Reduced-Order Modeling of Compressible Flows Using Supervised Dimensionality Reduction

Abhijnan Dikshit[1[0000−0002−8745−6873]], Leifur Leifsson[1[0000−0001−5134−870X]], Slawomir Koziel[2,3[0000−0002−9063−2647]], and Anna Pietrenko-Dabrowska[3[0000−0003−2319−6782]]

[1] School of Aeronautics and Astronautics, Purdue University, West Lafayette, Indiana 47907, USA
{adikshit@purdue.edu, leifur@purdue.edu}

[2] Engineering Optimization & Modeling Center, Department of Engineering, Reykjavík University, Menntavegur 1, 102 Reykjavík, Iceland
koziel@ru.is

[3] Faculty of Electronics Telecommunications and Informatics, Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdansk, Poland
anna.dabrowska@pg.edu.pl

Abstract. Data-driven reduced-order modeling (ROM) methods are widely used in aerodynamic flow modeling. They are mainly used to predict distributed quantities obtained from high-fidelity simulations such as surface pressure distributions and flow fields. However, such models only consider the distributed quantities and do not consider the data of the aerodynamic coefficients which are also available from high-fidelity simulations. This work proposes a novel supervised ROM architecture that learns from both distributed quantities and aerodynamic coefficients. The proposed model is characterized using a transonic airfoil modeling problem and compared with a standard ROM and neural network model. The results demonstrate that the supervised ROM architecture can outperform a standard neural network by 12% in predicting aerodynamic coefficients when only using 25 training samples. Even with 300 training samples, the supervised ROM can outperform the neural network by 1 or 2%. This can be achieved while maintaining the same level of accuracy as a standard ROM in predicting airfoil surface pressure distributions. This demonstrates that the proposed model can lead to sample-efficient aerodynamic modeling by reducing computational cost and enhancing model accuracy.

Keywords: Reduced-order modeling · Multi-task learning · Autoencoders · Deep learning · Compressible flows · Aerodynamics

1 Introduction

Aerodynamic modeling has been a cornerstone of the design process in the fields of aerospace and automotive engineering. Traditionally, high-fidelity numerical methods, such as computational fluid dynamics (CFD), have been used extensively to evaluate aerodynamic quantities of interest (QoI), such as flow fields

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

and surface distributions. However, the usage of CFD methods for aerodynamic modeling and design can be prohibited by the high computational cost of these methods. To mitigate the issue of computational cost, surrogate modeling methods have been utilized to create cheap-to-evaluate models based on data obtained from high-fidelity numerical simulations [24].

Data-driven reduced order modeling (ROM) methods are a popular surrogate modeling strategy for aerodynamics [4]. ROMs provide a parametric mapping between the inputs and outputs of a high-fidelity simulation that is efficient and cheap to evaluate. These models are primarily used to predict high-dimensional aerodynamic data, such as surface pressure distributions and flow fields [12]. These methods are also non-intrusive and do not require the original simulation code to be modified for implementation. They can be implemented using open-source machine learning packages, further enhancing the appeal of such modeling methods. ROMs typically work by using a dimensionality reduction method to transform the high-dimensional aerodynamic data into a latent space. The latent space is then modeled using an interpolation or regression model, with the parameter space used to generate the data used as an input.

Proper orthogonal decomposition (POD) has been used extensively to create ROMs and the effectiveness of the method has been proven in the case of aerodynamic modeling [14, 3]. However, previous studies have shown that POD is unable to capture the correct trends in compressible flow data with nonlinearities such as shock waves [24]. To increase the accuracy of ROMs in compressible flow, manifold learning and deep learning have been used. Decker et al. [4] and Iyengar et al. [13] developed ROMs using manifold learning methods and demonstrated them on flows in the compressible flow regime. However, the main drawback of manifold learning methods is that it is necessary to formulate and solve an optimization problem to reconstruct the high-dimensional solution from the latent space of the model [7]. As an alternative, deep learning methods were considered to create nonlinear models with an efficient means to reconstruct the high-dimensional solution. Wang et al. [27] used a variational autoencoder ROM to successfully model compressible flow fields around different airfoil shapes. Halder et al. [10] combined a convolutional autoencoder and a Gaussian process (GP) model to predict lid-driven cavity flows.

While the literature on ROMs has steadily grown in the past, such models ignore the integrated quantities, such as lift and drag coefficients, that are also available as part of the data generated from CFD simulations. Due to this, the use of ROMs has been limited in aerodynamic design work. This is because while the lift and drag coefficients can be computed from flow fields or surface quantities via numerical integration, the results may not be accurate enough to perform design work [19]. Including the data of the aerodynamic coefficients in the modeling architecture will enhance the applicability of such models to aerodynamic design and may also improve the generalizability and accuracy of the predictions.

