Non-Intrusive Reduced Order Model Formulation for Inverse Shape Design Including Deforming Meshes and Multiphysics Problems.

by

KAPIL ARYAL

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2023

To my father Sudarshan Aryal and mother Mitthu Aryal, who set the example and who made me who I am.

ACKNOWLEDGEMENTS

I would like to extend my heartfelt gratitude to my advisor, Dr. Brian Dennis, for his invaluable mentorship, unwavering support, and constant encouragement throughout my journey in the CFD Lab and during my Ph.D. program. Dr. Dennis, I am deeply thankful for allowing me to explore my area of interest and for the countless insightful discussions that enriched my thesis.

My sincere appreciation goes to my esteemed committee members: Dr. Zhen Xue Han, Dr. Bo Ping Wang, Dr. Kent L Lawrence, and Dr. Guojun G Liao. Your guidance, constructive feedback, and critical insights were instrumental in refining my work.

I extend my thanks to my fellow colleagues, both past and present, at the Computational Fluid Dynamics Lab (CFD Lab). Your support, camaraderie, and motivational conversations have been a tremendous source of inspiration during my time in the CFD Lab. I deeply value the meaningful discussions we've had on a wide range of captivating topics.

I am grateful to the MAE department at UTA for their financial support throughout my Ph.D. journey.

My heartfelt thanks go to my family for their unwavering support, boundless encouragement, and patience. Their unending love and blessings have made this remarkable journey possible, and I share this achievement with them.

To all my dear friends, your encouragement and shared belief in my aspirations have played a pivotal role in my pursuit of ambitious goals.

Lastly, I want to express my deepest gratitude to Apsara. She has been my pillar of strength, consistently stood by my side, and brought tranquility amidst life's challenges. Apsara, your unwavering presence has been my source of solace, and I am profoundly thankful for your constant support.

October 13, 2023

ABSTRACT

Non-Intrusive Reduced Order Model Formulation for Inverse Shape Design Including Deforming Meshes and Multiphysics Problems.

KAPIL ARYAL, Ph.D.

The University of Texas at Arlington, 2023

Supervising Professor: Brian H. Dennis

Despite significant advancements in computer capabilities for numerical simulations, engineers continue to face limitations when dealing with large-scale full-order model(FOM) simulations. These simulations often necessitate repeated solves, such as those encountered in inverse design, real-time solution prediction, error quantification, and solver convergence, among others. To address these challenges, reduced order modeling (ROM) has emerged as a valuable approach. This thesis focuses on the development of an ROM framework that combines Proper Orthogonal Decomposition (POD) with machine learning techniques. This integrated approach is applied to a diverse range of heat transfer and fluid flow inverse design problems.

POD constructs optimal sets of basis vectors from high fidelity numerical simulations which can be used in linear superposition to predict the FOM solution. The unknowns in the model are the coefficients of the basis vectors. To obtain these coefficients, various methods

can be employed, such as Galerkin projection or other optimization techniques, particularly when the solution field is known. Then such data can be utilized to train (ANNs), which will then be capable of making real time predictions of the coefficients for new sets of parameters. These coefficients when used with POD bases constructs the full solution field quickly. This versatile ROM framework is applied to address various heat transfer and fluid flow problems, enhancing computational efficiency in a range of scenarios.

Firstly, it is applied to 3D linear heat conduction within a deforming mesh configuration in a pipe, where the internal surface is governed by 20 parameters. The ROM accurately predicts the full temperature field with errors of less than 3.5% compared to the actual values. All 20 parameters are approximated within a remarkable 0.7% of the actual parameters. This framework establishes itself as a notably robust technique, surpassing other methods that rely directly on ANNs. The ROM demonstrates strong resilience to simulated errors in the target temperature.

Secondly, the ROM is applied to determine the detailed internal flow and temperature in a multiphysics, nonlinear conjugate heat and mass transfer problem within a hollow cylindrical channel with a spherical heat source at its center, for varying inlet flow rates. As flow rate information may not always be available, three nodal temperatures from the surface of the cylinder or sphere are used to re-parameterize the problem. The ROM exhibits satisfactory accuracy when temperature sensors are placed on the cylinder's surface and responds poorly to simulated errors. However, when the ROM is updated with sensors on the surface of the heated sphere, its performance significantly improves, providing highly accurate predictions even in the presence of substantial Gaussian noise in the sensor temperatures. Impressively, the ROM predicts temperature and absolute velocity within 0.5% and 2.5%, respectively, with errors not exceeding 5% for both temperature and velocity when subjected to Gaussian noise within the range of ±10°C.

In the final phase of the study, the ROM is applied to predict the pressure field for various 4-digit NACA airfoils at different angles of attack—a nonlinear 2D deforming mesh fluid flow problem. The predicted nodal pressure values are within 4% of the actual values for 25 test cases. In inverse approximation, the ROM demonstrates remarkable accuracy, recovering parameters and AOA within 2.5% of actual values, even when facing substantial Gaussian noise (equivalent to 5% of the highest pressure value). A groundbreaking discovery is made that this ROM is capable of eliminating multiple local minima (resembling noise) in the objective function.

The objective of this comprehensive investigation is to establish a generalized procedure applicable to a wide range of applications. By scrutinizing key factors such as modal trajectories, sample sizes for POD and ANN, the number of modes, and sampling techniques, the research aims to develop a versatile ROM framework capable of accommodating various scenarios.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Nomenclature

$\bar{U}$	Matrix of POD basis vectors
λ	Matrix of eigenvalues
$\mathbf{F}$	Body force
$\mathbf{K}$	Viscous stress tensor
μ	Dynamic viscosity
ρ	Density
$\underline{\mathrm{T}}$	FOM temperature field
$\vec{D}$	Vector of modal coefficients
$\vec{Q}$	Load vector in FEM
d	Number of POD modes after truncation
K	Global stiffness matrix in FEM formulation
k	Thermal conductivity
L^2	Squared Distance between two vectors
m	Maximum camber of a NACA airfoil
P	Pressure field
p	Location of maximum camber of a NACA airfoil
R_i	Radius of heated sphere
R_o	Radius of circular channel
Re	Reynolds number

S	Snapshot matrix
t	Maximum thickness of a NACA airfoil
U, V	Velocity vectors
v	Matrix of eigenvectors
x, y	Cartesian coordinates
Y	Covariance Matrix of S
Z	FOM solution
ANN	Artificial Neural Networks
BEM	Boundary Element Method
DMD	Dynamic Mode Decomposition
FDM	Finite Difference Method
FEM	Finite Element Method
FOM	Full Order Model
FVM	Finite Volume Method
GP	Galerkin Projection
LHS	Latin Hypercube Sampling
MSR	Molten Salt Reactors
NACA	National Advisory Committee for Aeronautics
PCA	Principal Value Decomposition
POD	Proper Orthogonal Decomposition
RBF	Radial Basis Function
RBM	Reduced Basis Method
RN	Residual Network
ROM	Reduced Order Model
SN	Sequential Network
SVD	Singular Value Decomposition

CHAPTER 1

Introduction

1.1 Background

Numerical solutions for scientific and engineering challenges have gained widespread popularity since the advent of computers. This popularity arises primarily because analytical solutions are often unavailable for most partial differential equations (PDEs). Despite substantial advances in computing power, engineers still encounter limitations, especially in the simulation of large-scale problems, particularly when frequent solving of high-fidelity models is necessary.

The primary constraint stems from the significant increase in dimensionality during the discretization process when employing high-fidelity models such as the Finite Difference Method (FDM), Finite Element Method (FEM), Finite Volume Method (FVM), Boundary Element Method (BEM), and similar techniques. Consequently, the computational expenses and time required for solving such models become prohibitively high. When these high-cost models are applied to address problems that require repeated solutions, such as optimization, inverse design, error quantification, or prediction of real-time solutions, the associated time and costs escalate exponentially.

To address these challenges, model order reduction techniques have emerged as highly effective tools. These techniques aim to reduce the dimensionality of the original

full-order model (FOM) while preserving the essential system behavior characteristics. As a result, these reduced models possess significantly fewer unknowns compared to the original FOM, resulting in a substantial reduction in computational demands while maintaining the accuracy of the solution.

Reduced order models (ROMs) are constructed through a model reduction process that involves several steps, including selecting an appropriate reduced basis, projecting the high-dimensional model onto this basis, and the subsequent solution of the reduced-order model. FOM solutions are utilized to construct the reduced basis that represent the behavior of the original system. These reduced basis are then used to construct a low-dimensional subspace that captures the essential dynamics of the system. The high-dimensional FOM is then projected onto this subspace, resulting in a lower-dimensional ROM that captures the critical behavior of the original model. ROMs have been essential in modeling, reducing computational costs in numerical simulations and making them valuable tools for both academic and industrial research, particularly in post-processing analysis, optimization, inverse problems, error quantification, and control-type problems where repeated solves of the model are required over a wide range of parameters [1].

The first step involved in model order reduction is the selection of the right model that can solve the system's behaviour to generate high fidelity solution. Then such data are used with appropriate model reduction techniques to come up with the right reduced model. Finally, such models are validated against the known solution to ensure the correct working of the models. The major shortcoming of using reduced model is that it comes with a compromise in the accuracy of the solution. The quality of FOM solution, model order reduction technique, and the complexity of the original model are the key factors that determine the accuracy and the efficiency of reduced-order models, and hence study of such models for their optimal applications in science and engineering is a major research topic.

Projection-based methods, such as Dynamic Mode Decomposition (DMD), Reduced Basis Method (RBM), and Proper Orthogonal Decomposition (POD) method, are commonly used for model reduction in which the original model is projected onto a reduced-dimensional approximation model [2]. DMD is a data-driven method for reduced-order modeling that can capture dominant modes of the system's evolution [3], [4]. RBM is a physics-based method that uses the Galerkin projection to reduce the dimensionality of the original system [5], [6]. POD is a projection-based method that uses statistical analysis to construct a reduced-order model that retains the dominant modes of the original system. While DMD and RBM are suitable for systems with non-linearities and time-dependent behavior, POD is more effective for capturing dominant modes that exhibit linear behavior. Similarly, POD modes represent spatial patterns in data variability, and thus give more physical interpretation than DMD or RBM methods. This is specially advantageous in areas like fluid dynamics or structural dynamics where understanding the underlying physics is a crucial part of the analysis. It captures dominant modes that exhibit linear behavior with higher fidelity [7]. Other alternative approaches like balanced truncation [8], Krylov Subspace Methods [9] and sparse grids [10] are frequently employed in various applications as model reduction techniques. Nevertheless, when considering various factors, including simplicity, superior physical interpretability, noise resilience, a well-established methodology, and seamless integration with other techniques, it becomes evident that POD stands out as a superior choice. POD, which was originally developed for data analysis in probability and statistics, is also known as Principal Component Analysis(PCA), Singular Value Decomposition (SVD), or Karhunen-Lo'eve expansion [11]. POD serves as a foundational technique for modal decomposition of a collection of functions, including data derived from experiments or numerical simulations. The characteristics of the POD imply that it is the favored foundation to employ in diverse applications. The result of its calculation often comes by the names of empirical eigenfunctions, empirical basis func-

tions, empirical orthogonal functions, proper orthogonal modes, or basis vectors [12]. The central concept of POD is to decrease a significant number of interconnected variables into a significantly smaller set of unrelated variables, while still maintaining the majority of the variance in the initial variables [13]. In POD, the high-fidelity solution is approximated as a linear superposition of orthogonal base functions which are capable of capturing most important structures in the data. POD technique constructs such basis vectors using a rotated co-ordinate system where the chosen rotation angles are such that the projections of the vectors on consecutive axes decay as quickly as possible [14]. In this theory, the decomposition extracts deterministic functions from second-order statistics of a random field and converges optimally fast in quadratic mean [15]. In other words, a random function can be expanded as a linear combination of a set of deterministic functions with random coefficients so that it is possible to separate the deterministic part from the random one. The POD procedure provides sets of empirical eigenfunctions which approximate an ensemble of a data set better than representations of the same dimension in terms of any other bases [16] [17]. The unknowns in the POD-based model are the coefficients of the base vectors. A comprehensive examination of these coefficients in the literature will be presented in the upcoming section, aligning with the motivations and objectives of this research, as it will be more contextually relevant to establish them there.

1.2 Literature Review of POD Applications in Diverse Fields.

POD has been applied to a wide variety of science and engineering disciplines very successfully. The following areas of strong applications of POD provide a basis for the thesis study: 1)Fluid mechanics: POD is used for flow analysis, turbulence modeling and control problems. 2)Structural mechanics: POD is used for modal analysis, damage detection and optimization. 3)Image processing: POD is used for feature extraction, com-

pression and denoising. 4)Medical study: POD is used in analyzing brain activity and abnormality detection, flow analysis inside the blood vessels and so on. 5)Computational dynamics: POD is used for real time solution prediction, optimization, inverse design, error quantification and so on.

1.2.1 POD in Fluid Mechanics

Rowley et al. [18] used POD for constructing reduced order models of turbulent flow and shows that POD is capable of capturing the dominant dynamics of the flow field and facilitates computationally efficient simulations. This paper focuses on spectral analysis of non linear flows since the traditional linear analysis methods fall short in capturing the intricate behaviour of non linear flows, such as turbulent flows. It emphasizes that such linear analysis of POD is very suitable even for nonlinear flow analysis.

Bewley et al. [19] applied POD in conjuction with Direct Numerical Simulation (DNS) for turbulence control strategies to determine the controls that minimizes the turbulent kinetic energy and drag for turbulent flows in plane channels at different Reynolds number.

Ly et al. [20] used POD for steady-state Rayleigh-Bénard convection by assuming that its mathematical model is unknown. Basically, the paper used ensemble of data to show that it can be used efficiently to accurately model the natural convection even in the absence of mathematical model.

Fogleman et al. [21] applied POD to datasets of internal combustion engine flows obtained through CFD and particle imaging velocimetry for different fixed crank-angle positions over a number of cycles. The method was used to analyze the instability mechanism involved in tumble breakdown and to analyze the flow in time-varying domain.

Zhang et al. applied POD to study the aerodynamics of a membrane wing under the micro air vehicle flight condition. The paper concluded that even though a substan-

tial number of eigenmodes are required to adequately reproduce the aerodynamic data, the framework is capable of offering a reduced ordered representation for the membrane aerodynamics.

1.2.2 POD in Structural Mechanics

Lu et al. [22] studied the continuous modeling of rotor systems using POD and also combined with a model based on network dynamics. It allowed an effective means of studying the dynamic characteristics of rotor system from the network dynamic perspective.

Han et al. [23] applied POD in the modal analysis of beams with different boundary conditions. It was shown that only one POD mode converges to the true normal modes of the structure. Experimental data from both a homogeneous free–free beam and a non-homogeneous free–free beam were utilized to perform a comparative analysis between proper orthogonal modes and structural normal modes. The outcomes strongly endorse the suitability of employing proper orthogonal decomposition for structural vibration analysis.

Bamer et al. [24] applied POD to structures subjected to transient earthquake loading to both linear and nonlinear structures. The POD approximations were very close to the actual solution for both cases even when small number of POD modes were utilized. It was shown that the advantage of Proper POD over Modal Truncation extends beyond the optimality of POD modes in terms of their energy representation; it also encompasses its suitability for nonlinear systems.

Thiene et al. [25] studied POD for its application to damage detection in homogeneous plates and composite beams. The study was conducted to detect damage and effects of several parameters such as damage severity, type of excitation, noise level, the model's sensitivity to sensor locations and modifications of the boundary conditions. It was shown that POD successfully detected defects in cantilever plates as well as damage along the length and thickness of the beam.

1.2.3 POD in Image Processing

Kumar et al. [26] applied PCA for face images for compression or dimensionality reduction and face recognition. The paper reviewed and compared the performance of three such methods, Eigen Decomposition, SVD, and Hebbian Neural Networks.

Ting et al. [27] introduced an innovative method that deviates from the conventional practice of applying PCA exclusively to rectangular or square Regions of Interest (ROIs). Instead, it employs PCA to effectively compress ROIs of arbitrary shapes in MRI brain images. Through factorization, this approach achieves successful compression of such ROIs at varying compression ratios. The simulation outcomes demonstrate that even at a total compression ratio as high as 80%, the test image exhibits no discernible distortion.

Gewers et al. [28] provided simple applications of PCA to real world data such as astronomy, biology, chemistry, computer, engineering, geography, weather, medicine and so on to outlier detection, variance explanation, pattern recognition and also to dimensionality recognition to such data.

1.2.4 POD in Medical Study

Bakhshinejad et al. [29] presented a novel method for merging 4D flow magnetic resonance imaging(MRI) with CFD to non-invasively measure blood velocities without being affected by acquisition noise, flow artifacts, and resolution limits. POD solution was used to compare against other denoising methods, and it was shown that POD method preserved the small flow structures than other methods, while also eliminating noise.

