POLITECNICO
MILANO 1863

Burgers' Physics-Informed Surrogate leveraging Proper Orthogonal Decomposition

Numerical Analysis for Machine Learning, 2023/2024

Nuno Costa

February 14, 2024

Context

- Traditional numerical methods, while precise, can be computationally intensive and time-consuming, especially for **high-dimensional problems and complex geometries** [4].
- A particularly promising approach in recent years has been the integration of Proper Orthogonal Decomposition (POD) with neural networks to create lower-dimensional mappings that are computationally less demanding [2].
- In this project, we explore an innovative expansion of the POD-neural network surrogate model through the lens of Physics-Informed Neural Networks (PINNs) [5].

Proper Orthogonal Decomposition (POD), also known as Principal Component Analysis (PCA) in the context of statistics, is a mathematical technique used to reduce the dimensionality of a system while preserving its most significant features [6].

The POD process seeks to find an orthogonal basis in which this high-dimensional data can be represented succinctly.

By projecting the original data onto a **limited number of the eigenvectors** of that basis, we obtain a set of POD coefficients that provide a compressed representation of the system.

$$\mathbf{c}_\mu := \mathbf{V}^T \mathbf{u}_\mu \tag{1}$$

where:

- $\mathbf{V}$ is the matrix whose columns are the POD basis vectors
- $\mathbf{u}_\mu$ is the state of the system parameterized by μ

A neural network ϕ can be utilized to map new parameters μ to POD coefficients, leveraging their ability to learn complex mappings after computing the POD basis.

The predicted coefficients can then be used to reconstruct an approximation of the system's state using the POD basis:

$$\mathbf{u}_\mu^{ROM} := \mathbf{V}\phi(\mu) \approx \mathbf{u}_\mu \tag{2}$$

This hybrid approach, known as POD-NN, retains the advantages of dimensionality reduction from POD while utilizing the expressive power of neural networks to interpolate within the reduced space.

Physics-Informed Neural Network

The concept of Physics-Informed Neural Networks (PiNNs) revolutionized the integration of machine learning with physical sciences by enforcing the laws of physics directly within the neural network architecture [5].

PiNNs are designed to respect the fundamental principles that govern the behavior of physical systems. This is achieved by incorporating differential equations as regularization terms in the training process, ensuring that the network output not only fits the data but also aligns with known physical laws.

Let's consider the example of a damped harmonic oscillator:

$$m\frac{d^2u}{dt^2} + \mu\frac{du}{dt} + ku = 0 \tag{3}$$

In a PiNN, the network outputs $\hat{u}_\theta(t)$ to approximate positions, aiming to minimize the gap between predictions and actual physical behavior during training. This discrepancy, known as the **residual** $\mathcal{R}$, is given by:

$$\mathcal{R} = m\frac{d^2\hat{u}_\theta}{dt^2} + \mu\frac{d\hat{u}_\theta}{dt} + k\hat{u}_\theta \tag{4}$$

Ideally, $\mathcal{R}$ should be zero if $\hat{u}_\theta(t)$ accurately models the oscillator's motion.

Burgers' equation serves as a fundamental test bed due to its simplicity and the rich set of behaviors it can exhibit [3].

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0 \tag{5}$$

For this project, we defined $u(x,t) \in \Omega = (-1,1)$ with Neumann boundary conditions.

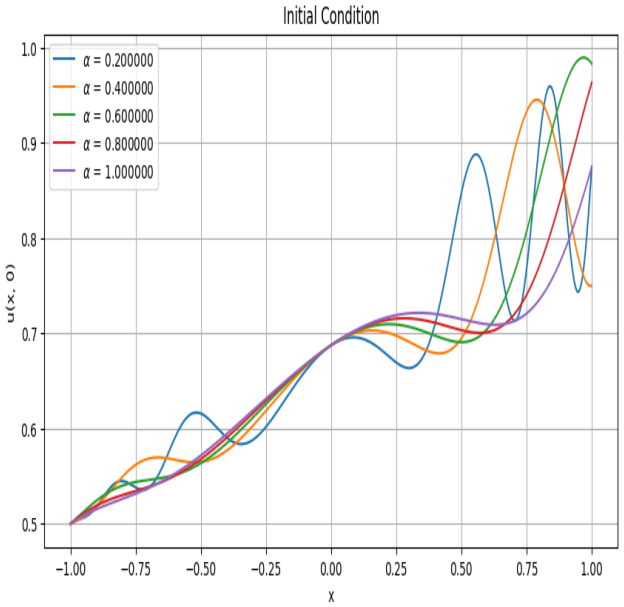

Figure: Initial Conditions with varying α

We chose the following initial conditions:

$$\lambda(x) = \left(\frac{(1 - \sin(\frac{\pi x^2}{\alpha}))}{16} + \frac{1}{8} \right) + 0.5 \quad (6)$$

$$u(x, 0) = \psi(x) = \max(x + 1, 0) \cdot \lambda(x) \quad (7)$$

with $x \in \Omega$.