To that end, this work explores a supervised dimensionality reduction method [18] that creates a latent space that accounts for the lift and drag coefficients. Di-

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

mensionality reduction methods used in previous ROM architectures, like most dimensionality reduction methods, are unsupervised. This means that the data used to train the dimensionality reduction method has no labels. On the other hand, a supervised method will utilize these labels in the model to enhance accuracy and generalization. In this way, this work proposes a novel nonlinear ROM architecture that learns from both aerodynamic coefficient data and surface pressure distribution data by a supervised encoding generated through an autoencoder and the corresponding loss function. The use of the autoencoder method is motivated by the ease of generating the latent space and reconstructing the high-dimensional data with a simple forward pass of an NN model. The latent space is modeled using a multi-task GP (MTGP) model [2] for enhanced prediction accuracy. The characteristics of the novel model are showcased on a transonic airfoil modeling problem. This demonstrates the capabilities of the new ROM architecture in the context of compressible flows.

These contributions produce a ROM that can combine multiple forms of data for enhanced surrogate modeling of aerodynamic flows. The remainder of the paper is organized as follows. Section 2 will describe the novel ROM architecture proposed in this work while Section 3 will describe benchmarking methods used as points of comparison. Section 4 provides a description of the transonic flow test problem and the results obtained for the problem. Finally, Section 5 highlights the main conclusions obtained in this work.

2 Supervised Reduced Order Modeling

This section introduces the novel reduced-order modeling architecture that uses a supervised embedding of high-dimensional aerodynamic data. In this work, high-dimensional aerodynamic data consists of the distributed surface pressure and skin friction values. The low-dimensional embedding is modeled using a multi-task GP model. The proposed architecture enables fast and accurate prediction of aerodynamic fields and coefficients. A graphical representation of the proposed modeling architecture is shown in Fig. 1.

2.1 Supervised embedding procedure

The novelty of the modeling procedure proposed in this work lies in the creation of a supervised embedding of the high-dimensional aerodynamic data. The supervised nature of the embedding will allow the model to use both aerodynamic coefficients and field data to increase the accuracy of the model. The embedding procedure is carried out using deep neural networks in the form of a supervised autoencoder [18]. Autoencoder neural networks [11] are a special type of a neural network model that create a nonlinear embedding of high-dimensional data.

The neural network architecture consists of two separate neural networks, an encoder and a decoder network. The encoder neural network, f_{enc}, compresses the high-dimensional data, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n] \in \mathbb{R}^{n \times m}$, to a low-dimensional

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

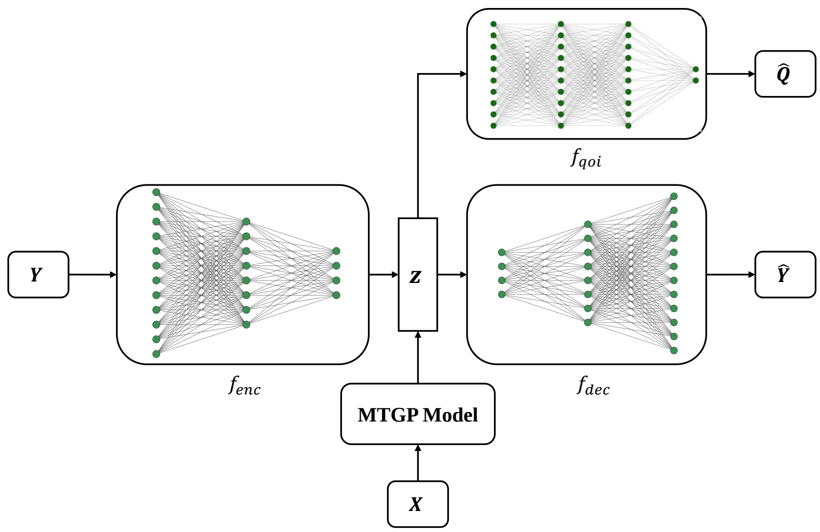


Fig. 1: Graphical representation of reduced order model with supervised embedding architecture proposed in this work.

representation, $\mathbf{z} \in \mathbb{R}^{n \times k}$, where k is much smaller than the original dimensionality m. The decoder network, f_{dec}, then projects the latent vector $\mathbf{z}$ to the original high-dimensional state to obtain a reconstruction of the high-dimensional data, $\hat{\mathbf{Y}}$. In this way, the autoencoder can create a low-dimensional representation of the original high-dimensional data. It can also regenerate the high-dimensional data from a given latent vector.

To enable a supervised embedding of the high-dimensional data, a subnetwork, f_{qoi}, is added to the autoencoder which takes the latent vector and transforms it into the supervised labels of the problem. In this case, the labels are the aerodynamic coefficients, i.e., lift and drag coefficients which are denoted by $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_n] \in \mathbb{R}^{n \times 2}$. This is done as the aerodynamic coefficients are directly correlated with the aerodynamic surface variables, i.e., the surface pressure and skin friction. By adding the aerodynamic coefficients to the autoencoder model, a multi-task model is created that can obtain better generalization and accuracy for each task considered by predicting correlated sets of data [30].

The hyperparameters of the neural network used for the numerical experiments performed in this work are shown in Table 1. These can different for other flow modeling problems of interest. A fully connected symmetric architecture is used for the autoencoder. Each layer of the encoder uses the SiLU activation function [6] to introduce nonlinearity in the dimensionality reduction process. The neural network is implemented using Pytorch [21].