Janiga et al. [30] visualized features of unsteady blood flow in intracranial aneurysms with the help of POD. The 4D flow was decomposed with the help of POD and the POD features were extracted with the help of POD velocity modes. Finally, the novel POD-based visualizations were illustrated for 4D PC-MRI and CFD.

Balasubramanian et al. [31] studied the efficacy of POD framework, designed for identifying structural glaucomatous alterations in the optic nerve head, in individuals participating in the UCSD Diagnostic Innovations in Glaucoma Study (DIGS). The POD framework exhibits potential in offering a high level of specificity for detecting glaucomatous changes, even with just a single follow-up examination.

1.2.5 POD in CFD and Heat Transfer

Samadiani et al. [32] reviewed reduced order modeling approaches used in predicting flow and temperature fields within data centers, focusing on design parameters. The motivation for such modeling in designing energy-efficient thermal management systems was discussed. Special attention was given to POD as a means of modeling turbulent flows. It was concluded that low-dimensional models, especially POD-based techniques, are valuable for accurately and rapidly predicting the thermal behavior of data centers, aiding in design and control.

Rulli et al. [33] introduced and applied phase-invariant POD analysis, conditional averaging, and triple and quadruple POD methods to an extensive database of particle image velocimetry data from a well-known research engine. The results highlight the ability of these methods to provide both quantitative and qualitative insights into cycle-to-cycle variations. Additionally, the authors proposed a new quadruple proper orthogonal decomposition methodology and compared it to existing techniques in the literature. Overall, these methods proved useful in identifying turbulent structures responsible for cycle-to-cycle variability, and they can be applied to both experimental and numerical datasets for detailed analysis and comparisons of turbulent fields.

Manthey et al. [34] applied POD to reduce the model for in-depth analysis of natural two-phase flow stability in a high pressure natural gas circulation system with great success

and laid a basis to develop a reduced model for low pressure thermal hydraulics. The validation of the ROM solutions showed very good agreements with the system code.

Vergari et al. [35] formulated a POD-ROM for multi-physics application to a homogeneous reactor model based on a Lid-Driven-Cavity configuration. The reduced-order simulations demonstrated a high degree of accuracy in reproducing the complete velocity, temperature, neutron flux, and precursor concentration fields that are coupled through temperature feedback mechanisms. This accuracy was reflected in the relative L^2 norms of the discrepancies between the full-order and reduced-order simulations, all of which remain below 1% for all the fields under consideration. The methodology introduced in this study presented an efficient and precise approach for addressing multi-physics challenges within the field of nuclear engineering while managing computational resources effectively.

German et al. [36] used POD aided Reduced-Basis technique to develop a multi-physics reduced model for the analysis of nuclear system and applied it to parametric simulations of Molten Salt Reactors(MSR). This model analyzed fluid dynamics, heat exchange and neutronics phenomena and proved to have a capability of speeding the simulations on the order of 1-10^5.

Lucia et al. [37] reviewed the ROM techniques and discussed their applicability to various problems in computational physics. Large increases in computational efficiency were obtained through ROM use thus justifying the initial computational investment in constructing these model. The emphasis was given to methods based on Volterra series representations, harmonic balance and POD.

Zimmermann and colleagues [38] employed Computational Fluid Dynamics (CFD) on an airfoil to generate Proper Orthogonal Decomposition (POD) basis functions. These basis functions were then combined with experimental lift and drag measurements to reconstruct the flow under various conditions.

Ruscher et al. [39] harnessed the symmetries present in the velocity field of an axisymmetric jet to address gaps caused by camera obstructions. They successfully utilized the integrated POD approach to restore missing data in the instantaneous flow field.

1.3 Motivations and Objectives

Accurate approximation of the POD coefficients is the fundamental for the effectiveness of POD-based applications across various domains, ranging from fluid dynamics and structural mechanics to image processing and machine learning. A lot of ideas on the determination of such coefficients can be found in scholarly works such as Galerkin projection [40], polynomial approximation [41], [42], [43], radial basis approximation [44], krigging [45],least squares method [46], [47], missing point estimation [48]collection methods, interpolation method [49], reduced basis method [50] and recently machine learning [51], [52], [53], [54], [2]. Therefore more exploration of the literature concerning the selection of such concepts is required. As the research focuses on POD-ANN ROM, this concise survey will primarily emphasize ANN-based approaches, their outcomes across diverse applications, and their comparative analysis with other established techniques. Ultimately, the findings and the objectives of this research will be summarized.

Swischuk et al. [2] applied POD with four different machine learning techniques—neural networks, multivariate polynomial regression, k-nearest-neighbors and decision trees—for two engineering examples. The first example considered prediction of the pressure field around an airfoil, while the second considered prediction of the strain field over a damaged composite panel. Among four methods, the K-nearest-neighbors outperformed other 3 methods. The author pointed out that the poor performance of the neural network approach is due to insufficient data being used in neural network. The case studies also demonstrate the importance of embedding physical constraints within learned models and highlight the

important point that the amount of model training data available in an engineering setting is often much less than it is in other machine learning applications, making it essential to incorporate knowledge from physical models.

Hesthaven et al. [55] proposed a non-intrusive reduced basis (RB) method for parameterized steady state partial differential equations based on POD and Artificial Neural Network (ANN). This approach was successfully applied to nonlinear Poisson equation and steady, viscous, incompressible Navier–Stokes equations. This study relied on Latin Hypercube Sampling(LHS) and Levenberg-Marquardt algorithm to search for the optimal number of neurons and the minimum number of training samples. This strategy provided the same predictive accuracy provided by POD-Galerkin method.

San et al. [1] compared POD-ANN approach with POD-GP (Galerkin Projection) in which it was shown that POD-ANN approach is superior to the POD-GP approach for the Burgers equation. Two different neural network architectures, namely sequential network (SN) and residual network(RN), were successfully used to train the trajectory of modal coefficients evolving through time. In the SN approach, the network was trained with modal coefficients at the current time step, current time and Reynold's number as input and the modal coefficients at the next step as output. In RN approach, the network was trained with the same inputs as in SN, this time the outputs are the difference between the modal coefficient at next time step and the modal coefficient at current time, also known as residual. It was found that POD-ANN-RN gave more accurate and stable results both inside and outside the database range than the standard intrusive Galerkin Projection model. An importation result found was that the PO ANN is a viable tool for extrapolation and interpolation beyond the data sets used to train its learning

Falkiewicz et al [56] studied the application of POD on thermal solution in hypersonic aeroelastic simulations. It was found that the error incurred in the reduced-order solution does not always decrease with the number of snapshots. Therefore, it is desired to

study the effect of number of snapshots used in the POD model reduction. Also, the error was found to increase outside of the time range considered by the snapshots.

Rogers et al. [57] applied POD integrated with a trained Radial Basis Function (RBF) network in inverse estimation of the parameters in heat transfer, elasticity and fracture mechanics. This method was successfully applied to problems with relatively large measurement errors or noise.

Everson et al [47] applied POD to restore the full image when partial data of the image is given. It was shown that the recovery is possible within reasonable accuracy by following least squares procedure with POD.

Huayamave et al. [58] tested and validated trained POD–RBF interpolation network by performing the fast-algebraic interpolation to obtain the pressure distribution on the photovoltaic system surface and they were compared to actual grid-converged fully turbulent 3D CFD solutions at the specified values of the design variables. The POD-RBF was validated and proved that large-scale CFD problems can be parametrized and simplified by using this framework. They were able to obtain pressure distribution within one percentage of the FOM solution for a case of five parameters.

Lee et al. [59] constructed ROM by POD-Kriging and POD-Radial Basis Function Network(RBFN) for temperature, O_2 and CO mass fraction fields in a 500 MWe tangentially fired pulverized coal boiler. They showed overall good agreement within an acceptable error range, while POD-Kriging showed slightly better accuracy than POD-RBFN.

Fic et al. [14] combined POD with FEM to solve transient nonlinear heat conduction problem. In this approach, the matrices in FEM solution were transformed by the matrix of POD basis vectors thus reducing the size of the systems of equations to be solved. This method was applied to both linear and non-linear transient heat conduction problem in a gas cooled turbine blade. The temperature field obtained was within a percentage in linear case and within 7% in non-linear case.

Ostrowski et al. [60] solved inverse steady state heat conduction problems by least square fit of the trained POD-RBF network to estimate heat conductivity, and film coefficient distribution for a given temperature field in an internally cooled 2D gas turbine blade. Some noise in the temperature field were introduced to illustrate the excellent filtration property of the POD basis which suppresses the high frequency noise. Even in the presence of errors much greater than the physically realistic values, the method was shown to be capable of producing stable results.

Here are the insights and main takeaways from this brief review. Galerkin Projection is a fundamental method for approximating POD coefficients where the governing equations are projected onto the POD modes to derive the ordinary differential equations for the coefficients. However, such approach requires prior knowledge of the underlying physics and also it could be challenging to use it for highly non-linear problems. Polynomial approximation, radial basis approximation, least squares method, missing point estimation, reduced basis method, krigging and interpolation methods, even though easier in mathematics, come with their own challenges. Such approximation techniques are inherently limited in their ability to approximate complex and nonlinear functions. The choice of right basis functions, loss of accuracy as the dimensions increase, difficulty with non-linear systems and generalization issues are some of the major limitations to choosing these approaches. On the other hand, the black box approach of machine learning, such as Artificial Neural Network(ANN), can easily capture non-linear and complex systems. Likewise, machine learning gives the flexibility of being completely data driven technique and generalizes well with unseen data or parameter sets [61]. It can also handle high dimensional and complex systems more effectively. Because of such valuable properties of machine learning, this research will use ANNs to construct POD-ANN ROMs. Additionally, based on the literature review and to the best of the author's knowledge, the utilization of such ROM has not been identified in the context of inverse parameter approximation involving domain de-

formation. Furthermore, no instances of applying this ROM to non-invasive determination of intricate flow and temperature characteristics were found. No examples of applying this ROM in the context of inverse multiphysics applications were identified in the literature. No research publications related to the quantification of potential reductions in the total number of FOM calls achievable through the implementation of this approach has been identified. Hence, the objective of this research is to expand the field of POD-ANN ROM by addressing the aforementioned gaps in the literature, using various practical engineering examples. This research attempts to study the following aspects of this ROM:

- What is the behavior of the coefficients of the basis vectors as they vary with parameters?
- How can we determine the optimal number of POD basis vectors required for this model?
- What sample size is optimal for POD and training the Artificial Neural Network?
- What impact does the sampling technique have, and does it warrant further investigation?
- How does the accuracy of this model compare to the FOM solution when applied to the forward problem?
- To what extent can this model accurately predict parameters in the context of inverse parameter estimation with a large number of parameters?
- How does this model's performance compare to other models that exclusively utilize ANNs?
- What is the computational time saved when applying this model to inverse problems in comparison to the Full-Order Model formulation?

1.4 Thesis Overview

The current chapter investigates the background, literature review, motivations and objectives, mathematical formulations and operational principles of POD-ANN ROM. Chapter 2 applies the ROM from chapter 1 to inversely determine the internal geometry of a hollow pipe from temperature measurements from its outer wall for 3-D linear heat conduction. In Chapter 2, the ROM's utilization for the non-invasive determination of intricate internal flow and temperature characteristics in a conjugate heat and mass transfer multiphysics application within a nuclear power plant is discussed. Chapter 3 is dedicated to recover the shape of a 2D airfoil for a given target pressure on its surface. In these chapters, different aspects of this ROM, such as choice of number of modes, number of POD samples, sampling techniques, sensitivity to simulated noise, comparison of this technique to other model reduction techniques, and its computational gain will be presented. Discussions on future work is presented in chapter 5 followed by conclding remarks in chapter 6.

1.5 Methodology/ Reduced Order model Formulation

POD starts with the collection of FOM solution field with m nodes and stored in a column vector, known as snapshot. Let, matrix S is the collection of n snapshots where each column is a snapshot, then the covariance of S of dimension m by n is,

$$[Y] = [S]^T [S] \tag{1.1}$$

The covariance matrix $[Y]$ can be decomposed as an eigenvalue problem as follows,

$$[Y] = [v][\lambda][v]^{-1} \tag{1.2}$$

Where (λ) is eigenvector and v is the eigenvalue matrix of the covariance matrix $[Y]$. If the eigenvectors are sorted in a descending order of their corresponding eigenvalues in (λ), then the original snapshot matrix when multiplied by the eigenvector matrix gives the non-normalized POD basis vectors, i.e.

$$[\bar{u}] = [S][\lambda] \tag{1.3}$$

Finally, all the column vectors of $[\bar{u}]$ are normalized by dividing by their corresponding $L2$ norm and updated as $[\bar{U}]$, which are referred to as POD modes (or basis vectors), i.e.

$$[\bar{U}]_{*i} = \frac{[\bar{u}]_{*i}}{||[\bar{u}]_{*i}||} \tag{1.4}$$

Now, any FOM solution, Z(x,p), a function of both space and parameters, can be projected onto these basis vectors as given by equation 1.5 [62].

$$Z(x,p) = \sum_{k=1}^{m} D_k(P)\bar{U}_k(x) \tag{1.5}$$

Where $\vec{D}$ is the column vector that stores the coefficients of the base vectors. $\bar{U}$ is also known as modal matrix and $\vec{D}$ is its modal coefficient vector. Note that $\vec{D}$ is exclusively a function of parameters only, including time, whereas $\bar{u}$ is solely a function of space. In other words, $\vec{D}$ encapsulates information related to parameters, whereas the spatial characteristics of the solution are encoded within $\bar{U}$. This separation enables the independent examination of the solution concerning parameters and spatial aspects.

The contribution of all the modes are not equally significant, and it diminishes as the eigenvalue of the corresponding eigenvector diminishes. Therefore, it is possible to truncate the number of modes used in POD without losing any significant accuracy. Such truncation is dependent upon the eigenvalue of the corresponding base vector and user specified

criteria for the accuracy, as will be obvious in the later section. The physical interpretation of eigenvalue is the amount of energy(or information) of the system contained by the corresponding mode. Therefore, in most literature, the truncation criteria is given by equation 1.6 where the percentage of energy captured is dictated by the portion of eigenvalues used.

$$d = argmin \left\{ I(d) = \frac{\sum_{k=1}^{d} \lambda_i}{\sum_{k=1}^{n} \lambda_i} : I(d) \geq \frac{p}{100} \right\} \tag{1.6}$$

Where d is the number of required modes with highest eigenvalues for a desired percentage accuracy of p and n is the total number of modes. As noted earlier, Galerkin Projection has been one of the popular techniques used in determining $\vec{D}$ where the original Partial Differential Equation (PDE) is projected onto the base vectors and the residual is minimized [40]. However, the technique requires prior knowledge of the governing PDE and if the problem is non-linear or involves parameters that affect shape or material properties, then such method is not computationally efficient. The challenge can be circumvented by realizing this unique characteristics, namely, the orthogonality of $\bar{U}$. Therefore, for any snapshot included in the snapshot matrix, the modal coefficient can determined easily as given by equation 1.7.

$$\vec{D} = \bar{U}^T Z \tag{1.7}$$

Equation 1.7 is applicable only for parameters whose FOM solution(Z) is known. Oftentimes, it is desirable to find the solution of the parameters that are not utilized in generating snapshots for POD. Then the challenge is to determine $\vec{D}$ corresponding to parameters whose solution field is unknown. Therefore, a new technique is utilized where the modal coefficients of the snapshots included in the snapshot matrix are first calculated and then they are used to train an Artificial Neural Network (ANN) with parameters as input and $\vec{D}$ as output. This ANN, when successfully trained, is able to predict the modal coefficients

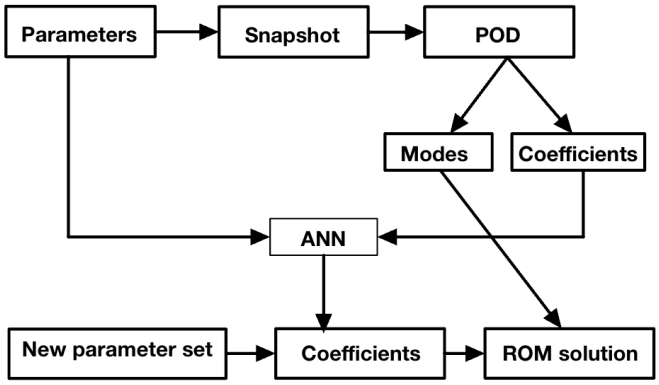


Figure 1.1. Step chart of the proposed POD-ANN framework.

for any other snapshot (or parameter set) not included in the snapshot matrix. Essentially, a relationship between the parameters of the PDE is established with the modal coefficients of the corresponding solution field. Such model is non-intrusive, unlike the Galerkin approach, since no prior information about the governing PDE is required once the snapshots are extracted. Access to the solver source code is not required and implementation simply requires access to the FOM mesh and field data. The overall flowchart of the proposed framework is shown in figure 1.1.

