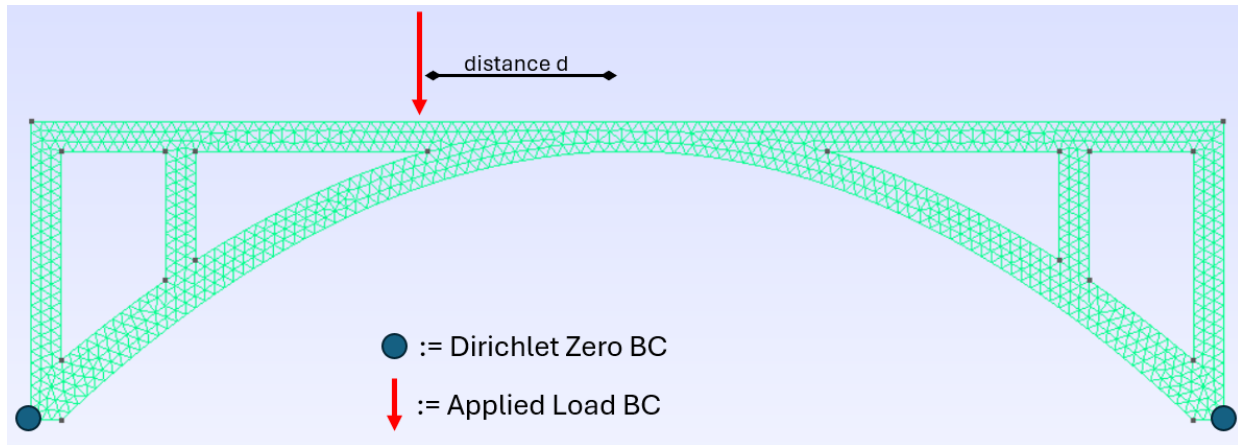


Figure 5: Boundary Conditions on an unstructured mesh of an arch bridge

\mathcal{W} of size 7726×5 , resulting in the linear equation $B\hat{u} = \hat{f}$, where $B \in \mathbb{R}^{5 \times 5}$, which is very cheap to solve. We compute the predicted displacement for both the full order model and the reduced order model, and find a maximum relative error of 7.8% and an average relative error of 3.15%, while also seeing a 70-fold reduction in cost versus solving the full order linear system. A plot of the average absolute error for a wide range of loading locations is shown in Figure 6.

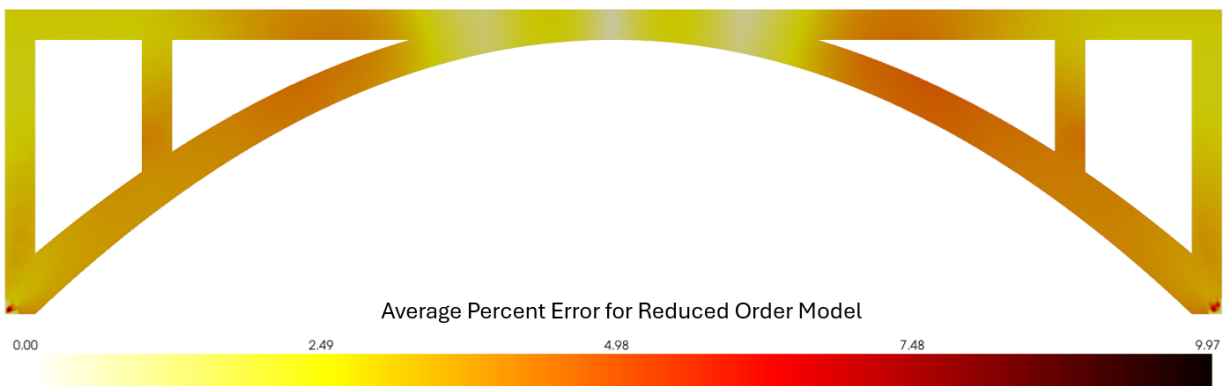


Figure 6: Average Error by mesh location for a range of applied loading conditions

6. Conclusion

In this work, we develop the NLP-ROM, a method for modeling reduced order nonlinear operators based on Whitney forms. We demonstrate the numerical equivalence between Whitney forms learned through NLP-based methods and Whitney forms learned through Structure-Preserving Neural Networks, and apply these NLP-based methods to free boundary elasticity problem. Future extensions to this work will include problems with additional constraints and refinement studies. NLP-based methods for learning operators show significant promise for constrained problems, where gradient-descent based algorithms may struggle with implementation of constraints.