The loss function of the autoencoder can be written as the sum of two losses. The first loss is the mean squared error (MSE) between the original high-dimensional data and the reconstruction of the autoencoder. The second loss

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

Table 1: Hyperparameter selection for the supervised autoencoder.

Hyperparameter	**Value used**
f_{enc} Hidden Layer Neurons	[512, 128, 64]
f_{dec} Hidden Layer Neurons	[64, 128, 512]
f_{qoi} Hidden Layer Neurons	[32, 16, 8]
Latent dimension	20
Activation function	SiLU
Learning Rate	1e-4
Number of Epochs	10000

function is the MSE between the original and predicted aerodynamic coefficients. The combined loss function can then be written as

$$\mathcal{L} = \mathrm{w}\frac{1}{N}\sum_{m=1}^{N}(\hat{\mathbf{Y}}^{(m)} - \mathbf{Y}^{(m)})^2 + (1-\mathrm{w})\frac{1}{N}\sum_{m=1}^{N}(\hat{\mathbf{Q}}^{(m)} - \mathbf{Q}^{(m)})^2, \tag{1}$$

where w determines the weighting of each term in the loss function. The loss function is optimized using the adaptive moments (ADAM) optimization algorithm [16], and the gradients are computed using backpropagation [23].

Once the autoencoder is trained, high-dimensional aerodynamic predictions can be made for new input parameters using the corresponding latent vector. However, this latent vector is not known for a new set of input parameters. It is, therefore, necessary to create a parametric mapping between the input space and the latent space to be able to obtain the latent vector at a new design point.

2.2 Latent space interpolation

The parametric mapping between the parameter space and the latent space is created using a MTGP model. This model will use the parameter space $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{n\times p}$ as inputs and the latent space vector $\mathbf{z}$ as the outputs. MTGPs are multi-task models created in the context of Gaussian processes. MTGP models learn to predict multiple related outputs simultaneously. The model not only learns the outputs but also the correlation between the outputs. This allows the outputs to be modeled using a multivariate Gaussian distribution with the correlation between the outputs providing a boost in the prediction capability of the model.

In this work, the free-form parameterization formulation of MTGPs [2] is used. In this formulation, the correlation between the outputs is treated as an additional hyperparameter that is optimized during the training process of the MTGP. The models are implemented using GPyTorch [8] and BoTorch [1], Python packages for creating GP models and training them using GPU acceleration. The radial basis kernel function was used to train the models and the optimization of the hyperparameters was carried out using the maximum log-likelihood criterion [22].

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

3 Benchmarking Methods

This section briefly describes the benchmarking modeling methods that are used as a point of comparison for the model architecture proposed in this work. The methods cover modeling approaches that consider surface quantities or aerodynamic coefficients alone.

3.1 Unsupervised ROM architecture

The first benchmark modeling method that is used in this work is the conventional ROM architecture that utilizes an unsupervised autoencoder method, similar to a previously used deep learning ROM [5], to make aerodynamic predictions. For a fair comparison, the unsupervised embedding is created using a fully connected autoencoder, following much of the same procedure as described in Section 2.1. The only difference is that the autoencoder does not contain a subnetwork that predicts the aerodynamic coefficients, making it a completely unsupervised dimensionality reduction process. In this case, the loss function only contains the first term in (1). The optimization and implementation procedure is the same. The architecture and hyperparameters of the encoder and decoder network are also the same as shown in Table 1.

The latent space of the unsupervised ROM is also modeled using a MTGP model with the same implementation as the supervised ROM proposed in this work. Maintaining these similarities in the implementations will isolate the effect of the supervised embedding on the performance of the ROM. This will provide insight into the performance benefits of the supervised embedding.

3.2 Standard neural network model

Another comparison that must be made in this study is the comparison to a standard NN model that has been previously used to predict lift and drag coefficients for a given set of inputs [28]. The standard NN model directly predicts the aerodynamic coefficients. This will be a multi-output NN model where the output layer contains two outputs, corresponding to the lift, and drag coefficients. The input layer of the NN model contains the parameter space of the problem given by $\mathbf{X}$. The architecture of the NN model is the same as the f_{qoi} subnetwork that is described for the supervised ROM in Section 2.1. Each layer uses the SiLU activation function [6] and the NN model is trained for 10000 epochs. The ADAM optimization algorithm [16] is used along with backpropagation [23] to calculate the gradients. In this case, the loss function for the NN model contains the second term of (1) which represents the MSE between the original and predicted aerodynamic coefficients.

4 Numerical Experiments

This section describes the numerical experiments conducted to characterize the proposed supervised ROM. The test case used in this work is an airfoil modeling problem.