CHAPTER 2

Inverse Unknown Boundary Shapes Identification to a Deforming Mesh 3D-Heat Conduction in a Pipe: A Linear Problem.

The POD-ANN reduced model is applied to a 3-D steady state heat conduction problem whose geometry and mesh deform as the parameters change. Specifically, the ROM is applied to a forward problem where we want to approximate the temperature field when the parameters governing the internal surface of a pipe are known and to an inverse approximation where we want to recover the parameters governing the internal surface of the pipe when its temperature on the outer surface is known. In the next section, we will discuss domain parameterization and the forward problem of 3D heat transfer. We'll also cover a solver mesh convergence study and analyze modal trajectories. The selection of modes, sample size, and sampling techniques for the forward problem will be addressed. Additionally, we'll apply the methodology to inverse parameter estimation, aiming to reconstruct the internal pipe surface. Towards the end of this section, we'll conduct a comparative analysis against other ROMs and discuss the computational efficiency of our approach.

2.1 Forward Problem

In our study of POD-ANN, we picked a forward problem of solving a heat transfer equation in a hollow sphere to determine its temperature field for a given set of parameters that govern its internal surface. The outer surface of the pipe is circular with radius (R) 1, which is maintained at a constant temperature of $1000K$. The pipe has length (L) 10 times the outer radius. The inner surface, which is maintained at a constant temperature of $200K$, has geometry that is a function of parameters. The front and back of the pipe are insulated. The pipe has isotropic thermal property with a unity coefficient of thermal conduction(K). The schematic of the problem is shown in Figure 2.1a. The steady state temperature field$(\underline{\mathrm{T}})$ on this pipe can be obtained by solving equation 2.1. Such choice of heat transfer problem is made with the objective of addressing engineering challenges where it is not always feasible to take a direct snapshot of the internal geometry and have to rely on remote heat transfer data.

$$\nabla^2 \underline{\mathrm{T}} = 0 \tag{2.1}$$

2.2 Internal Geometry Creation

To get the internal surface of the pipe, we divided it into 5 planes along Z axis, with each plane having 4 vertices that are fixed in θ and Z direction but can be varied between 0.4 and 0.6 units from the center of the pipe in radial direction. This gives us a total of 20 parameters that control the internal surface of the pipe (in the sections to follow, when we say parameters, we refer to this set of radii). The schematic of a cross section of the pipe is shown in Figure 2.1b. Even though the parameters are varied radially, such variations are converted to cartesian co-ordinate system and thus rest of the calculations are done in Cartesian co-ordinate system. Then, closed cubic spline curves are formed that connect the

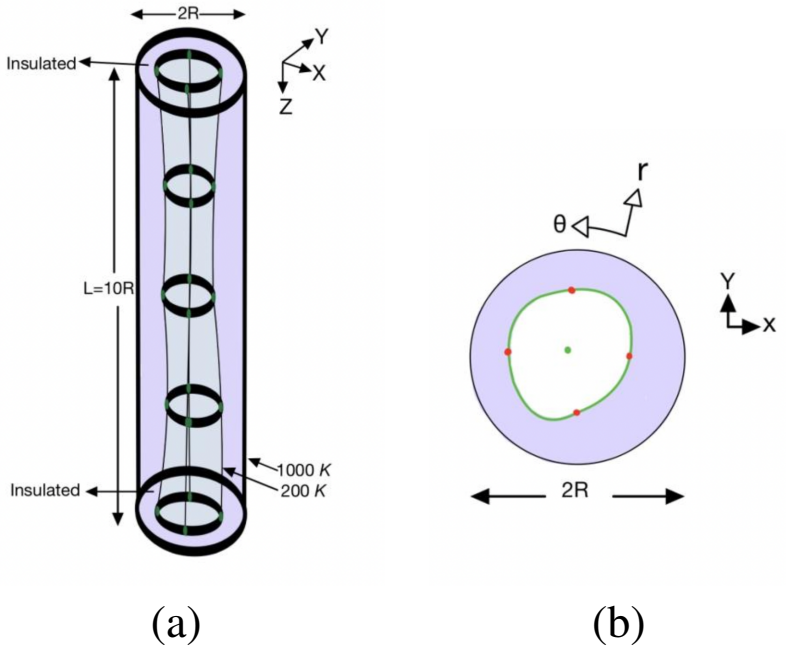


Figure 2.1. (a) Schematic of the problem, (b) Cross section of the pipe used in the study

4 vertices in each plane. Also, open cubic spline curves are formed along the axial direction to connect all 5 planes. The geometry is initialized by picking a random set of parameters- An example is shown in Table 2.1. Each column in Table 2.1 represents the radii in each plane. The internal surface of the pipe corresponding to this set of parameters is shown in Figure 2.2. With these parameters, we built a reference solution for a forward problem using full order finite element model (FOM) using hexahedral elements with linear basis functions.

Table 2.1. A random set of parameters that govern the internal surface of the pipe.

	Plane Number				
Radius	1	2	3	4	5
1	0.52948	0.47852	0.55670	0.55237	0.50327
2	0.52828	0.47314	0.47973	0.49369	0.44941
3	0.51404	0.48399	0.42630	0.56448	0.47028
4	0.49101	0.55936	0.53230	0.52660	0.50214

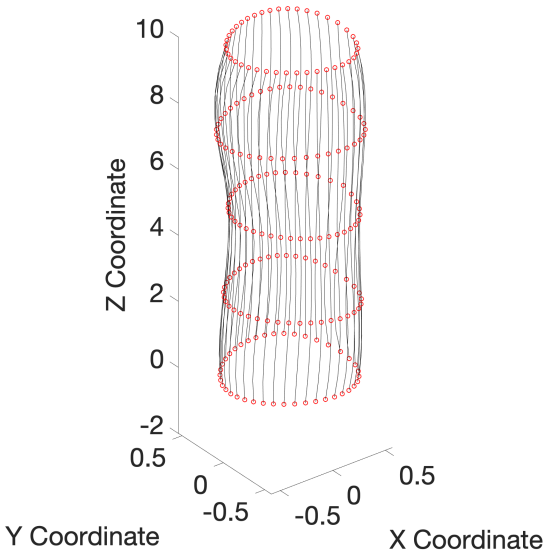


Figure 2.2. Sample internal surface of the pipe for the parameters shown in Table 2.1.

2.3 Mesh Convergence Study

The forward problem is solved for a given set of parameters using inhouse FEM code (written in MATLAB). When creating the mesh it is crucial to keep the same number of nodes in every snapshot. Given that the geometry undergoes transformations with varying parameters, it becomes evident that the mesh must adjust to accommodate this need. To achieve this, as the geometry evolves, the spacing between the nodes are stretched or contracted in radial direction all while preserving both the total number of nodes and their sequential ordering. To put simply, the number of nodes between the given radius and the outer surface remains constant, but the distance between the nodes can change according to the changing nature of the radius. Consequently, this issue pertains to a deforming mesh problem. We started with mesh convergence study for the forward solution of the problem with the parameters from Table 2.1. The results are shown in Table 2.2. In Table 2.2, T_{mean} is the average of entire temperature field and T_{mid} is the temperature at plane 1 $(Z = 0)$ of the pipe halfway between the inner and outer radius and at an axis that is zero degrees to X-axis for different number of nodes. Mesh 4 (4480 nodes) was chosen for the problem since the error values are within a quarter of a percentage and allows for faster computations while maintaining accuracy.

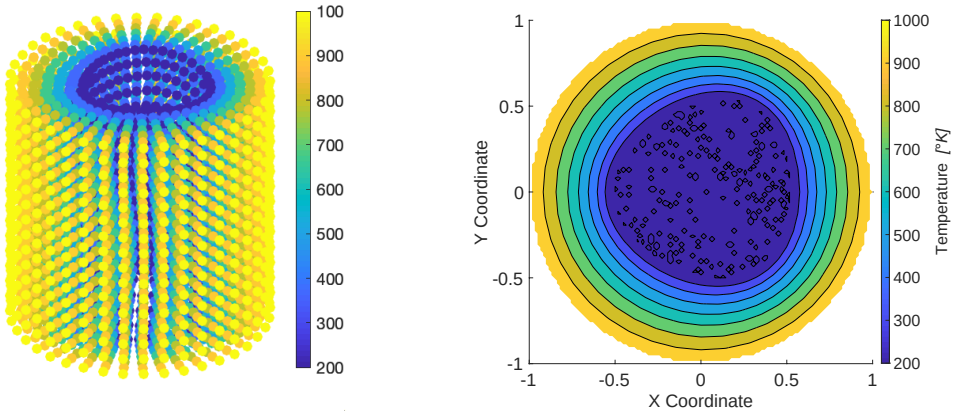
The temperature field at each node in the entire geometry using mesh number 4 is shown in figure 2.3a. The temperature contour corresponding to the temperature in the first plane $(Z = 0)$ is shown in Figure 2.3b.

In the study of inverse problem in the following sections, we will use the temperature field on the outer surface of the pipe in the optimizer's objective function to recover the parameters. Thus, it is important to have a unique temperature field on the outer surface of the pipe for a given set of parameters when creating snapshots for POD. Therefore, we

Table 2.2. Results for FOM mesh convergence study with the parameters from Table 2.1.

Mesh	# of Nodes	Entire Geometry		At a node	
		T_{mean}	% Error	T_{mid}	% Error
1	1120	634.47	0.49	659.71	0.32
2	2240	636.89	0.11	660.04	0.37
3	3360	637.39	0.03	659.13	0.23
4	4480	637.39	0.03	659.13	0.23
5	5600	637.46	0.02	658.77	0.18
6	6720	637.50	0.01	658.50	0.16
7	7840	637.53	0.01	658.29	0.14
8	20160	637.58	0	657.64	0.00
9	21288	637.58	0	657.62	0.00
10	22400	637.58	0	657.61	0.00

will give heat flux boundary condition ,as opposed to maintaining a constant temperature, on the outer surface of the pipe when creating POD snapshots. This heat flux vector will come from the nodal heat fluxes of the forward problem used in mesh convergence study in the preceding section.



(a) (b)

Figure 2.3. Mesh study results for mesh # 4 (from Table 2.2) for parameters in Table 2.1.(a)Temperature distribution in the pipe at each node in the entire domain. (b)Temperature contour at $Z = 0$ plane of the pipe.

2.4 Trajectories of Modal Coefficients

We created POD basis vectors using 80 snapshots whose parameter sets are randomly chosen between 0.4 and 0.6. Then, we allowed one of the 20 parameters to vary uniformly between 0.4 and 0.6 while keeping the remaining 19 parameters the same as above and calculated their FOM solution as well. Since the objective is to study how the trajectories of modal coefficient change with respect to a single parameter variation, we have few options to calculate the modal coefficient for this new set of FOM solution. First, we considered finding the coefficients by minimizing the L^2 norm(distance) between the FOM solution and ROM solution. Second, we included all additional FOM solution as extra snapshots and used equation 1.7. The other technique utilized Galerkin method where the residual of

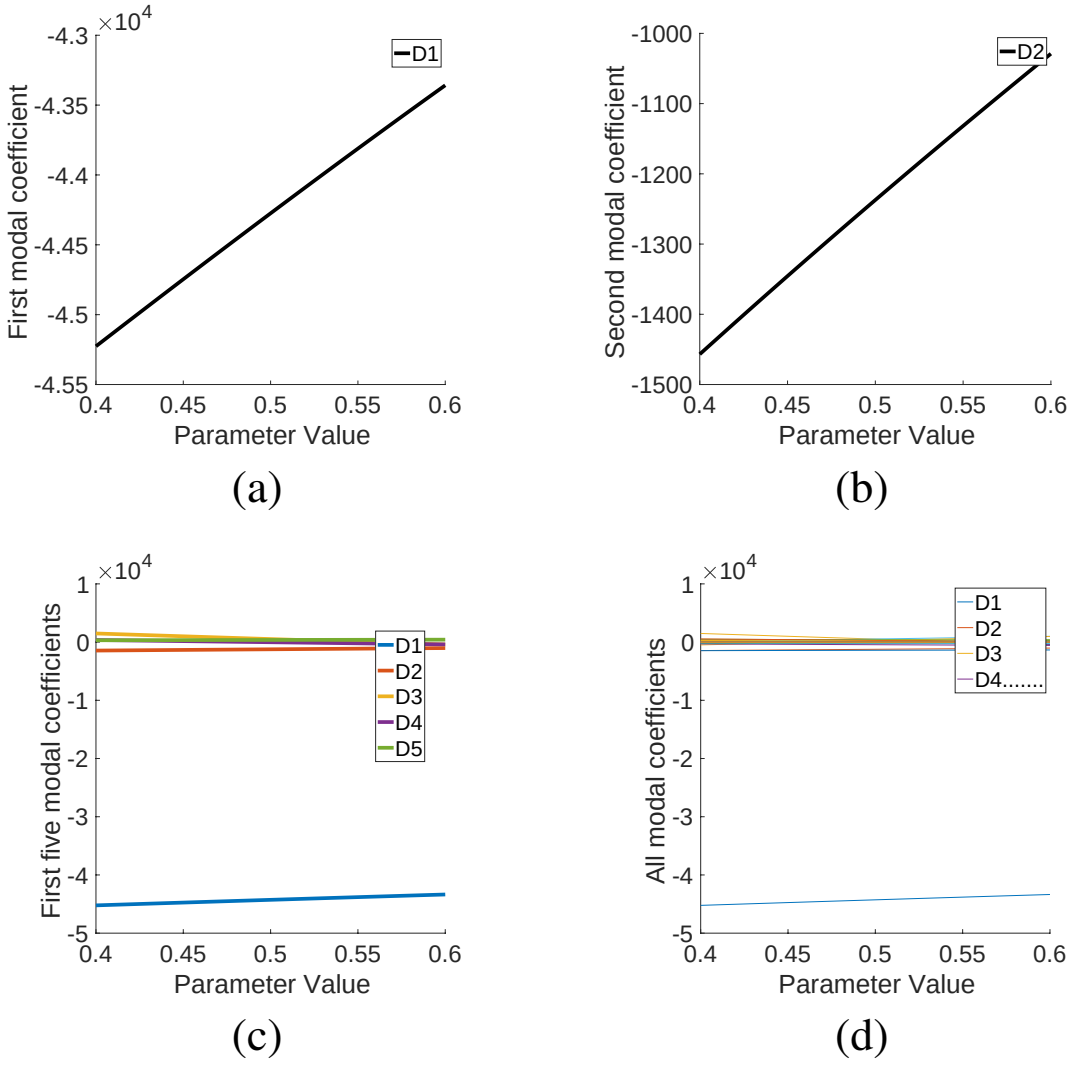


Figure 2.4. Trajectories of the modal coefficients when the third radius in second plane of the pipe (from parameter Table 2.1.) is varied while keeping all other radii the same for 80 snapshots POD. (a) First (b) Second (c) First five and (d) all modal coefficients.

the PDE is minimized. Dennis at al. [40] combined POD, FEM and Galerkin projection for heat transfer analysis, whose final equation is given by equation 2.2.

$$[\bar{U}]^T[K][\bar{U}]\vec{D} = [\bar{U}]\vec{Q} \tag{2.2a}$$

$$\text{i.e., } K_{ROM}\vec{D} = \vec{Q}_{ROM} \tag{2.2b}$$

$$\text{where, } K_{ROM} = [\bar{U}]^T [K][\bar{U}] \tag{2.2c}$$

$$\vec{Q}_{ROM} = [\bar{U}]\vec{Q} \tag{2.2d}$$

where, $[K]$ is the global stiffness matrix, and $\vec{Q}$ is the load vector in FEM equation 2.3,

$$[K]\underline{T} = \vec{Q} \tag{2.3}$$

Remarkably, regardless of the methods, the obtained modal coefficients were found to be exactly the same. Such observation confirms an existence of a unique combination of modal coefficients for a given set of POD basis vectors. Figures 2.4 show the variation of elements of $\vec{D}$ as the third parameter from second plane in Table 2.1 changes. As observed, there is a smooth transition in the coefficients as this parameter varies. Because we observe a smooth variation in $\vec{D}$ as one of the parameters changes, it is reasonable to expect that ANNs can effectively capture the combined effects resulting from alterations in every parameter due to its interpolation capability, as also suggested by Ferrari et al. [63]. Therefore, we trained feed-forward ANNs with parameters as input and $\vec{D}$ as output using Levenberg–Marquardt algorithm [64], a combination of Gradient Descent and the Gauss-Newton method, particularly designed for solving nonlinear least squares problems, and mainly useful when there are many parameters, as is the case here. Separate networks are trained for each element of $\vec{D}$. In training the networks, the number of hidden layers and units are automated in MATLAB code such that the root mean squared error(rmse) is minimum for 25% data selected for the validation.