Problem Setup - Insights

We can take the following insights from this setup:

- The chosen initial conditions ensure input velocity stability and the function's range between 0 and 1.
- The sinusoidal component adds variability, allowing comprehensive exploration of solution space, and we can assess sensitivity to time and α.
- The selected parameter space $\mu = [t, \alpha]$ spans $[0, 2] \times [0.2, 1]$, offering a wide range of scenarios for Burgers' equation.

We utilized *phiflow* [7] for high-fidelity data generation, simplifying the process, and ensuring physically accurate, computationally efficient model training.

Residual Definition

We can't define the residual in terms of differential equations as in a traditional PiNN directly since the operating network is working in a different space.

For our inviscid Burgers' equation, a different residual in the full space can be derived based on the equation's intrinsic property. For a given parameter vector μ and its corresponding reduced-space prediction $\mathbf{c}_\mu$, the full-space solution $\mathbf{u}_\mu$ must satisfy:

$$\mathbf{u}(\mathbf{x}, \mathbf{t}, \alpha) = \psi(\mathbf{x} - \mathbf{u}(\mathbf{x}, \mathbf{t}, \alpha) \cdot \mathbf{t}, \alpha) \tag{8}$$

The residual $\mathcal{R}$ for our problem is then:

$$\mathcal{R} = \mathbf{u}_\mu(\mathbf{x}) - \psi(\mathbf{x} - \mathbf{u}_\mu(\mathbf{x}) \cdot \mu_t, \mu_\alpha) \tag{9}$$

To integrate physics-informed constraints in the reduced space, the process involves predicting POD coefficients $\mathbf{c}$ with a neural network and projecting these coefficients to the full-dimensional physical space for obtaining the predicted solution $\mathbf{u}_{pred}$.

$$\mathbf{u}_{pred} = \mathbf{V}\mathbf{c}_{pred} \tag{10}$$

Given the nature of the inviscid Burgers' equation, the residual $\mathcal{R}$ in the full-dimensional space is derived by comparing the predicted solution $\mathbf{u}_{pred}$ against the solution obtained by the characteristic equation of the Burgers' equation:

$$\mathcal{R} = \mathbf{u}_{pred} - \psi(\mathbf{x} - \mathbf{u}_{pred} \cdot \mu_t, \mu_\alpha) \tag{11}$$

Loss Definition

For our ROM framework, the loss function is a combination of the mean squared error (MSE) between the predicted and target POD coefficients and the physics-informed residual:

$$\mathcal{L}(\theta) = \sum_i (\mathbf{c}_\theta(\mu_i) - \mathbf{c}_i^*)^2 + \alpha_0 \sum_j \mathcal{R}(\mu_j)^2 \tag{12}$$

where $\mathbf{c}_i^*$ are the target POD coefficients, $\mathbf{c}_\theta(t_i)$ are the predicted POD coefficients by the neural network, and α_0 is a hyperparameter.

Results: Considerations

For this project, we:

- Selected 20 POD modes for computational efficiency and essential dynamics capture of Burgers' equation, which set the minimum reconstruction error at 0.03%.
- The minimum possible residual was set at 0.0007 due to simulation errors of *phiflow*.
- Defined the neural network structure with one hidden layer with 100 neurons (ReLU activation), utilizing the *dlroms* package [1].

Results: Training Time

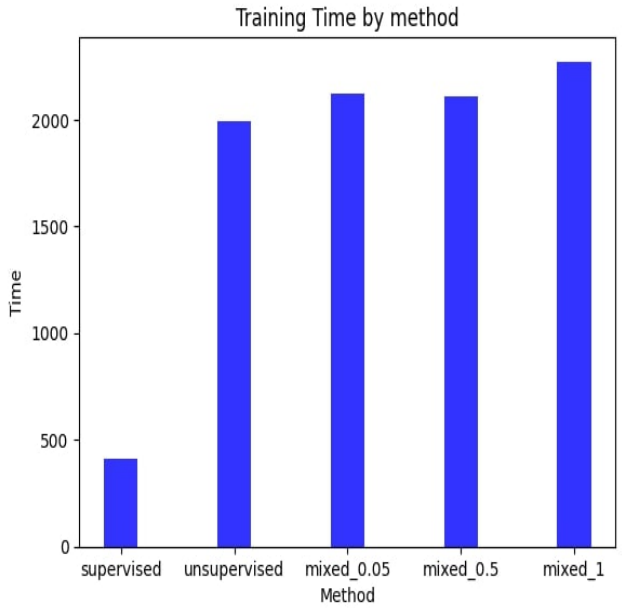

Figure: Training Times of POD-PiNN

- Clear disparity between supervised and other methods that integrate residual calculations.
- Unsupervised method shows slightly faster training times than mixed methods, as it does not compute MSE term.
- Variability in training times among mixed methods is attributed to computational processing differences without a clear conclusion on the impact of different weights.