DOI: 10.1007/978-3-031-97557-8_29

4.1 Problem setup

In this test case, the surface pressure distribution (C_p) of airfoils is used as a distributed aerodynamic quantity with the lift (C_l) and drag coefficients (C_d) being the supervised labels of the autoencoder. Figure 2 shows the RAE 2822 airfoil, force coefficients, and a typical pressure distribution of the airfoil. The figure indicates the non-dimensionalized lift and drag force which act perpendicular to and along the direction of the incoming freestream flow, respectively. The aerodynamic coefficients, C_l and C_d, are set as the output $\mathbf{Q}$ of the ROM while the entire set of pressure distribution values, C_p, is set as the output $\mathbf{Y}$ of the ROM. The airfoil is assumed to be operating in a flow with a Mach number (M_∞) of 0.734 and a Reynolds number of 6.5×10^6. The parameter space of the problem contains the shape variables describing the airfoil shape as well as the angle of attack (α) of the airfoil shown in Fig. 2. The blue shaded region around the airfoil in Fig. 2 indicates the bounds of the airfoil shape variables with RAE 2822 as the baseline shape. Twelve class shape transformation (CST) variables [17] are used to describe the shape of the airfoil. The angle of attack varies between $-3°$ and $5°$. Figure 2 also indicates the shock location and shock strength on the surface pressure distribution. The shock location is a point of sudden rise in pressure on the airfoil surface because of local acceleration of flow. The rise in pressure that occurs at this point is called the shock strength.

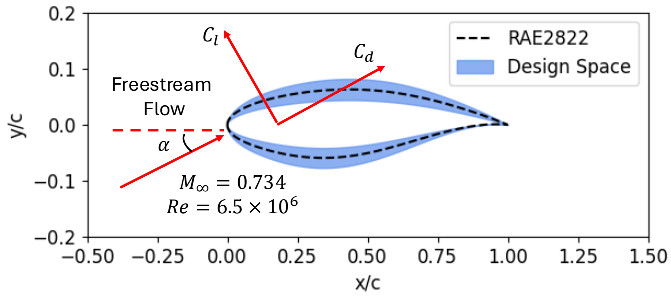
The data for this test problem is generated through a design of experiments created using Latin Hypercube sampling (LHS) [15]. The number of samples in the training data varies between 25 and 300 samples. There is a separate testing data set of 75 samples that is also generated through LHS.

4.2 Computational modeling

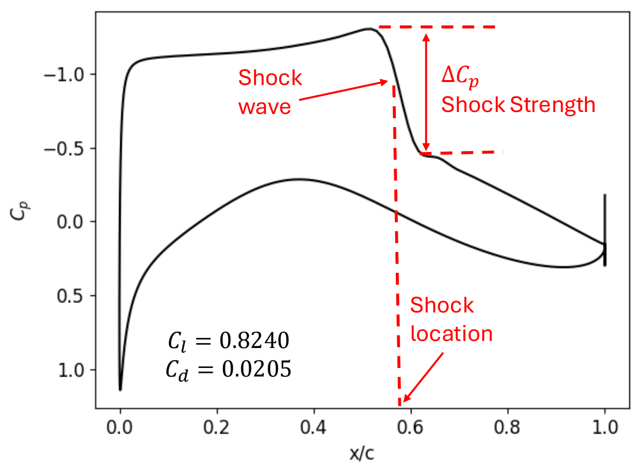
The high-fidelity aerodynamic data of various airfoil shapes at different angles of attack was generated using ADFlow [20]. The solver uses a combination of the Approximate Newton-Krylov method [29] and the full Newton-Krylov method to solve the steady compressible Reynolds-averaged Navier Stokes (RANS) equations with the Spalart-Allmaras turbulence model [26]. The grid topology used in this work is an O-grid topology shown in Fig. 3. The grid is created by extruding an airfoil surface mesh using the hyperbolic marching method implemented using pyHyp [25]. The geometry manipulation and airfoil parameterization are implemented using pyGeo [9], an open-source CAD-free geometry framework. For every new geometry, pyGeo is used to alter the design variables, and the mesh is regenerated once the new airfoil shape has been generated.

To balance accuracy and computational cost, a mesh convergence study is performed as shown in Table 2. The mesh convergence study was conducted at the designated flow conditions and a fixed lift coefficient of 0.824. The computations were run in parallel on 64 processors using a high-performance computing cluster. For the airfoil computations performed in this work, the L1 mesh is chosen as refining the mesh further does not significantly change the value of the drag coefficient and the angle of attack for achieving a lift coefficient of 0.824. This mesh will balance accuracy and computational cost.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29



(a)



(b)

Fig. 2: The problem setup with (a) the RAE 2822 airfoil and the design space, and (b) its surface pressure distribution.