2.5 Number of Modes

In POD method, it is possible to truncate the number of POD basis without losing the accuracy of approximation. This is because some of the POD modes might be linearly dependent upon others and truncation of such modes is desirable for ROM computational gain. Therefore, we studied the effect of number of POD vectors in POD-ANN accuracy with 60 snapshots and 200 snapshots. The temperature field for a random set of parameters (not included in the snapshots) was approximated using ROM. The maximum percentage

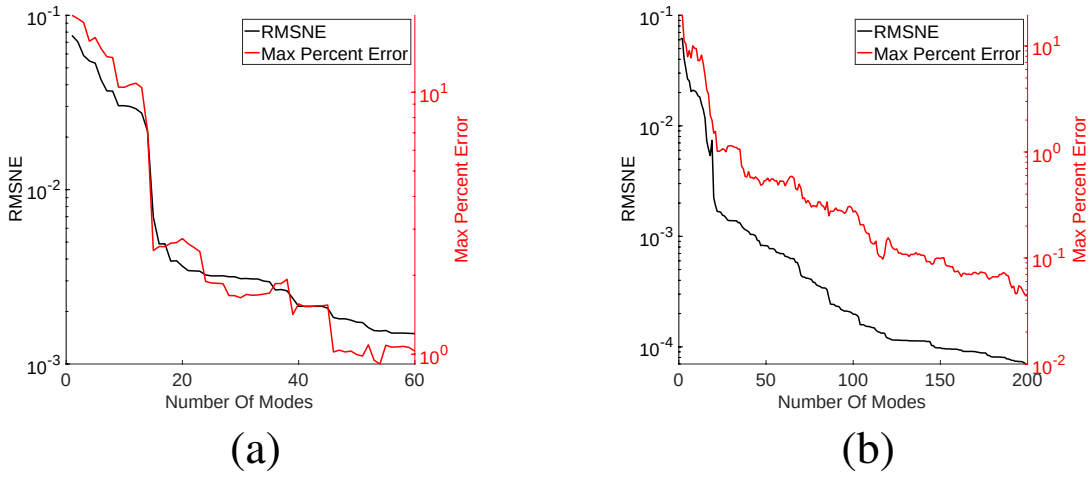


Figure 2.5. RMSNE and maximum Percentage error in temperature recovered by POD-ANN ROM as a function of number of POD modes for a parameter set not included in the snapshots. ROM (a) has 60 snapshots and ROM (b) has 200 snapshots.

error and root mean squared normalized error (RMSNE, equation 2.4) between the ROM approximated temperature and FOM temperature field is calculated as we varied the number of modes used in ROM. We made two such calculations, once using 60 snapshots ROM and the other using 200 snapshots ROM. These errors as a function of number of modes are shown in Figures 2.5 (in log scale). Clearly, the errors do not settle to a steady value as the

number of modes increases for both 60 snapshots case and 200 snapshots case. Therefore, no POD modes are truncated in the further development and study of POD-ANN ROM whenever the number of snapshots used in ROM is less than 200 throughout this chapter unless otherwise specified.

$$RMSNE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{T_{FOM,i} - T_{ROM,i}}{T_{FOM,i}}\right)^2} \tag{2.4}$$

2.6 Number of Samples

Another important consideration to be made in this approach is the choice of number of POD snapshots and ANN training sample size. We tested POD-ANN ROM built with different sample sizes to approximate the temperature field for 20 randomly selected sets of parameters that are not included in the original snapshots. Consequently, we calculated the average of the RMSNE and the average of maximum percentage error (averaged across 20 solution sets) in ROM-recovered temperature field as a function of number of samples used in ROM and plotted them in Figure 2.6. These metrics change only very marginally after about 60 samples. Also, after about 85 samples, the errors stabilize at cetrain values. This is because only about 85 samples are required to effectively train the networks, as such, adding more to the training would not significantly enhance their performances. Therefore, in the further study of POD-ANN, we will employ 85 samples, unless specific conditions dictate otherwise.

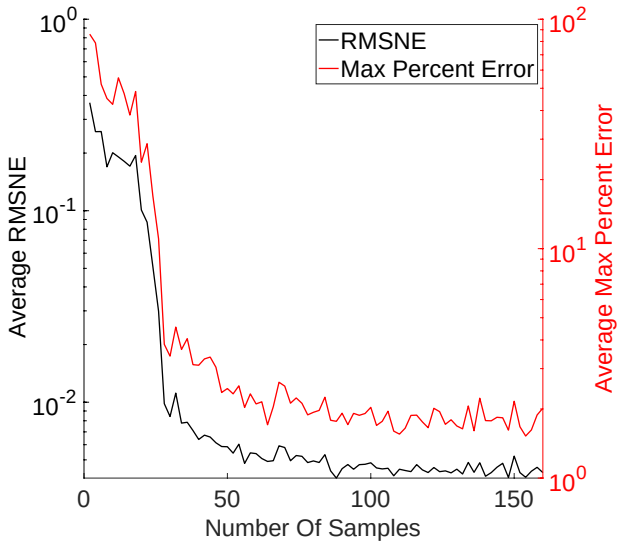


Figure 2.6. maximum percent error and RMSNE (Averaged over 20 sets of error calculations) between FOM solution and POD-ANN ROM solution Vs. number of snapshots used in the ROM.

2.7 Sampling Technique

Performance of POD approach and neural networks are highly dependent upon sampling of their inputs and outputs. To explore this aspect, we created ROM with 85 snapshots whose parameters are obtained through three distinct sampling techniques: Random sampling, Latin Hypercube Sampling(LHS), and a custom sampling method we devised based on experimental observations, denoted as 'Method 3'. In Method 3, we divided each parameter space into a set of N uniformly spaced parameters, where N is the number of snapshots desired in ROM. Then, these sets of evenly spaced parameters were shuffled to give N sample parameters. Finally, this process is repeated for every parameter individually.

ROM created with each sampling technique is used to approximate the temperature field for the above specified 20 sets of randomly generated parameters. Figure 2.7 through Figure 2.9 show the maximum percentage error and RMSNE between the actual tem-

perature distribution and ROM recovered temperature for different sampling techniques. Clearly, Method 3 sampling outperformed LHS and random sampling in terms of error results. While the accuracy of Method 3 sampling over random sampling may not be significantly higher, the real advantage is its low likelihood of encountering an exceptionally poor set of parameters because each parameter in Method 3 sampling has uniform data from every parametric space that are later randomized. However, Method 3 sampling is necessarily not the best sampling technique, and therefore further detailed studies are warranted. In the sections to follow, we will continue to use Method 3 sampling technique in our analysis.

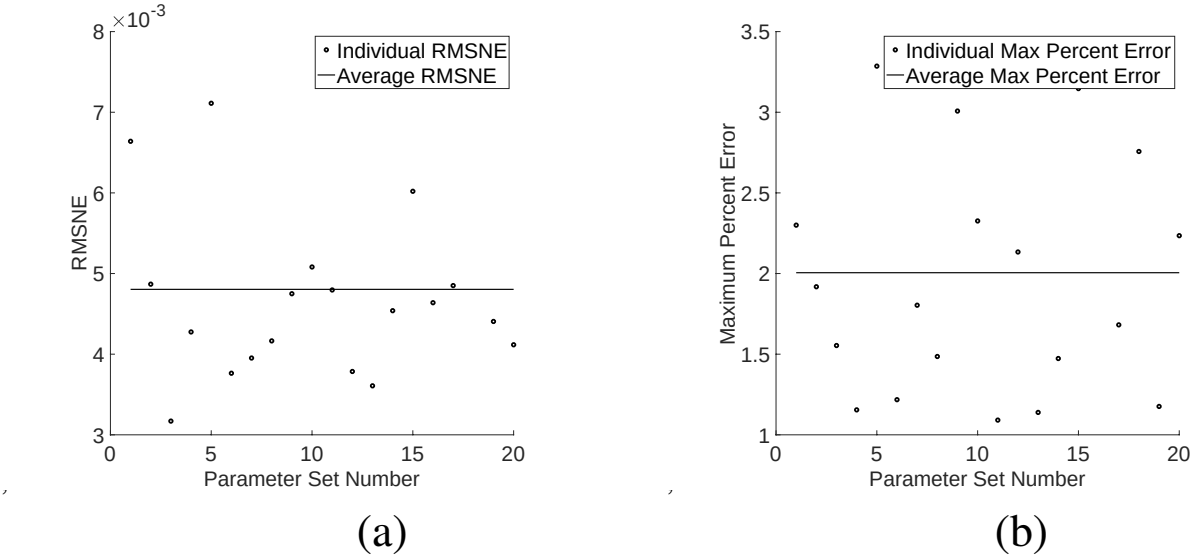
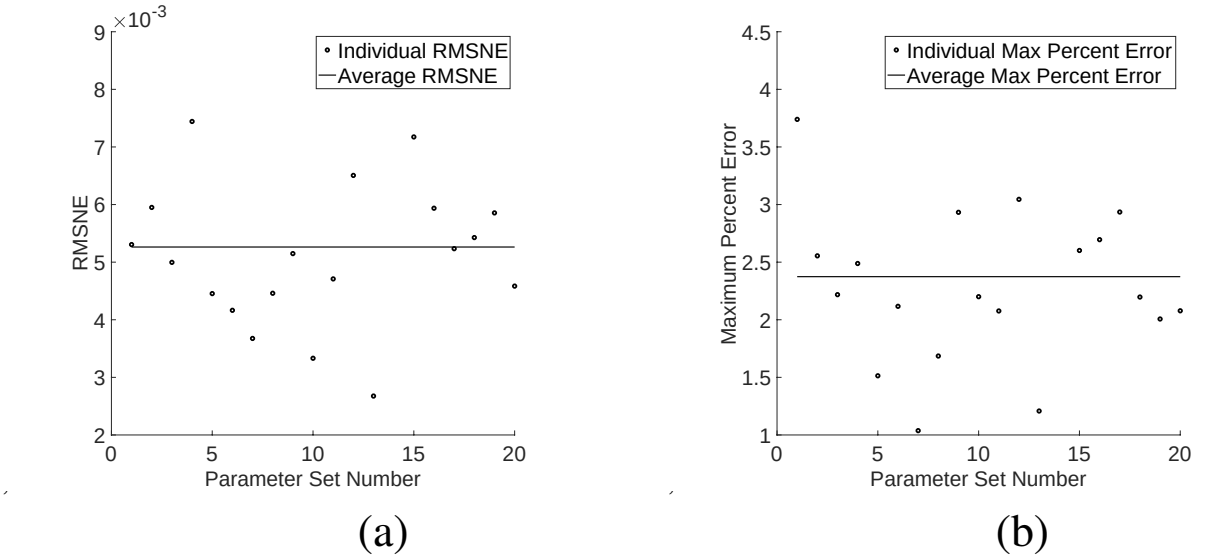
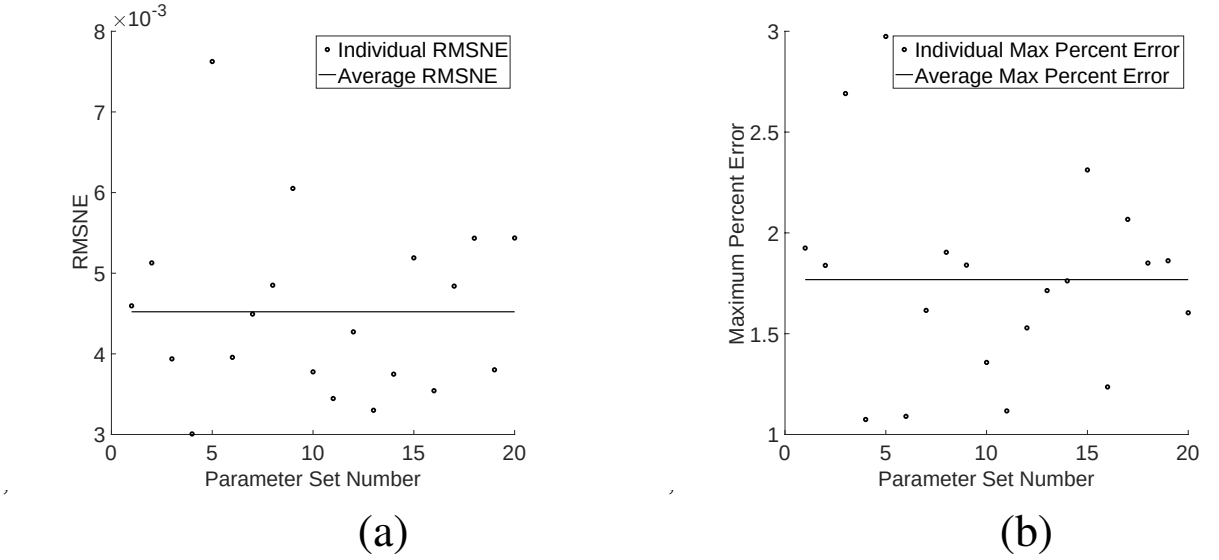


Figure 2.7. (a) RMSNE and (b) Maximum percentage error in temperature recovered using POD-ANN with random sampling for 20 random sets of parameters.



(a) (b)

Figure 2.8. (a) RMSNE and (b) Maximum percentage error in temperature recovered using POD-ANN with Latin Hypercube sampling for 20 random sets of parameters.



(a) (b)

Figure 2.9. (a) RMSNE and (b) Maximum percentage error in temperature recovered using POD-ANN with 'Method 3' sampling for 20 random sets of parameters.

2.8 Fully Formulated ROM Applied to Forward Problem

Combining all results from previous sections, the ROM formulation utilizes the Method 3 sampling technique, with 85 samples, and no truncation in POD modes (uses all 85 modes). With this information, we built POD-ANN ROM (which will also be used in inverse approximation of parameters in later sections) and approximated the temperature field. Figures 2.10 show the maximum percentage error and RMSNE in temperature field approximated by this ROM for 20 random sets of parameters. The maximum percent error in the approximated temperature is about 3.25%, with the average being around 1.7%. The actual temperature field and ROM recovered temperature field corresponding to parameters from table 2.1. are shown in Figure 2.11.

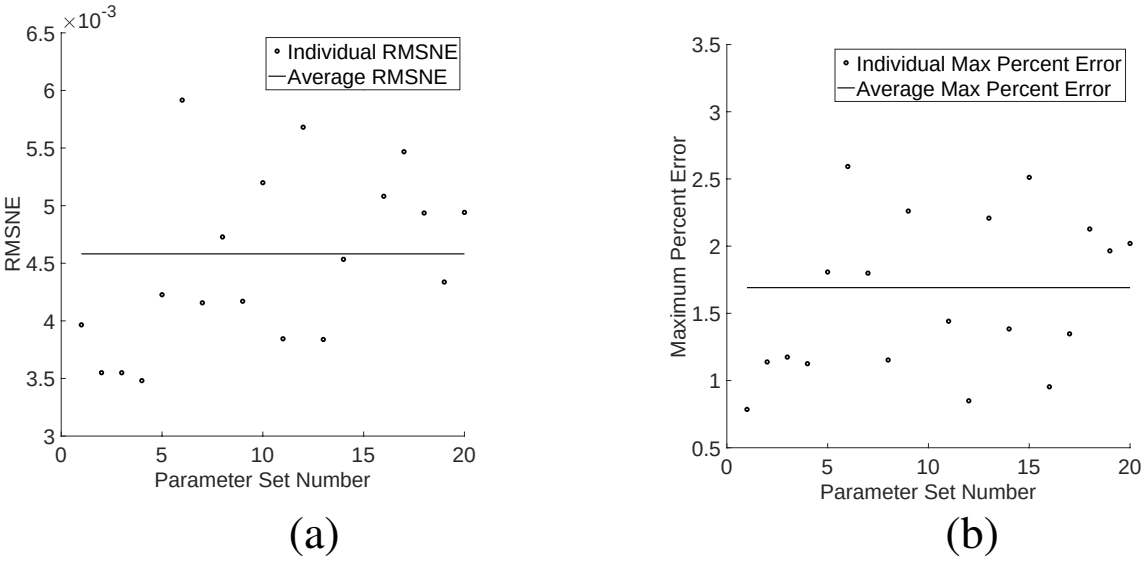


Figure 2.10. Errors in temperature recovered by fully formulated ROM for 20 random sets of parameters.