References

- [1] F. Salmoiraghi *et al.*, “Advances in geometrical parametrization and reduced order models and methods for computational fluid dynamics problems in applied sciences and engineering: overview and perspectives,” in *Proceedings of the ECCOMAS Congress 2016, 7th European Conference on Computational Methods in Applied Sciences and Engineering*, vol. 1, pp. 1013–1031, Institute of Structural Analysis and Antiseismic Research School of Civil Engineering National Technical University of Athens (NTUA) Greece., 2016.
- [2] S. Rubino, “Numerical analysis of a projection-based stabilized pod-rom for incompressible flows,” *SIAM Journal on Numerical Analysis*, vol. 58, no. 4, pp. 2019–2058, 2020.
- [3] M. Balajewicz and E. H. Dowell, “Stabilization of projection-based reduced order models of the navier–stokes,” *Nonlinear Dynamics*, vol. 70, no. 2, pp. 1619–1632, 2012.
- [4] R. Elzohery and J. Roberts, “Application of model-order reduction of non-linear time-dependent neutronics via pod-galerkin projection and matrix discrete empirical interpolation,” *Annals of Nuclear Energy*, vol. 179, p. 109396, 2022.
- [5] D. Amsallem and C. Farhat, “On the stability of reduced-order linearized computational fluid dynamics models based on pod and galerkin projection: descriptor vs non-descriptor forms,” in *Reduced order methods for modeling and computational reduction*, pp. 215–233, Springer, 2014.
- [6] X. Sui, K. Djidjeli, Z. Sun, and J. T. Xing, “Reduced order modeling (rom) based method for the two-dimensional water exit problem using snapshot proper orthogonal decomposition (pod) and cfd simulations,” *Applied Ocean Research*, vol. 161, p. 104697, 2025.
- [7] J. L. Nicolini, D.-Y. Na, and F. L. Teixeira, “Model order reduction of electromagnetic particle-in-cell kinetic plasma simulations via proper orthogonal decomposition,” *IEEE Transactions on Plasma Science*, vol. 47, no. 12, pp. 5239–5250, 2019.
- [8] X. Xie, M. Mohebujjaman, L. G. Rebholz, and T. Iliescu, “Data-driven filtered reduced order modeling of fluid flows,” *SIAM Journal on Scientific Computing*, vol. 40, no. 3, pp. B834–B857, 2018.
- [9] X. Wang and V. Yang, *Emulation of Complex Fluid Flows: Projection-Based Reduced-Order Modeling and Machine Learning*, vol. 1. Walter de Gruyter GmbH & Co KG, 2025.
- [10] D. Ryckelynck, F. Casenave, and N. Akkari, “Learning projection-based reduced-order models,” in *Manifold Learning: Model Reduction in Engineering*, pp. 9–37, Springer, 2024.
- [11] B. Kinch, B. Shaffer, E. Armstrong, M. Meehan, J. Hewson, and N. Trask, “Structure-preserving digital twins via conditional neural whitney forms,” *arXiv preprint arXiv:2508.06981*, 2025.

- [12] N. Trask, A. Huang, and X. Hu, “Enforcing exact physics in scientific machine learning: a data-driven exterior calculus on graphs,” *Journal of Computational Physics*, vol. 456, p. 110969, 2022.
- [13] J. A. Actor, X. Hu, A. Huang, S. A. Roberts, and N. Trask, “Data-driven whitney forms for structure-preserving control volume analysis,” *Journal of Computational Physics*, vol. 496, p. 112520, 2024.
- [14] N. Trask, “Discovery of structure-preserving finite element spaces for multiscale.,” tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2021.
- [15] W. M. Boon, N. R. Franco, A. Fumagalli, and P. Zunino, “Deep learning based reduced order modeling of darcy flow systems with local mass conservation,” *arXiv preprint arXiv:2311.14554*, 2023.
- [16] E. M. Dogo, O. Afolabi, N. Nwulu, B. Twala, and C. Aigbavboa, “A comparative analysis of gradient descent-based optimization algorithms on convolutional neural networks,” in *2018 international conference on computational techniques, electronics and mechanical systems (CTEMS)*, pp. 92–99, IEEE, 2018.
- [17] L. T. Biegler, “Optimization of chemical processes,” *American Institute of Chemical Engineers. AIChE Journal*, vol. 49, no. 1, p. 286, 2003.
- [18] A. Drud, “Conopt: A grg code for large sparse dynamic nonlinear optimization problems,” *Mathematical programming*, vol. 31, no. 2, pp. 153–191, 1985.
- [19] A. Wächter and L. T. Biegler, “Line search filter methods for nonlinear programming: Motivation and global convergence,” *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 1–31, 2005.
- [20] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [21] L. T. Biegler and V. M. Zavala, “Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization,” *Computers & Chemical Engineering*, vol. 33, no. 3, pp. 575–582, 2009.
- [22] Z. Shumaylov, P. Zaika, J. Rowbottom, F. Sherry, M. Weber, and C.-B. Schönlieb, “Lie algebra canonicalization: Equivariant neural operators under arbitrary lie groups,” *arXiv preprint arXiv:2410.02698*, 2024.
- [23] H. S. Venkataram, V. Constantinou, D. Wrench, A. Forouzani, and R. Watkins, “Super-resolution based topology optimization for rapid generation of low mass structural designs,” in *2024 IEEE Aerospace Conference*, pp. 1–10, IEEE, 2024.
- [24] P. K. Jha, “From theory to application: A practical introduction to neural operators in scientific computing,” *arXiv preprint arXiv:2503.05598*, 2025.
- [25] Y. Choi, G. Oxberry, D. White, and T. Kirchdoerfer, “Accelerating topology optimization using reduced order models,” tech. rep., Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States), 2019.
- [26] L. Gkimisis, N. Aretz, M. Tezzele, T. Richter, P. Benner, and K. E. Willcox, “Non-intrusive reduced-order modeling for dynamical systems with spatially localized features,” *arXiv preprint arXiv:2501.04400*, 2025.
- [27] A. Wächter, “Short tutorial: Getting started with ipopt in 90 minutes,” Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2009.