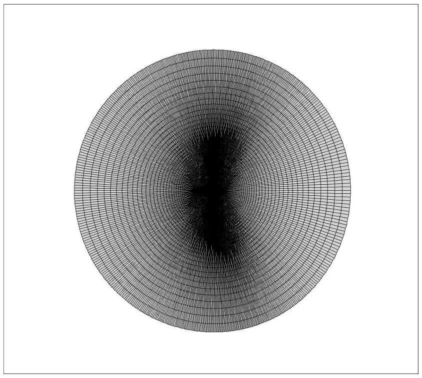
4.3 Model evaluation metrics

The performance of the proposed and benchmark modeling methods is evaluated using the metric defined in this section. To evaluate the prediction of the distributed airfoil surface quantities, the mean relative error will be used as a performance metric. The mean relative error is computed as

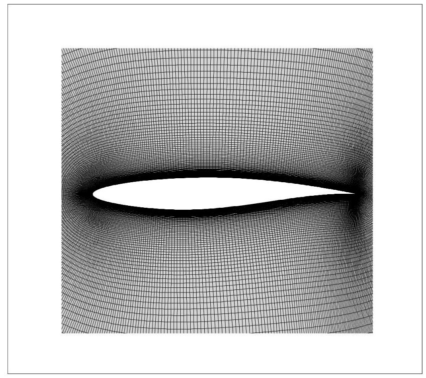
$$e_{rel} = \frac{1}{N} \sum_{i=1}^{N} \frac{||\hat{\mathbf{Y}} - \mathbf{Y}||_2}{||\mathbf{Y}||_2}, \tag{2}$$

where $\hat{\mathbf{Y}}$ is the prediction, $\mathbf{Y}$ is the CFD solution and N is the number of testing samples.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29



(a)



(b)

Fig. 3: Mesh topology for aerodynamic data generation (a) in the far-field, and (b) near the airfoil surface.

Table 2: Mesh Convergence study of RAE2822 airfoil.

Level of Mesh	Size of Mesh	C_d (10^{-4})	C_l	α (deg.)
L0	256,000	205.12	0.824	2.93
L1	128,000	205.15	0.824	2.95
L2	32,000	214.86	0.824	3.05

The prediction of aerodynamic coefficients is assessed using the normalized root mean squared error (NRMSE) metric. Although, the models used in this work produce both the lift and drag coefficient simultaneously as two outputs, the NRMSE of each will be calculated independently. The NRMSE of the model prediction can be obtained as

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}}{\max(y) - \min(y)}, \tag{3}$$

where y is the C_l or C_d and $\hat{y}$ is the prediction of the C_l or C_d. The RMSE is normalized using the range of values of the C_l or C_d in the testing dataset and this is calculated as $\max(y) - \min(y)$.

4.4 Results

Figure 4 shows the variation of the mean relative error of the testing dataset with the number of training samples. The results illustrate that the mean relative error of the supervised ROM architecture is dependent on the value of the weight in (1) and the value of the weight is varied to demonstrate this variation in performance. Increasing the value of the weight in (1) improves the performance of the supervised ROM architecture. Once the value of the weight in (1)

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

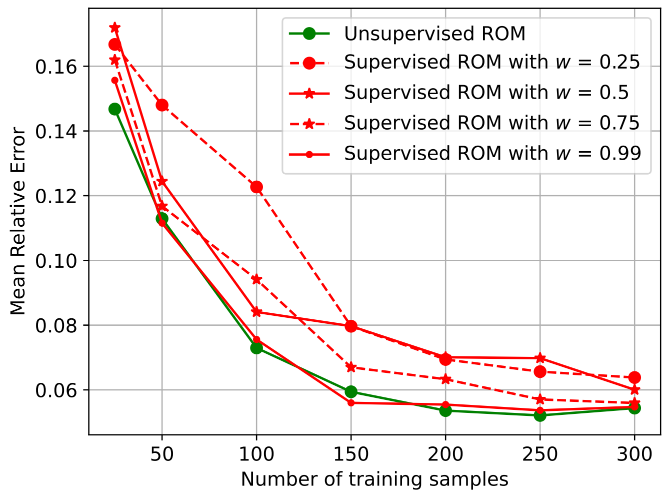


Fig. 4: Mean relative error versus number of training samples for the prediction of surface pressure distribution (C_p).

is high enough, it can match the performance of an unsupervised ROM architecture. With the current setup and architecture, the supervised ROM can match the performance of an unsupervised ROM, however, the supervised ROM still provides the benefit of simultaneously predicting the aerodynamic coefficients.

To assess the prediction of lift coefficients, Fig. 5 shows a plot of the variation of NRMSE values with the number of training samples. The supervised ROM architecture achieves a NRMSE of approximately 3.5% with 25 samples and approximately 1% with 300 samples. On the other hand, the standard neural network achieves a NRMSE of approximately 15.5% with 25 samples and approximately 2.1% with 300 samples. The results illustrate that the supervised ROM architecture achieves a significant improvement in the prediction of the lift coefficient over the standard neural network architecture. The supervised ROM is also more sample efficient which means that it has a lower NRMSE value with lower number of training samples. The supervised ROM improves the NRMSE by almost 12% for a sample size of 25. Eventually, as the number of training samples increase the performance of the two models becomes similar, but the supervised ROM still has slightly better performance. A similar trend can be observed for the drag coefficient in Fig. 6. The supervised ROM achieves an NRMSE of approximately 8% at 25 samples which is a 12% improvement over the standard neural network. At 300 samples, the best performance of the supervised ROM provides a 2% benefit over the standard neural network architecture. The prediction of the drag coefficient does seem to be affected by the value of the weighting of each term in (1). As the weighting of the pressure prediction is