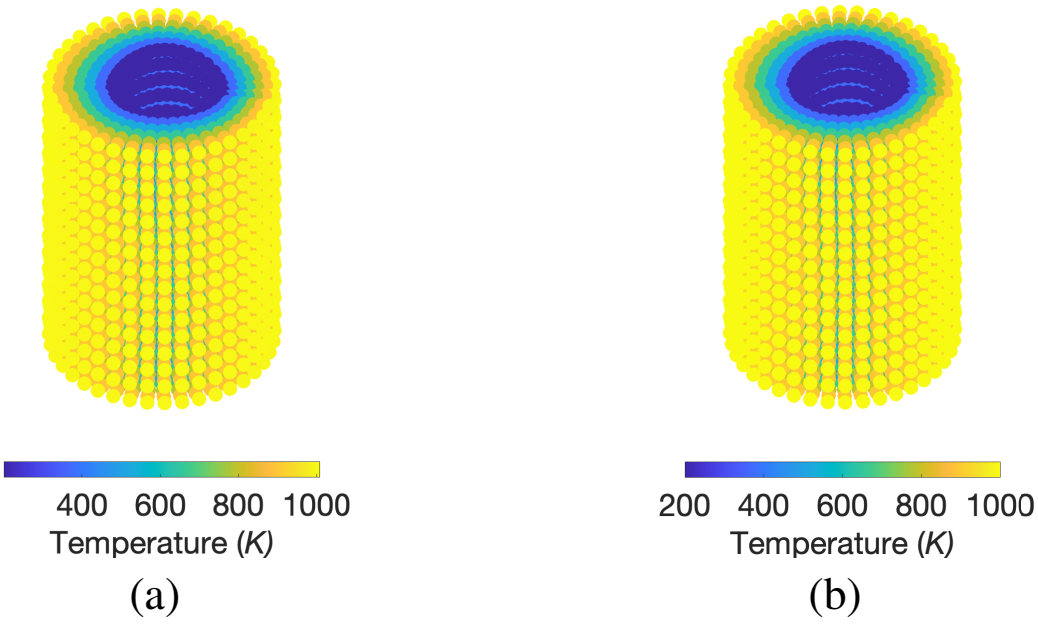


Figure 2.11. Comparison of temperature distribution in the pipe obtained using (a)full order FEM and (b) ROM.

2.9 Inverse Approximation

To further test the feasibility of the POD-ANN-ROM formulation, it is applied to an inverse problem aiming to approximate the parameters in Table 2.1 when the temperature field at each node on the outer surface of the cylinder is known.

For optimization, We used 'fmincon' with 'interior-point' algorithm option [65] in MATLAB. 'Interior-point' algorithm works best for problems with many design variables, which aligns with the characteristics of our study. This method reaches the best solution by traversing the interior of the feasible region. The error objective function for the optimizer is the RMSND between the target temperature distribution and inverse solution for a given set of parameters, as given by equation 2.5. We solely consider temperature field from the

outer surface of the pipe in the error objective function and so N in equation 2.5 represents the nodes on the outer surface of the pipe rather than encompassing the entire geometry. This mirrors the real case scenario where it is desired to know the internal surface of the pipe when only the temperature of its outer surface ($1000K$) is known. Additionally, this demonstrates the effectiveness of the POD-ANN formulation to solve inverse problems even when only partial data of the field variable is known.

For optimization, the algorithm was set to stop when it attempted to take a step size smaller than 5×10^{-4}. We started the optimization with initial guess shown in Table 2.3 (its geometry is shown in Figure 2.12), while the parameters to be recovered are from Table 2.1. Table 2.4 and Figure 2.13 show the percentage errors in ROM recovered parameters of Table 2.1, the maximum error being 0.69% with the mean error of about 0.2%. The internal surface of the pipe corresponding to the actual parameters and POD-ANN approximated parameters are shown in Figure 2.14.

$$Error_{obj} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{T_{actual,i} - T_{ROM,i}}{T_{actual,i}}\right)^2} \tag{2.5}$$

Table 2.3. Initially guessed set of parameters.

	Plane Number				
R	1	2	3	4	5
1	0.404	0.404	0.404	0.404	0.404
2	0.404	0.404	0.404	0.404	0.404
3	0.404	0.404	0.404	0.404	0.404
4	0.404	0.404	0.404	0.404	0.404

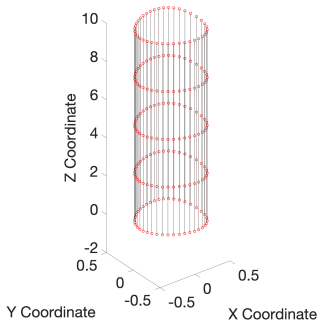


Figure 2.12. Internal surface of the pipe corresponding to the parameters in Table 2.3.

Table 2.4. Percentage errors in POD-ANN recovered parameters of Table 2.1 .

	Plane Number				
Radius	1	2	3	4	5
1	0.46	0.09	0.69	0.08	0.09
2	0.04	0.03	0.17	0.26	0.20
3	0.33	0.28	0.60	0.33	0.66
4	0.17	0.12	0.10	0.23	0.06

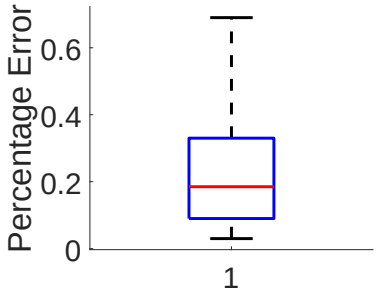


Figure 2.13. Box and whisker plot showing the errors in POD-ANN recovered parameters of Table 2.1.

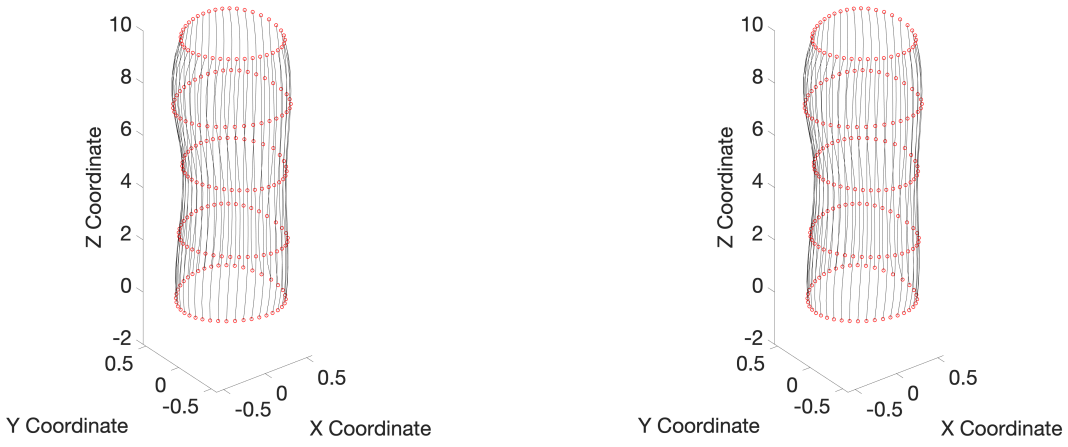


Figure 2.14. Comparison of POD-ANN recovered geometry with the actual geometry.

2.10 Comparison of POD-ANN with Other ANN Based Approaches

There are many other approaches that leverage ANN in solving similar problems. One such approach is to train an ANN with parameters as input and the field variable of interest as output or vice-versa and then subsequently using the trained network to approximate the solution field or parameters. Another approach in inverse problem is to train an ANN with parameters as input and the distance between the actual field solution and the field solution at different parameter set as output. Then, it is desired to find the minimum distance in order to recover the actual parameters. In the subsequent sub-sections, we will compare the accuracy and efficiency of POD-ANN with such approaches.

2.10.1 POD-ANN Vs ANN Trained with Temperature Field and Parameters.

In this direct approach, we trained an ANN with the temperature on the outer surface of the pipe as input and the parameters that govern its internal geometry as output. We trained an ANN for each parameter separately using Levenberg Marquardt algorithm whose number of hidden layers and units were dictated by the same rmse criteria explained in earlier section. To maintain consistency, we selected the same sample of 85 snapshots used in POD-ANN. Then, the parameters in Table 2.1 were recovered by supplying its corresponding nodal temperature data on the outer surface of the pipe to the network. The maximum percentage error of the recovered parameters is 4.75% and mean error is about 2% with direct ANN training approach (shown in Table 2.5 and Figure 2.15). In comparison, POD-ANN method yielded lower errors (0.66% and 0.2% respectively). The inputs for ANN for this direct approach come from the temperature field, which could be a very large number of inputs for the neural networks to handle and thus it makes this approach impractical to use sometimes.

Table 2.5. Recovered Parameters of Table 2.1 using ANN trained with temperature and parameters.

	Plane Number				
R	1	2	3	4	5
1	1.07	0.26	0.15	4.75	2.38
2	3.01	0.84	0.22	4.43	0.01
3	0.59	1.91	3.52	3.24	0.93
4	2.33	4.26	2.41	1.49	0.01

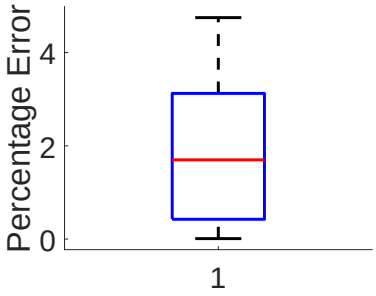


Figure 2.15. Box and whisker plot showing the errors in recovered parameters of Table 2.1 using ANN trained with temperature and parameters.

2.10.2 POD-ANN ROM Vs ANN Trained with RMSNE and Paramrters

Another more commonly used approach in approximating the parameters is to train a neural network with parameters as input and root mean squared normalized distance (RMSND) between the actual temperature and the temperature corresponding to those parameters as output. Then, in inverse parameter estimation, we seek to find a set of parameters such that the network would give a minimum RMSND value. When experimented with various training methods, we found that the approach used in POD-ANN worked best for this particular case as well. Then, using 'fmincon' and interior point algorithm in MATLAB, we tried to recover the parameters in Table 2.1. To make sure it is not stuck at some local minimum, we started the optimizer using 40 different initial guesses. The percentage errors in the recovered parameters are shown in Table 2.6 and Figure 2.16. Evidently, the recovered parameters significantly vary from the actual parameters. The errors are confined to about 25% due the constraint established in the optimizer that each parameter should fall between 0.4 and 0.6, otherwise these errors would be significantly higher.

Table 2.6. Recovered Parameters of Table 2.1 using ANN trained with parameters as input and RMSNE as output.

	Plane Number				
R	1	2	3	4	5
1	10.22	10.67	23.78	24.40	16.60
2	10.15	10.65	19.65	9.60	6.76
3	9.73	14.78	0.80	13.89	11.14
4	19.05	1.35	7.67	11.34	7.49

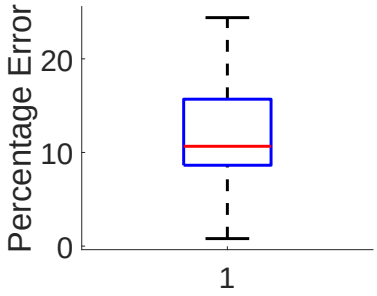


Figure 2.16. Box and whisker plot showing the errors in recovered parameters of Table 2.1 using ANN trained with parameters and their RMSNE.

The suboptimal results in this approach are not due to issues with ANN training or optimization algorithm, but due to an insufficient number of samples used in training the network. To demonstrate this, we could increase the number of POD snapshots and size of ANN training sample, however, given that there are 20 parameters, all the possible combinations among them would be exceedingly large and we would need an unrealistic number of samples in the study. Alternatively, we tried this approach with fewer parameters. We only considered 3 parameters each from first two planes in Table 2.1. This means that the internal surface of the pipe is constructed using 3 vertices at each end of the pipe. There are 720 possible combinations among the 6 parameters, and therefore we attempted to estimate these 6 parameters again with 85 samples and 720 samples by minimizing RMSND. The percentage errors are shown in Table 2.7, Table 2.8, Figure 2.17, and Figure 2.18 respectively. The maximum percentage error in the parameter is 5.58% with 85 samples and 3.49% with 720 samples.

Table 2.7. Percentage errors in recovered parameters of Table 2.1 using ANN trained with parameters and their RMSNE:6 parameters, 85 samples

	Plane Number	
Radius	1	2
1	2.91	0.40
2	3.00	1.17
3	2.91	5.58

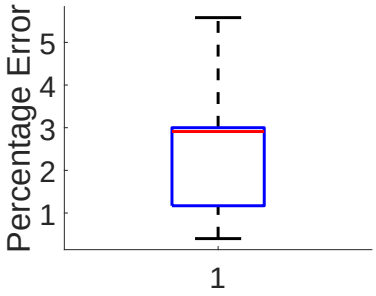


Figure 2.17. Box and whisker plot showing percentage error in recovered parameters using ANN trained with temperature and RMSNE with 6 parameters and 85 samples.

Table 2.8. Percentage errors in recovered Parameters using ANN trained with temperature and RMSNE: 6 parameters, 720 samples.

	Plane Number	
Radius	1	2
1	0.97	3.49
2	1.63	0.38
3	3.18	0.03

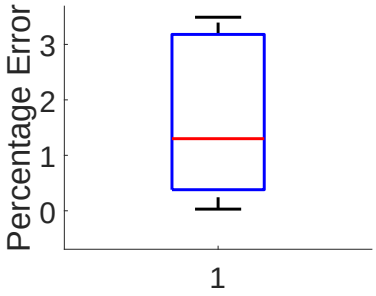


Figure 2.18. Box and whisker plot showing percentage error in recovered parameters using ANN trained with temperature and RMSNE with 6 parameters and 720 samples.

Finally, to make comparisons, we repeated the parameter estimations (for 6 parameters case) using POD-ANN with 85 samples and 720 samples, the percentage errors in the parameters are shown in table 2.9 and table 2.10 respectively. The POD-ANN recov-

Table 2.9. Percentage errors in recovered Parameters using POD-ANN ROM with 6 parameters and 85 samples.

	Plane Number	
Radius	1	2
1	0.20	0.21
2	0.18	0.23
3	0.29	0.24

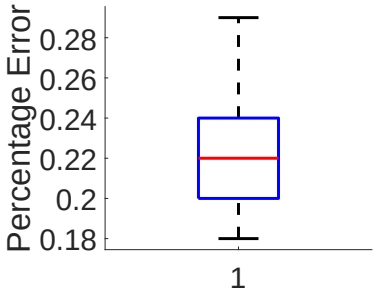


Figure 2.19. Box and whisker plot showing percentage errors in recovered Parameters using POD-ANN ROM with 6 parameters and 85 samples.

ered parameters are within a quarter of a percentage of the actual parameters for both 85 samples and 720 samples as opposed to 5.58% and 3.49% with RMSND minimization approach. These results undeniably highlight a superiority of POD-ANN approach over the commonly used approach that directly trains parameters vs errors for inverse approximation, especially when dealing with high dimensional space. It is worth noting that the POD-ANN approach did not gain any performances by increasing the number of samples from 85 to 720. As stated in the forward solution in earlier section, this is because most of the significant characteristics of the system are already captured by 85 modes. Additionally, this quantity of sample is also sufficient for ANN training, yielding no benefits in the context.

Table 2.10. Percentage errors in recovered Parameters using POD-ANN approach:6 parameters, 720 samples.

	Plane Number	
Radius	1	2
1	0.21	0.24
2	0.19	0.22
3	0.20	0.24

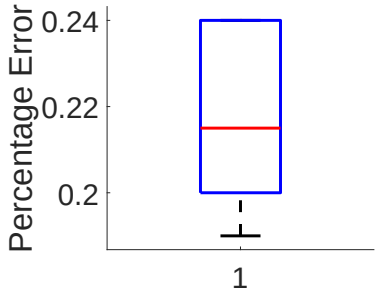


Figure 2.20. Box and whisker plot showing percentage errors in recovered parameters using POD-ANN approach with 6 parameters and 720 samples.

2.11 Inverse Approximation in the Presence of Noise

In order to test the feasibility of this ROM formulation in the presence of noise, we added two different levels of white Gaussian noise into the target solution. The Gaussian noises had maximum values of $\pm 2\%$ and $\pm 10\%$ of the maximum temperature temperature, corresponding to $\pm 20^o K$ and $\pm 100^o K$, respectively. The recovered parameters with such noisy data are shown in Tables 2.11 and 2.12. The percentage errors in the calculation visualized through box and whisker plots in Figures 2.21 and 2.22. The results clearly show the robustness of this framework as an excellent filtration tool even with large levels of noises.

Table 2.11. Recovered Parameters of Table 2.1 using ROM and ±2% Gaussian noise.

	Plane Number				
R	1	2	3	4	5
1	0.527	0.478	0.553	0.555	0.498
2	0.527	0.475	0.479	0.492	0.449
3	0.517	0.483	0.427	0.565	0.476
4	0.491	0.561	0.531	0.528	0.491

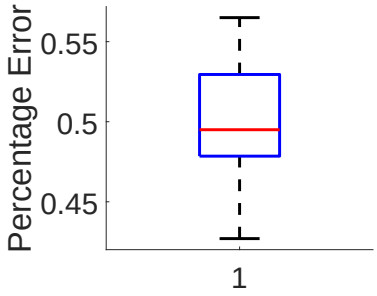


Figure 2.21. Box and whisker plot showing the errors in ROM recovered parameters of Table 2.1 with ±2% Gaussian noise noise.

Table 2.12. Recovered Parameters of Table 2.1 using ROM and ±10% Gaussian noise.