Results: Accuracy

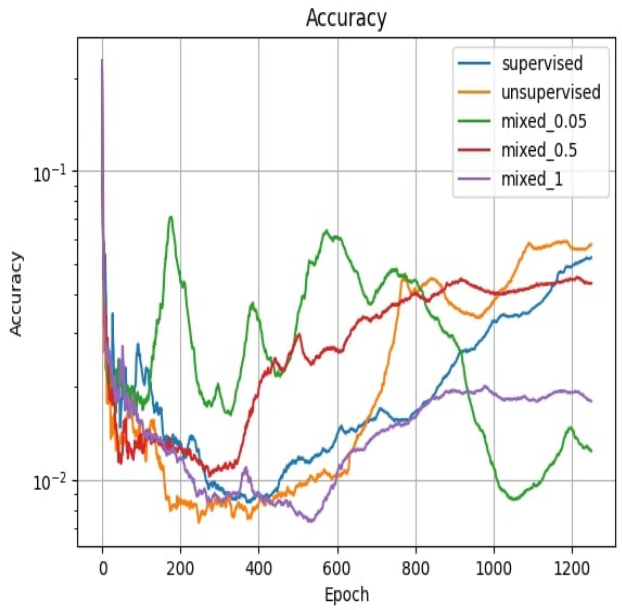

Figure: Validation Accuracy of POD-PiNN

- Supervised method quickly reaches an accuracy plateau, indicating early convergence at around 400 epochs.
- Mixed approach with $\alpha_0 = 1$ achieves the highest accuracy, suggesting heavier physical regularization weight improves physics representation accuracy.
- Mixed approach with $\alpha_0 = 0.05$ shows erratic progress, indicating potential for further accuracy improvements beyond observed epochs.

Results: Residual

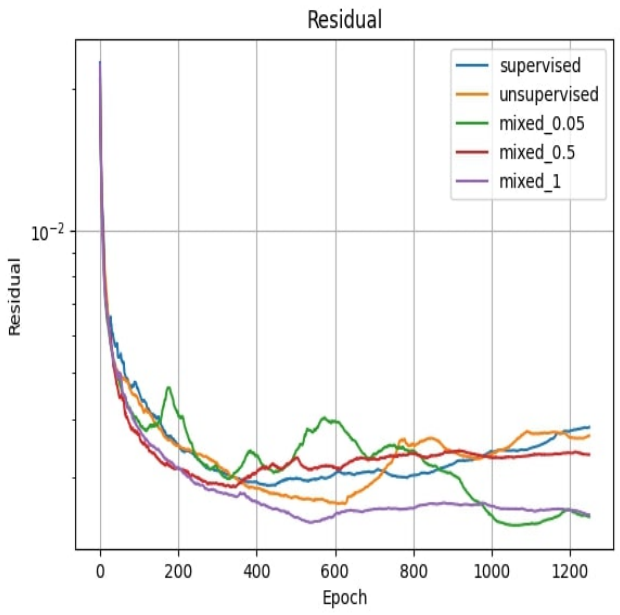

Figure: Residual Error of POD-PiNN

- Supervised method's residual error declines initially, reaching a minimum around the 400th epoch, then increases, indicating possible overfitting.
- Mixed method with $\alpha_0 = 1$ achieves minimum residual error swiftly and maintains a decreasing trend, indicating effective physical dynamics learning.
- Mixed method with $\alpha = 0.05$ exhibits fluctuations, highlighting the optimization challenges in balancing data fidelity and physical law adherence.

Final Remarks

- Embedding physical knowledge into neural network training enhances model fidelity but lengthens training times.
- Mixed-method approaches with optimal physics-regularizer weighting balance accuracy and physical fidelity effectively.
- We demonstrated the importance of domain knowledge in developing machine learning models for scientific applications like fluid dynamics.
- Potential for post-processing pipeline development to refine PiNN model predictions using the inviscid Burgers' Equation properties.
- Such a pipeline could improve ROM utility by correcting model fidelity issues, making predictions more accurate and physically trustworthy.

References I

N. R. Franco.
DIroms: a python package for deep learning based reduced order models.
https://github.com/NicolaRFranco/dlroms, 2023.

C. Gräßle, M. Hinze, and S. Volkwein.
Model order reduction by proper orthogonal decomposition, 2020.

P. J. Olver.
Introduction to partial differential equations.
Springer, 2020.

References II


A. Quarteroni, R. Sacco, and F. Saleri.
Numerical Mathematics.
Springer Science & Business Media, Oct. 2006.


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.
Journal of Computational Physics, 378:686–707, Feb. 2019.


G. Strang.
Linear Algebra and learning from data.
Wellesley-Cambridge Press, 2019.


TUM.
phiflow.
https://github.com/tum-pbs/PhiFlow, 2023.