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

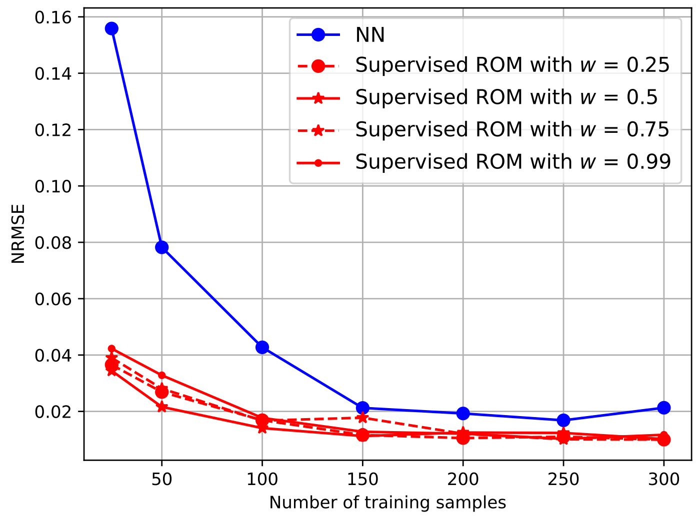


Fig. 5: NRMSE versus number of training samples for the prediction of lift coefficient (C_l).

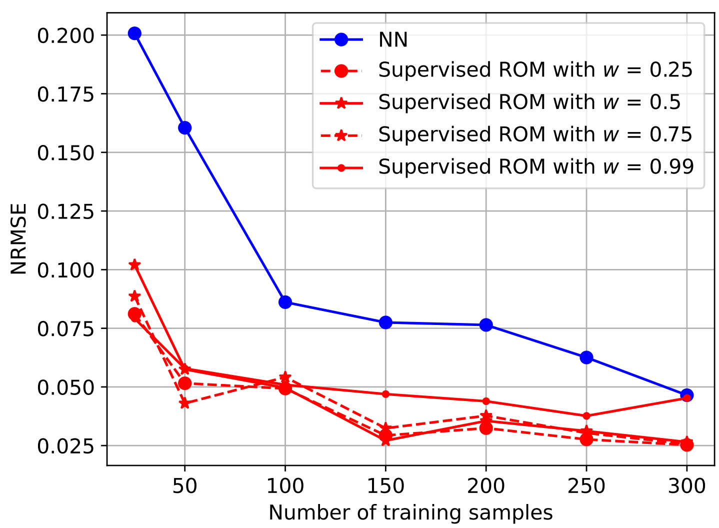


Fig. 6: NRMSE versus number of training samples for the prediction of drag coefficient (C_d).

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

increased, the NRMSE of the drag coefficient prediction increases. Even accounting for the variations caused by the weighting, the supervised ROM architecture can outperform a standard neural network architecture in the prediction of the drag coefficient. This demonstrates the superiority of the multi-task learning paradigm that is introduced in the supervised ROM architecture.

To demonstrate the prediction of the ROM architectures, Fig. 7 shows the prediction of the surface pressure distribution of two test airfoils with a shock wave occurring on the upper surface. The prediction results show that the ROM architectures can predict the pressure distribution of the airfoil and the shock location and shock strength to a high degree of accuracy. As expected, the supervised ROM trained with a weighting of 0.99 generally captures the pressure distribution better than a ROM trained with a weighting of 0.25.

5 Conclusion

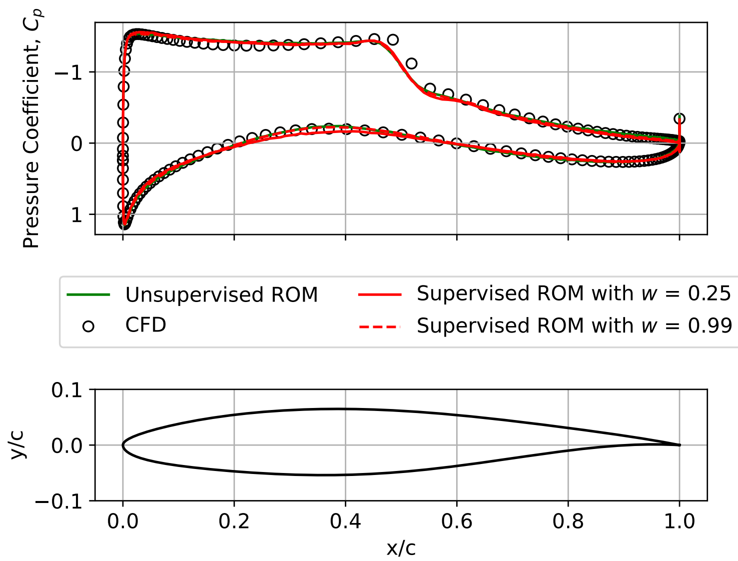
This work proposes a supervised ROM architecture that uses a supervised embedding procedure to embed high-dimensional aerodynamic flow information into a latent space that is informed using supervised labels. This introduces a multi-task formulation into the ROM architecture that will improve the accuracy of the model. A parametric map is created between the parameter space and the latent space using an MTGP model to enable predictions for new design points in the parameter space.