	Plane Number				
R	1	2	3	4	5
1	0.530	0.495	0.568	0.548	0.546
2	0.560	0.467	0.484	0.401	0.442
3	0.478	0.489	0.438	0.582	0.439
4	0.495	0.562	0.521	0.519	0.499

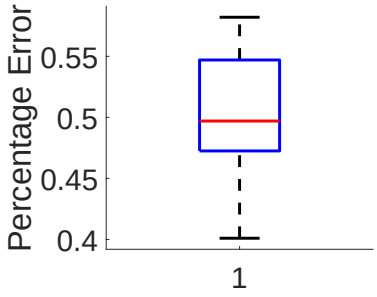


Figure 2.22. Box and whisker plot showing the errors in ROM recovered parameters of Table 2.1 with ±10% Gaussian noise noise.

2.12 Trade-off between Number of Samples in ROM and FOM Calls in Inverse Parameter Estimation.

Through Figure 2.6, we established that the change in error values using POD-ANN were not appreciable after 60 samples and plateaued out after about 85 samples when ap-

plied to the forward problem. In inverse estimation for 6 parameters case earlier, it was shown that there is no added benefit to the framework's performance by increasing the sample size from 85 to 720. Therefore, it is justified to seek an optimal sample size for inverse problem as well. On the other hand, after selecting the right sample size, the ROM recovered parameters are precise enough for most of the applications, however, if higher precision in the parameters in inverse solution is desired, then further use of optimizer with FOM is needed. In such case, we want to start optimizer with the parameters recovered by ROM as the initial guess since our ROM recovered parameters are very close to the actual parameters already. The sample size and the efficiency of increasing POD snapshots in such case is not obvious. Therefore, it is desired to seek an optimal trade-off between the sample size used in ROM and the number of FOM runs required to recover the parameters whenever precision better than ROM recovered parameters is desired. We created POD-ANN ROM with sample size of 40 using Method 3 sampling technique and applied it to approximate the parameters in Table 2.1. With ROM recovered parameters as initial guess, we noted the number of FOM runs required by the optimizer to satisfy a specified constraint of step size tolerance of 5×10^{-4}.

Since different sets of snapshots yield different results even with the same sample size, we repeated these calculations for 12 times for sample size of 40, each time with different snapshots used in ROM obtained using Method 3 sampling technique. Then, we kept on increasing the sample size by 20 at a time until the size of 120 to create ROMs and noted the total FOM calls required for each sample size in inverse approximation whose results are shown in figure 2.23a. In Figure 2.23a, each circle represents the number of FOM runs required by the optimizer plus the corresponding sample size used in ROM when the optimizer was initialized with ROM recovered parameter set. The maximum, average and minimum number of FOM runs are represented by the red, blue, and black

solid lines respectively. Each circle at zero sample location shows the number of FOM runs required by the optimizer at various random initial guesses without ROM.

Notably, around 60 to 80 samples, we reach a minimum, which is also the location of an optimal trade-off between the number of samples used in ROM and total FOM calls required. This is because the accuracy of ROM plateaus after this sample size and increasing the size would only add to the number of total FOM runs (as demonstrated in figure 2.23b). The optimal sample size shown here is in direct agreement to what was observed in figure 2.6 where about 85 samples were identified as optimal in the forward problem. The slight decrease in the optimal sample size here is due to improvements in ROM as sampling technique used here is changed from random sampling to Method 3 sampling. Therefore, it will be safe to use the optimal sample size in forward problem as the optimal number of samples for inverse problem too. In figure 2.23, the number of FOM runs at a given sample size vary every time the snapshots used in the sample is changed, even when Method 3 sampling technique is used. This result also underscores the need for a more stable sampling technique to get better consistency in the results out of this model.

The difference between FOM runs required with ROM and without ROM is very significant. As seen in figure 2.23a (note the small circles at 0 samples), the maximum number of FOM runs required without ROM is 700 (which could go even higher depending on the initial guess) while the minimum number of total FOM runs required with ROM is 197 (note the small circle at sample size of 80), which is 28% of the maximum FOM runs. This means, if we had the most optimal sampling technique, we could have saved at least 72% or even higher number of FOM runs (depending on the target and the initial guess used in the optimizer). It is noted that the percentage of FOM runs represented here depends on the desired step size tolerance of the optimizer, which in our case was 5×10^{-4}.

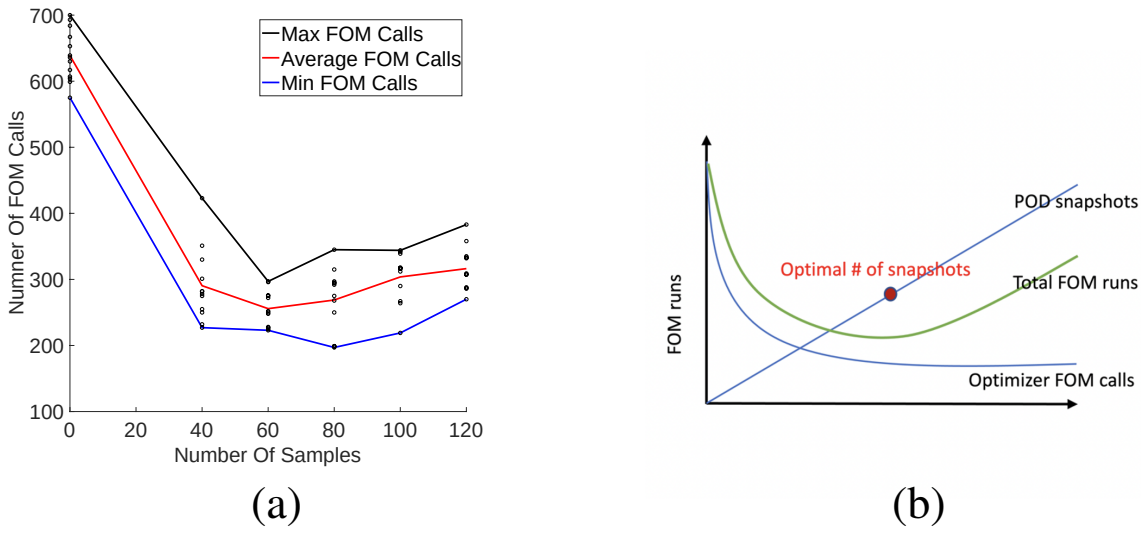


Figure 2.23. (a)FOM runs required in inverse problem to satisfy optimizer's step tolerance of 5×10^{-4} when the optimizer was started at POD-ANN ROM recovered parameters Vs. # of samples used in POD-ANN. (b) Illustration on why the plot on the left has such shape.

2.13 Conclusion

We have shown that a model that combines POD and ANN is a very robust approach for estimating the solution to forward as well as inverse problem with very large number of parameters for a deforming mesh 3-D heat conduction problem. The FOM was reduced to a model of 85 unknowns, represented by the coefficients of POD. ANNs were trained with parameters as input and the coefficients of POD as output from which we achieved an ability to calculate POD modal coefficients for any given set of parameters very quickly. This in turn allowed a real time approximation of its corresponding field variable. We applied this methodology to a steady state linear heat conduction in a hollow sphere with varying internal surface (governed by 20 parameters). Using POD-ANN approach, we were able to get temperature field within 3.5% of the FOM solution and all 20 parameters within 0.69% of the actual parameters.

We also investigated other approaches in inverse parameter estimation where ANNs were trained with parameters vs RMSNE or parameters vs the actual temperature field. POD-ANN approach proved more robust ROM than above methods. POD-ANN ROM worked successfully in inverse approximation even in the presence of large noises. Additionally, we studied the total FOM runs required to get the parameters within certain level of step size tolerance, both with and without utilization of POD-ANN ROM. Our findings revealed that using the proposed ROM could potentially save 72% or even higher number of FOM iterations in inverse approximation. We also suggested an improved sampling technique compared to Latin Hypercube Sampling and random sampling, even though there is room for improvement and we identify this as a topic of further investigation.

CHAPTER 3

Inverse Determination of Detailed Internal Flow And Temperature Fields from Wall Measurements: A Non-Linear Multiphysics Application.

3.1 Chapter Overview

The ROM methodology described in chapter 1 is now applied to the inverse problem of estimating field data based on wall measurements for conjugate heat and mass transfer, a multi-physics application that involves heat conduction in a solid body coupled with viscous fluid flow. The computational model consists of a heated sphere in the center of a cylindrical channel with forced convection(fig. 3.1). The model is split into two domains, one within the heated sphere where the heat conduction calculations are applied and one in the fluid domain where the viscous incompressible fluid flow calculations are applied.

The next section will describe the forward problem, and FOM solution procedure along with its governing equations. Since chapter 2 lays the foundation for chapter 3, some of the analysis will be skipped and the findings in chapter 2 will be directly applied here, such as selection of a method to calculate modal coefficients when the FOM solution is known. After FOM forward solution is described, the following section briefly summarizes

the metrics of the ROMs used. The next section discusses choice of number of modes followed by the analysis of the modal trajectories. Subsequently, the ROM results for both temperature and absolute velocity are compared with FOM solutions. Moving on, the next section discusses the sensitivity of the model to some simulated noise. Finally, the ROMs are updated to tackle the high sensitivity of this model to noise.

Full Order Model Forward Solution

The model consists of a heated sphere of radius (R_i) = 1 cm placed at the axial and longitudinal center of a circular channel of radius (R_o) = 2 cm and length (l_o) = 10 time the outer radius. The heated sphere, made up of carbon steel, is modeled as a uniform heat source emitting 100 W/cm^3 of heat into the fluid domain. Water enters the channel at inlet temperature of 25° C from one end and exits from the other end at a given flow rate, which is also a parameter of the model. The walls of the channel are insulated. The schematic of the problem is shown in figure 3.1

As the 3D geometry is axisymmetric (since the 3D geometry can be created by rotating the right or left domain along the centerline, fig.3.2), it is converted into a 2D axisymmetric model for the simulations. In figure 3.2, the centerline is the axis of symmetry and therefore, either the left or right side of of this centerline will be the fluid domain (shown in figure 4.2). The model is meshed with unstructured triangular elements (a zoomed in image of the mesh is shown in figure 4.2). Since, physics based and very fine mesh options are selected in commercial software COMSOL Multiphysics ®, mesh convergence analysis is skipped here. The boundary conditions applied are as follows- 1) mass flow rate (variable) and temperature of the fluid (constant, 25°C) at the inlet, 2) No-slip adiabatic condition for the channel walls, 3) No-slip boundary condition at heated body surface, 4) Open (Adia-

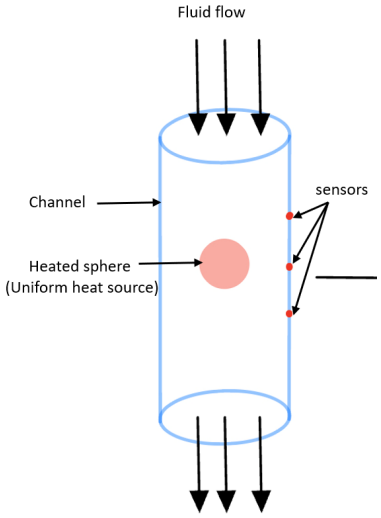


Figure 3.1. Schematic of domain.

batic) boundary with 0 gauge pressure at the outlet, and 5) axisymmetry boundary condition along the centerline.

The fluid flow (Navier-Stokes) and heat transfer (Energy) equations were solved using COMSOL ®. The fluid flow utilizes a laminar flow model and its governing equations are given by

$$\nabla \cdot \mathbf{u} = 0 \tag{3.1}$$

$$\rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot [-p\mathbf{I} + \mathbf{K}] + \mathbf{F} \tag{3.2}$$

$$\mathbf{K} = \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T) \tag{3.3}$$

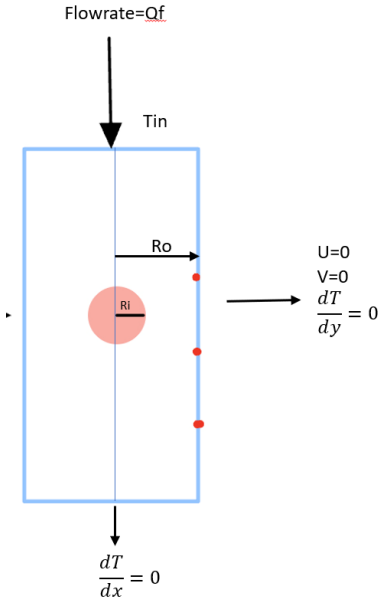


Figure 3.2. Boundary conditions of axisymmetric model.

Where $\mathbf{u}$ is the velocity vector, ρ is the density, p is the pressure, K is the viscous stress tensor, and F is the body force. The heat transfer equations are as follows,

$$\rho C_p \mathbf{u} \cdot \nabla T + \nabla \cdot \mathbf{q} = Q + Q_p + Q_{vd} \tag{3.4}$$

$$\mathbf{q} = -k\nabla T \tag{3.5}$$

Where C_p is the specific heat capacity at constant pressure, T is the temperature, $\mathbf{q}$ is conductive heat flux, Q is heat source, Q_p is point heat source, and Q_{vd} is viscous dissipation.

A stabilized FEM with triangular elements and linear basis function was used in COMSOL® to solve for laminar fluid flow and heat transfer. Sample solution for temperature and velocity fields are shown in figures 3.4 for an inlet flow rate of $1 L/min$.

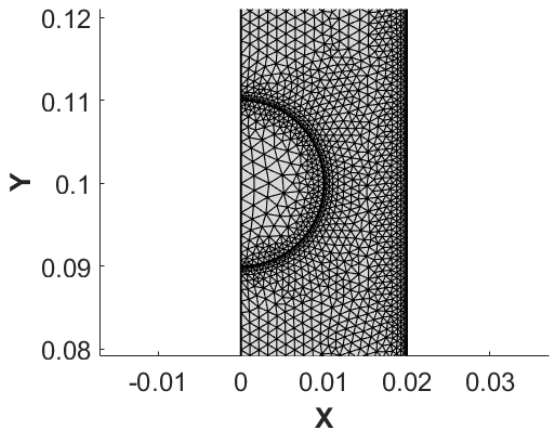


Figure 3.3. Unstructured mesh used for the model discritization

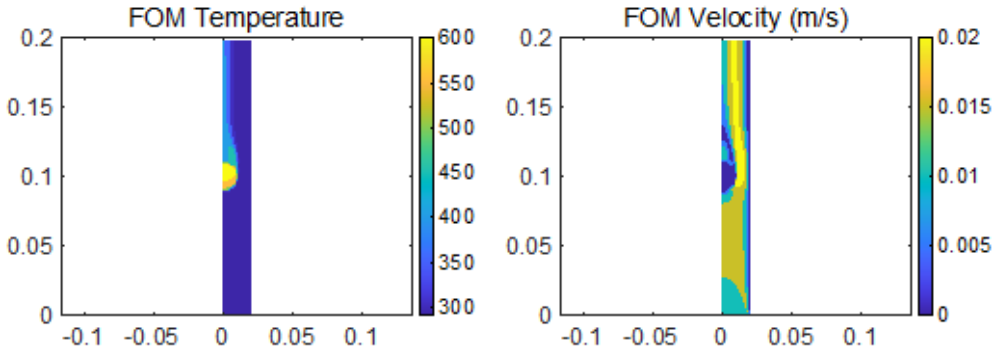


Figure 3.4. (a) Temperature field (b) Velocity field for inlet flow rate of $1L/min$

3.2 Methodology and Details of ROM

The snapshots, 50 in total, were created from FOM simulations that were run as steady-state flow for inlet flow rates from $0.1L/min$ up to $1.5L/min$. Once the FOM solutions were obtained, the temperatures at 3 different nodes (shown by red dots on fig. 3.2) on the internal surface of the channel were recorded for each snapshot. These temperature data will later be used as parameter sets of the FOM solution rather than the actual inlet flow rates. This was done to account for the fact that the flow rate information may not always be easily obtainable, such as in a nuclear power plant where engineers have to rely on some remotely measured data from the surface of a solid body. Using POD, 50 modes and their coefficients were calculated. Feed-forward ANNs were trained in commercial software MATLAB® with the sensor temperature data as input and the modal coefficients($\vec{D}$) as output using Levenberg-Marquardt algorithm [66] and logarithmic sigmoid transfer function for all hidden units. The number of hidden layers and its units were automated such that the root mean squared errors(rmse) on the validation data (25% of the total data used) were minimized. Each coefficient was trained in a separate network. Once successfully trained, any new sets of temperature data were then used as inputs to the ANNs and the ANNs gave the corresponding $\vec{D}$, which were further used with POD modes to approximate the full field variable. Two separate ROMs were built, each for temperature field and velocity field inside the channel, both parameterized by sensor temperatures.

3.3 Trajectories of Modal Coefficients

We started by extracting POD modes and coefficients following the procedures outlined in Chapter 1. In order to determine the optimal set of snapshots, we studied the trajectories of these modal coefficients taking into account the variations in the inlet flow

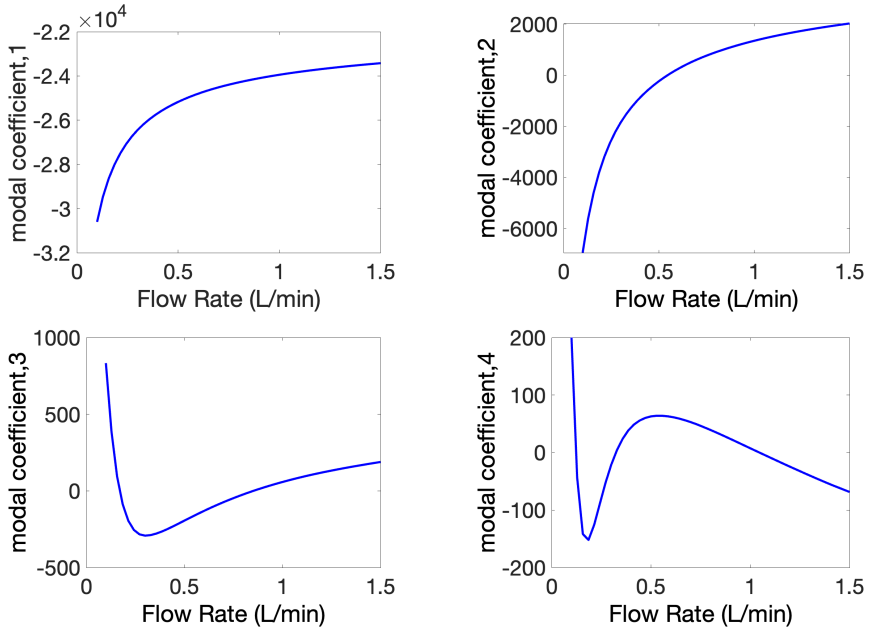


Figure 3.5. Modal trajectories for temperature.

rate. These trajectories are plotted in (fig. 3.5 and fig. 3.6) for temperature and velocity respectively as a function of inlet flow rate.

It's worth noting some key differences in the findings from chapter 2. The trajectories we observed were mostly linear (or very close), however, these trajectories, except for the first one for velocity, deviate significantly from the linear behaviour. Such behaviours are expected because of inheriantly non-linear behaviour of the problem at hand, as opposed to a linear case in chapter 2.

Observe the growing complexities of the trajectories as we move from 1^{st} to the 4^{th} one. Such intricacies introduce more burden for ANNs, therefore, it demands to truncate the size of mode, unlike in chapter 1 where we used all the available modes. As explained in chapter 1, doing so reduces the computational burden without compromising the ROM accuracy significantly. Furthermore, we see sharp shifts in trajectories around 0.3 to 0.5 L/min and therefore we sampled more snapshots around this section.

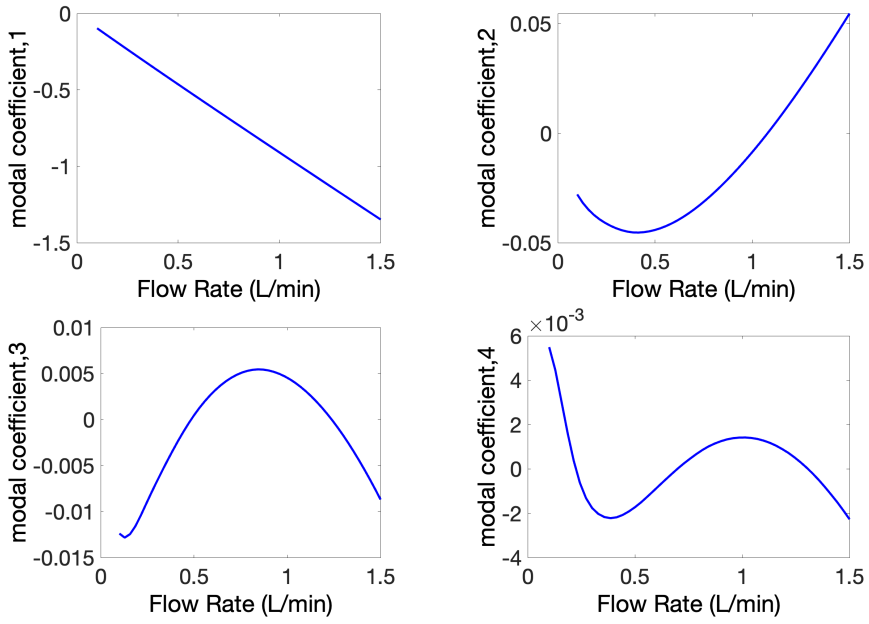


Figure 3.6. Modal trajectories for velocity.

3.4 Mode Size

The maximum percentage error and maximum rmse of the 50 snapshots as a function of number of modes were calculated for each snapshot. Then the maximum errors out of all the snapshots were extracted and plotted in figures 3.7 and 3.8 for different mode sizes. Since some of the nodal velocities are zero, the percentages calculated are with respect to the maximum value of the field variable. The truncation criteria for the number of modes was set such that the maximum percentage error on the plot was less than 0.5%, as such, from figures 3.7 and 3.8, 7 modes are selected for an ROM for temperature and 8 modes for an ROM for velocity.

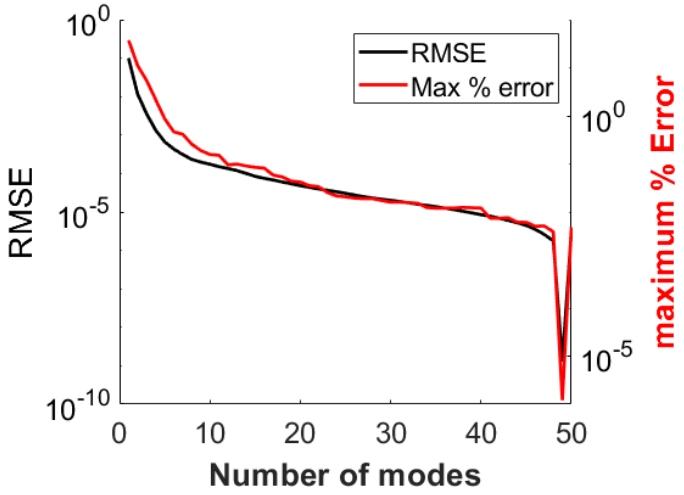


Figure 3.7. Maximum percent error and rmse as a function of number of modes for temperature.

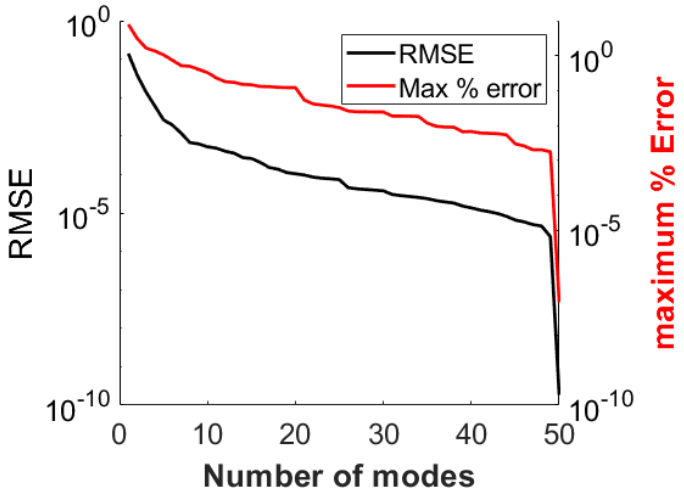


Figure 3.8. Maximum percent error and rmse as a function of number of modes for velocity.

3.5 Results and Discussion

As stated, separate ROMs for temperature and velocity were built. The FOM and ROM solution for a flow rate of $Q = 1L/min$ (not included in the original snapshot matrix)

were evaluated. Figure 3.9 shows a comparison of the FOM and ROM temperature profiles, and the percent error (with respect to the maximum absolute value of the field variable) between the two. The ROM accurately captures the temperature profile in the domain and the error does not exceed more than 1 percent anywhere in the model. Similarly, figure 3.10 shows a comparison of the FOM and ROM velocity profiles, and the percent error between the two. Here too, the ROM accurately captures the velocity profile in the domain and in fact, the error is even smaller than the temperature profile at less than 0.2 percent across the domain.

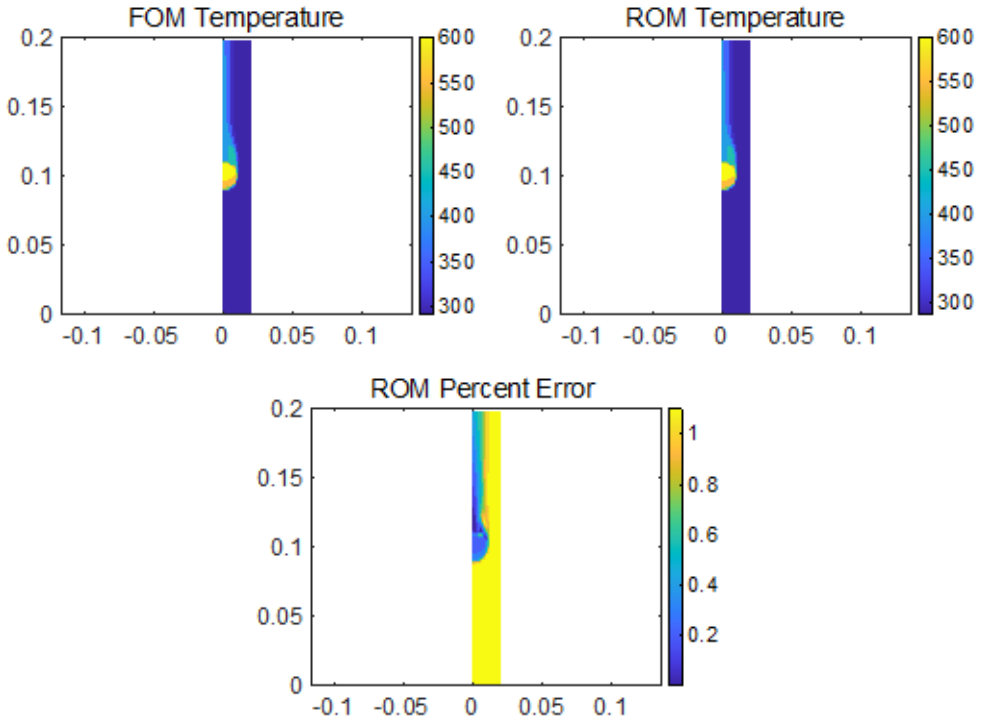


Figure 3.9. Temperature profile of FOM (top) and ROM (middle), and percent error between ROM and FOM (bottom) for $Q = 1L/min$.

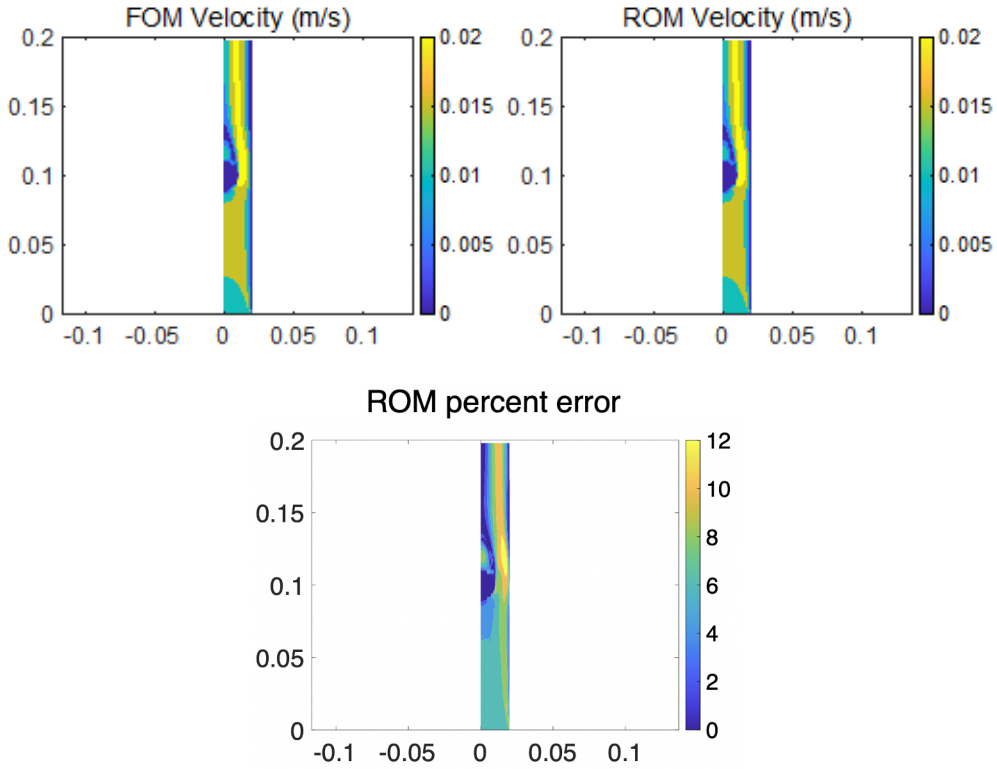


Figure 3.10. Velocity profile from FOM (top) and ROM (middle), and percent error between ROM and FOM (bottom) for $Q = 1L/min$.

To better understand the worst case scenarios, 50 more set of parameters were fed to the ROM and the results were compared against the FOM solutions, whose maximum percentage (w.r.to maximum absolute value of the field variable) errors in each snapshot are shown in box and whisker plot in figure 3.11. The errors do not exceed more than 8% of the maximum temperature value and 15% of the maximum velocity in any of the test cases. The results are fairly acceptable. We will investigate more on how to improve the accuracy of this results in the upcoming section.

The interesting behaviour worth taking a note is how well the POD-ANN ROM captured the coupling between the temperature field and the velocity field, since only the temperature information was used to predict the full velocity field.

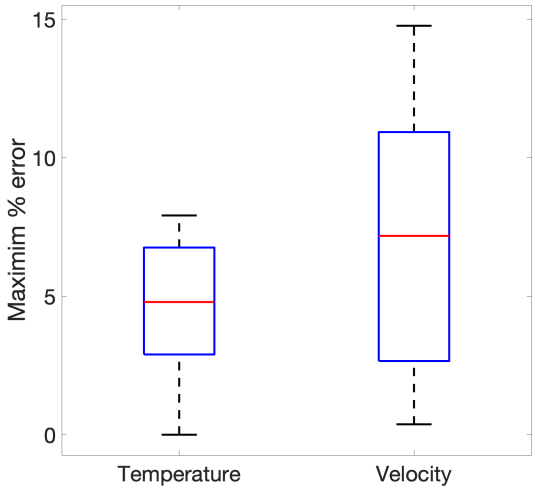


Figure 3.11. Box and whisker plot for maximum percentage error in ROM approximation for temperature and velocity.

3.6 Sensitivity Analysis

In order to study the ROM sensitivity to some simulated errors in the parameters, we introduced Gaussian noise to the parameters whose range is $\pm 1/1000\ ^{\circ}C$. Such noisy parameters were used to obtain the ROM solutions for the same above sets of 50 parameters, whose maximum percentage errors as compared to FOM solution are shown in figure 3.12.

Clearly, the errors increased very significantly even for such small noise in the parameters. To understand such high percentage errors in some of the approximations, we plotted the inlet flow rate vs the temperature profile for each of the sensor locations.

As shown in figure 3.13, the temperature profiles are already noisy. Such noise can be filtered out using different data smoothing techniques (one is shown here in figure 3.13 where the original parameters were reconstructed by taking a 5 point moving average). Then, the ROMs were built on this smoothed parameters, whose maximum percentage errors in the approximated solutions are shown in figure 3.14, which are still high.

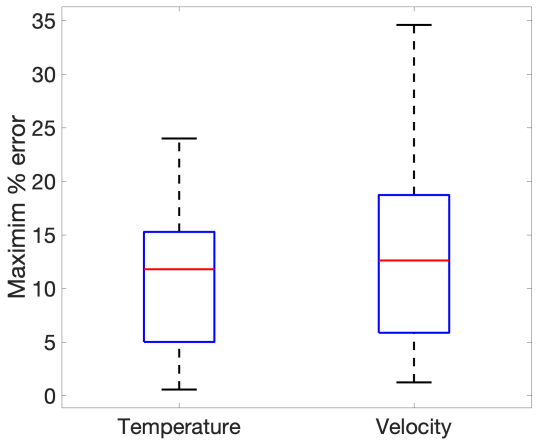


Figure 3.12. Maximum percentage errors for temperature and velocity using ROM approximation with added noise.