The proposed model is demonstrated on a transonic airfoil modeling problem. The supervised ROM architecture shows significant improvements in the prediction of the lift and drag coefficients of the airfoil and outperforms a standard neural network architecture designed to predict only the coefficients. Upon evaluating the prediction of the pressure distributions, it was found that the supervised ROM architecture can match the performance of the unsupervised ROM architecture. In general, a higher weighting must be given to the quantity that is more difficult to predict. In this case, the pressure distribution is more difficult to predict and a higher weighting must, therefore, be given to the pressure distribution to ensure sufficient accuracy. The prediction of the aerodynamic coefficients is less sensitive to the value of the weighting.

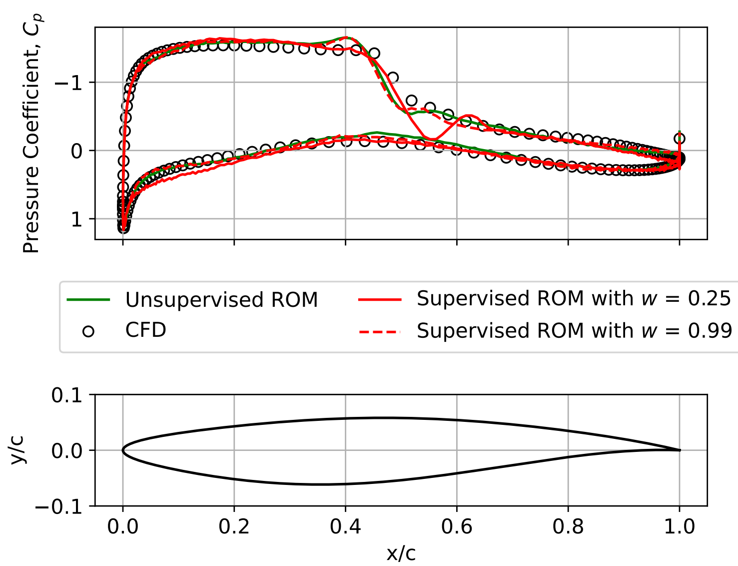
Future work will be dedicated to fine-tuning the parameters of the architecture and exploring different formulations to further improve the architecture. Efforts will also be dedicated to creating optimization loops that incorporate the supervised ROM architecture to perform constrained aerodynamic shape optimization. This will demonstrate the use of such supervised ROM in important aerodynamic design tasks.

Acknowledgments

The work done in this paper was supported in part by the U.S. National Science Foundation (NSF) award number 2223732 and the Icelandic Centre for Research (RANNIS) award number 239858.



(a)



(b)

Fig. 7: Prediction of surface pressure distributions for (a) a test airfoil at $\alpha = 4.733°$ and (b) a test airfoil at $\alpha = 3.667°$ using an unsupervised ROM and the proposed supervised ROM.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

References

1. Balandat, M., Karrer, B., Jiang, D.R., Daulton, S., Letham, B., Wilson, A.G., Bakshy, E.: Botorch: A framework for efficient monte-carlo bayesian optimization. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. p. 21524–21538. Vancouver, Canada (6-12 December, 2020)
2. Bonilla, E.V., Chai, K., Williams, C.: Multi-task gaussian process prediction. In: Advances in Neural Information Processing Systems. vol. 20. Vancouver, Canada (3-6 December, 2007)
3. Chang, C.M., Tsai, H.Y., Chiu, T.Y., Cheng, C.J., Lin, W.A., Chen, R.B., Chou, Y.J.: Proper-orthogonal-decomposition–based emulation of spatiotemporal evolution of turbulent wakes. AIAA Journal pp. 1–13 (3 2025). https://doi.org/10.2514/1.J064779
4. Decker, K., Iyengar, N., Rajaram, D., Perron, C., Mavris, D.: Manifold alignment-based nonintrusive and nonlinear multifidelity reduced-order modeling. AIAA Journal **61**, 454–474 (1 2023). https://doi.org/10.2514/1.J061720
5. Dikshit, A., Leifsson, L.: Efficient design of transonic airfoils using non-intrusive reduced order models and composite bayesian optimization. In: AIAA Aviation Forum and ASCEND, 2024. Las Vegas, NV (30 Jul–02 Aug 2024). https://doi.org/10.2514/6.2024-3666
6. Elfwing, S., Uchibe, E., Doya, K.: Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural Networks **107**, 3–11 (11 2018). https://doi.org/10.1016/j.neunet.2017.12.012
7. Franz, T., Zimmermann, R., Görtz, S., Karcher, N.: Interpolation-based reduced-order modelling for steady transonic flows via manifold learning. International Journal of Computational Fluid Dynamics **28**, 106–121 (3 2014). https://doi.org/10.1080/10618562.2014.918695
8. Gardner, J.R., Pleiss, G., Bindel, D., Weinberger, K.Q., Wilson, A.G.: Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. pp. 7587–7597. Montreal, Canada (3-8 December, 2018)
9. Hajdik, H.M., Yildirim, A., Wu, N., Brelje, B.J., Seraj, S., Mangano, M., Anibal, J.L., Jonsson, E., Adler, E.J., Mader, C.A., Kenway, G.K.W., Martins, J.R.R.A.: pyGeo: A geometry package for multidisciplinary design optimization. Journal of Open Source Software **8**(87), 5319 (2023). https://doi.org/10.21105/joss.05319
10. Halder, R., Fidkowski, K.J., Maki, K.J.: Non-intrusive reduced-order modeling using convolutional autoencoders. International Journal for Numerical Methods in Engineering **123**, 5369–5390 (11 2022). https://doi.org/10.1002/nme.7072
11. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006). https://doi.org/10.1126/science.1127647
12. Iyengar, N., Rajaram, D., Mavris, D.: Uncertainty quantification in flows with discontinuities: A probabilistic approach on nonlinear manifolds. In: AIAA Aviation and Aeronautics Forum and Exposition, 2023. San Diego, CA (12–16 Jun 2023). https://doi.org/10.2514/6.2023-4092
13. Iyengar, N., Rajaram, D., Mavris, D.: Domain decomposition strategy for combining nonlinear and linear reduced-order models. AIAA Journal **62**, 1375–1389 (2024). https://doi.org/10.2514/1.J063361
14. Jang, B., Lee, W., Lee, J.J., Jin, H.: Artificial neural network-based temperature prediction of a lunar orbiter in thermal vacuum test: Data-driven

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29

reduced-order models. Aerospace Science and Technology **145** (2 2024). https://doi.org/10.1016/j.ast.2023.108867
15. Jin, R., Chen, W., Sudjianto, A.: An efficient algorithm for constructing optimal design of computer experiments. Journal of Statistical Planning and Inference **134**, 268–287 (9 2005). https://doi.org/10.1016/j.jspi.2004.02.014
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (ICLR). San Diego, CA (12 May 7 - 9, 2015)
17. Kulfan, B.M.: Universal parametric geometry representation method. Journal of Aircraft **45**, 142–158 (2008). https://doi.org/10.2514/1.29958
18. Le, L., Patterson, A., White, M.: Supervised autoencoders: Improving generalization performance with unsupervised regularizers. In: Advances in Neural Information Processing Systems. vol. 31. Montreal, Canada (2–8 Dec 2018)
19. Li, J., Du, X., Martins, J.R.: Machine learning in aerodynamic shape optimization. Progress in Aerospace Sciences **134** (10 2022). https://doi.org/10.1016/j.paerosci.2022.100849
20. Mader, C.A., Kenway, G.K.W., Yildirim, A., Martins, J.R.R.A.: ADflow: An open-source computational fluid dynamics solver for aerodynamic and multidisciplinary optimization. Journal of Aerospace Information Systems **17**(9), 508–527 (Sep 2020). https://doi.org/10.2514/1.I010796
21. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Xamla, A.K., Yang, E., Devito, Z., Nabla, M.R., Tejani, A., Chilamkurthy, S., Ai, Q., Steiner, B., Bai, L.F.J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: 33rd Conference on Neural Information Processing Systems (NeurIPS 2019). Vancouver, Canada (8-14 December, 2019)
22. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA (2005)
23. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**, 533–536 (1986). https://doi.org/https://doi.org/10.1038/323533a0
24. Sabater, C., Stürmer, P., Bekemeyer, P.: Fast predictions of aircraft aerodynamics using deep-learning techniques. AIAA Journal **60**, 5249–5261 (9 2022). https://doi.org/10.2514/1.J061234
25. Secco, N., Kenway, G.K.W., He, P., Mader, C.A., Martins, J.R.R.A.: Efficient mesh generation and deformation for aerodynamic shape optimization. AIAA Journal **59**, 1151–1168 (2021). https://doi.org/10.2514/1.J059491
26. Spalart, P.R., Allmaras, S.R.: One-equation turbulence model for aerodynamic flows. In: 30th Aerospace Sciences Meeting and Exhibit. pp. 5–21. Reno, NV (Jan 6 - 9, 1992). https://doi.org/10.2514/6.1992-439
27. Wang, J., He, C., Li, R., Chen, H., Zhai, C., Zhang, M.: Flow field prediction of supercritical airfoils via variational autoencoder based deep learning framework. Physics of Fluids **33** (8 2021). https://doi.org/10.1063/5.0053979
28. Xu, Z., Saleh, J.H., Yang, V.: Optimization of supercritical airfoil design with buffet effect. AIAA Journal **57**, 4343–4353 (2019). https://doi.org/10.2514/1.J057573
29. Yildirim, A., Kenway, G.K., Mader, C.A., Martins, J.R.: A jacobian-free approximate newton–krylov startup strategy for rans simulations. Journal of Computational Physics **397**, 108741 (11 2019). https://doi.org/10.1016/j.jcp.2019.06.018
30. Zhang, Y., Yang, Q.: A survey on multi-task learning. IEEE Transactions on Knowledge and Data Engineering **34**(12), 5586–5609 (2022). https://doi.org/10.1109/TKDE.2021.3070203

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97557-8_29