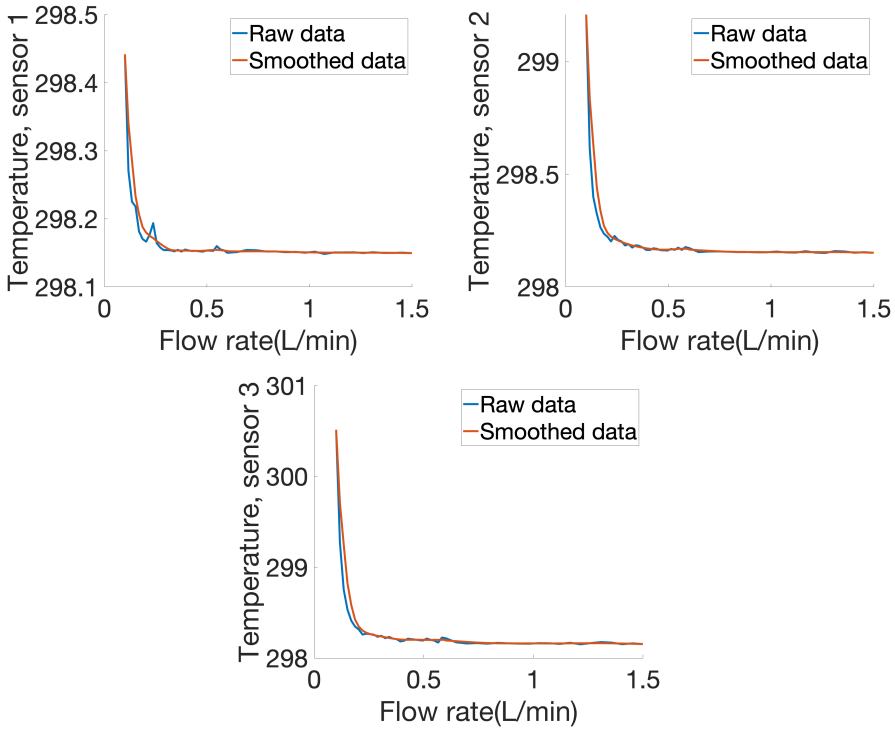


Figure 3.13. Temperature profile at the sensor locations for different flow rates.

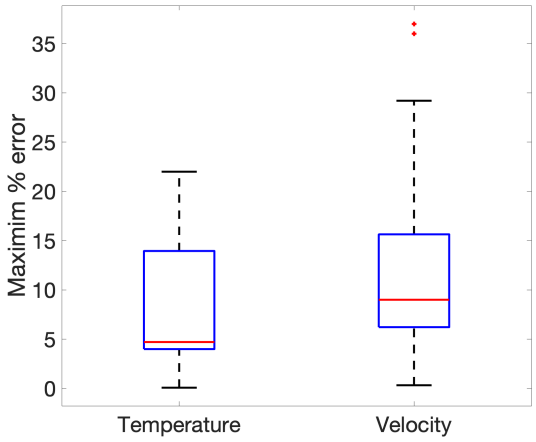


Figure 3.14. Maximum percentage errors in ROM approximation after smoothing the noisy input parameters.

3.7 Updated ROMs

Observe the range of the temperatures recorded by the sensors in figure 3.13, which is about half a degree only for the first sensor. Therefore, even a small deviation from actual measurement in the temperatures at the wall caused a significant error in the field approximation. As a result, smoothing the data did not help to improve the solution results even slightly. To circumvent this limitation, the location of the sensors were moved to the surface of the heated sphere (shown in figure 3.15), where the range of temperature change is higher (fig. 3.16) as compared to the temperatures at the wall and such parameters were used to update the above ROM.

The maximum percentage errors of temperature and velocity with the updated ROM for the same 50 parameter sets without any simulated noise are shown in figure 3.17. The plot clearly shows that a slight update in the location of sensors improved the accuracy of the ROM significantly.

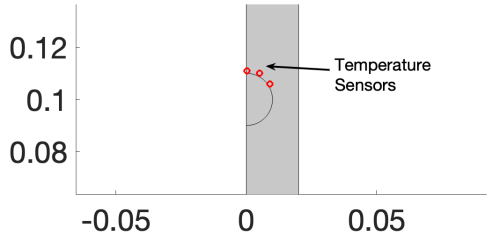


Figure 3.15. Updated locations of the sensors.

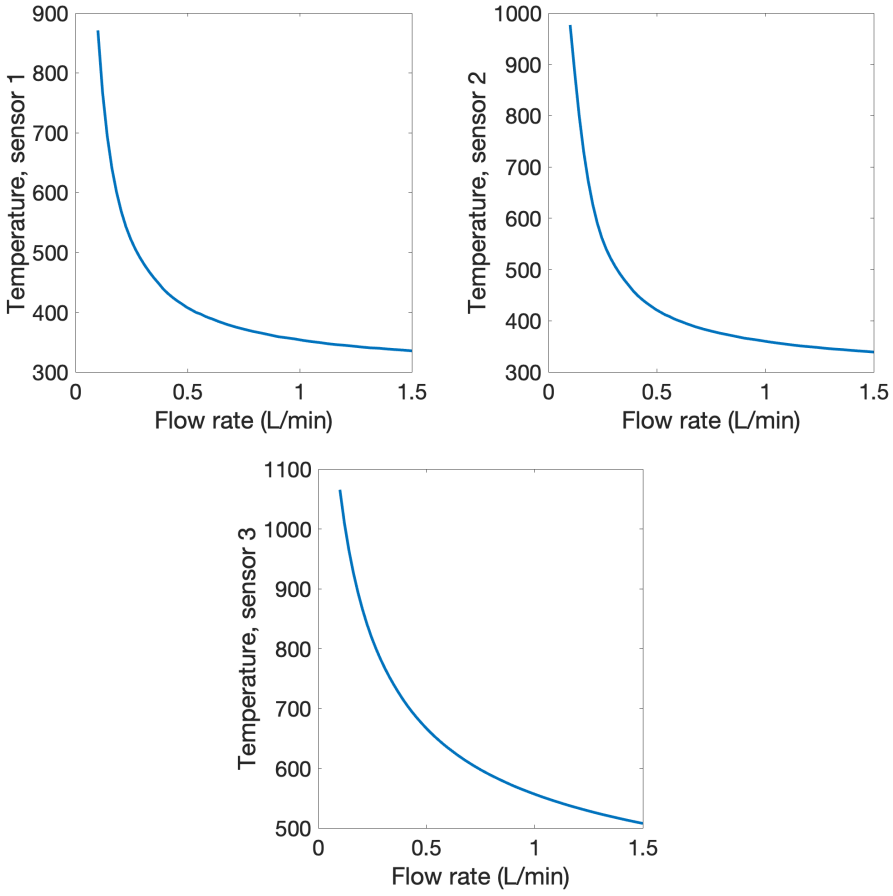


Figure 3.16. Temperature profiles at new sensor locations.

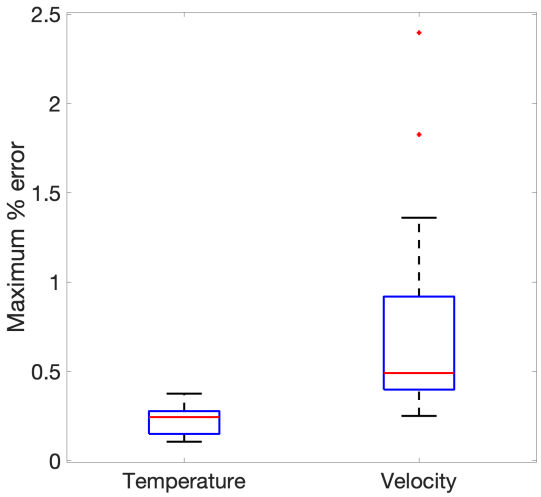


Figure 3.17. Maximum % error in temperature and velocity approximation with updated ROM without any simulated noise in the parameters.

Finally, we added different level of Gaussian noise to the test parameters and predicted the temperature and velocity field, the results of which are shown in figure3.18 and figure 3.19 respectively. In figures 3.18 and 3.19, the x-axis represents the maximum value of the Gaussian noise introduced in the parameters. As observed, the sensitivity of the measurement errors to the temperature at the new location decreased very dramatically. Such improvements are attributed to the fact that the range of new parameters are significantly higher which can tolerate high levels of noise.

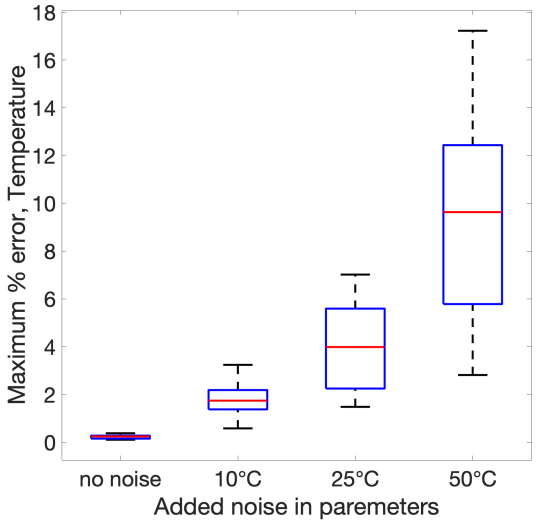


Figure 3.18. Maximum percentage errors in temperature for different levels of simulated noise in the parameters.

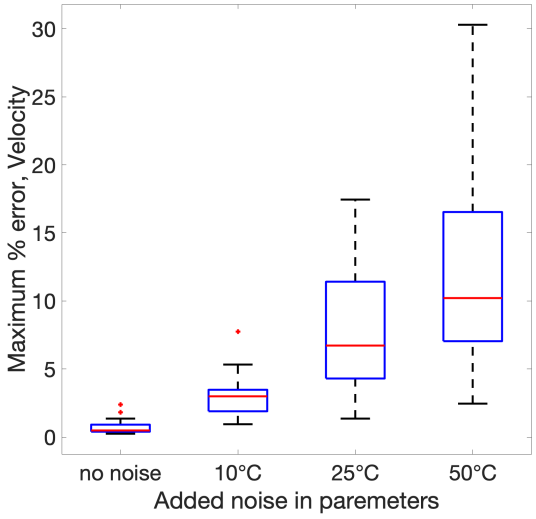


Figure 3.19. Maximum percentage errors in velocity for different noise levels in the parameters.

3.8 CONCLUSIONS

A non-intrusive reduced order model(ROM) based on proper orthogonal decomposition and artificial neural network was successfully applied in inversely estimating the temperature and velocity field inside a cylindrical channel with a heat source at its center in a conjugate multiphysics heat and mass transfer problem. Instead of using actual flow rate information, the nodal temperature information were utilized to predict the full fluid flow and temperature characteristics. The ROM based approach gave fairly satisfactory results with the sensor location on the surface of the channel but worked poorly with simulated noise, and very accurate results for both temperature and velocity field when the sensors were moved to the surface of the heated sphere. The updated ROM showed very robust results even in the presence of high levels of Gaussian noise in the parameters. Furthermore, this framework showed an excellent ability to capture the relationship between the two field variables, namely temperature and velocity, since temperature information was used to predict the full velocity field. Such results can be utilized in a nuclear power plant to non-invasively get the full temperature and velocity field.

CHAPTER 4

Inverse Airfoil Shape Identification: A Non-Linear Deforming Mesh Problem.

4.1 Chapter Overview

The ROM methodology described in chapter 1 is further applied to inversely identify the airfoil shapes parameterized by 4 digits National Advisory Committee for Aeronautics(NACA) parameters for a given pressure distribution on its surface and different angle of attacks(AOAs), for a 2D-viscous fluid flow problem.

The next section will describe the forward problem, and FOM solution procedure along with its governing equations. Like in Chapter 3, some of the analysis will be skipped, and the findings in Chapter 2 will be directly applied here, such as selection of a method to calculate modal coefficients when the FOM solution is known. After the FOM forward solution is described, the following section briefly summarizes the metrics of the ROMs used, including the choice of number modes. Sometimes, getting the trajectories of the modal coefficients can be costly, especially with large number of parameters. Therefore, we attempted to study the ROM application for this case without any prior knowledge of the modal trajectories. Subsequently, the ROM results for the forward problem are compared with FOM solutions. Moving on, the next section discusses the sensitivity of the model to

some simulated noise. Finally a breakthrough result of the ROM, its capability to perfectly denoise the small local minima of the objective function in inverse design, is presented.

4.2 Forward Problem

The forward problem is to solve for a 2D-viscous fluid flow to extract the pressure distribution on the surface of an airfoil for a given set of parameters that govern the airfoil geometry. A lot of airfoil parameterization techniques are available in the literature to create the airfoil geometry, such as the PARSEC method [67] or other parametric curve techniques such as splines, B-spline parameter [68], Bezier curves [69] and so on. For simplicity, the airfoil in this study was created using a 4-digit parametrization of the National Advisory Committee for Aeronautics (NACA) 4-digit parametrization. This airfoil depends on the mean camber line and the airfoil thickness (see fig.4.1), which are parameterized by three parameters: airfoil maximum camber(m), location of maximum camber(p), and the maximum thickness(t). Once these parameters are picked, the airfoil coordinates are generated using mathematical equations that are widely available in literature[70][71]. The mean camber line is given by equations 4.1.

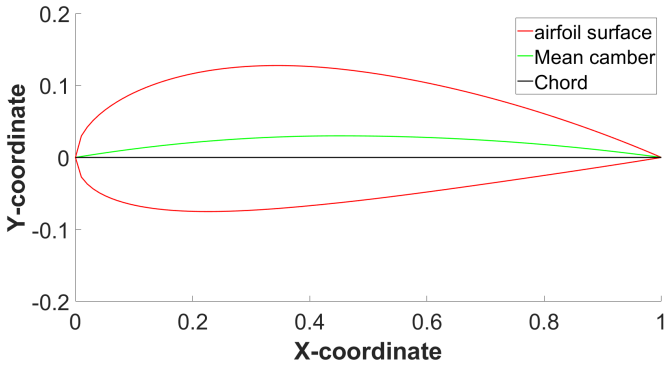


Figure 4.1. A sample airfoil for parameters in table 4.1.

$$y_c = \frac{m}{p^2}(2px - x^2) \qquad for\ x = 0\ to\ x = p \tag{4.1a}$$

$$y_c = \frac{m}{(1-p)^2}[(1-2p) + (2px - x^2)] \qquad for\ x = p\ to\ x = 1 \tag{4.1b}$$

Once the mean camber is determined, adding thickness to the mean camber location gives the y-coordinate of the airfoil. The thickness distribution for this airfoil is given by equations 4.2.

$$\pm y_t = \frac{t}{0.2}(0.2969\sqrt{x} - 0.3516x^2 + 0.2843x^3 - 0.1015x^4) \tag{4.2}$$

A set of NACA parameters are shown in Table 4.1. Figure 4.1 also shows the airfoil shape corresponding to the parameters in Table 4.1.

Table 4.1. A set of NACA airfoil parameters.

m	p	t
0.03	0.45	0.2

Once the airfoil surface co-ordinates are generated for a given parameter set, the geometry is imported and solved for pressure distribution using standard Finite Element Method (FEM) in commercial software 'COMSOL Multiphysics ®'. An $'OC'$ fluid domain is created and meshed with quadrilateral element with linear basis function as shown in figure 4.2. Since this is a deforming mesh problem, it is noted that the number of nodes and the order of node numbering are the same in every snapshot. The boundary conditions applied are as follows: i) The outer surface of the fluid domain is at horizontal velocity of $cos(\alpha)m/s$ and vertical velocity of $sin(\alpha)$, where α is the angle of attack(AOA) ii) No slip

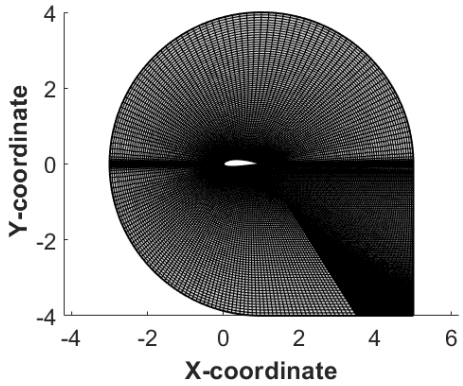


Figure 4.2. Fluid domain with 'OC' mesh for airfoil in table 4.1.

on the surface of the airfoil iii)0 gauge pressure at a point at location $(x,y) = (5,0)$. The fluid has a density of 1 kgm^{-3} and the dynamic viscosity of $1/1000$ $kgm^{-1}s^{-1}$. The fluid flow (Navier-Stokes) equations were solved in COMSOL ® using a laminar flow model whose governing equations are given by

$$\nabla \cdot \mathbf{u} = 0 \tag{4.3}$$

$$\rho(\mathbf{u} \cdot \nabla)\mathbf{u} \; = \; \nabla \cdot [-p\mathbf{I} + \mathbf{K}] + \mathbf{F} \tag{4.4}$$

$$\mathbf{K} \; = \; \mu(\nabla \mathbf{u} \; + \; (\nabla \mathbf{u})^T) \tag{4.5}$$

Where $\mathbf{u}$ is the velocity vector, ρ is the density, p is the pressure, K is the viscous stress tensor, and F is the body force. The pressure coefficient for the airfoil in figure 4.1, above boundary conditions and 0 AOA is shown in figure 4.3.

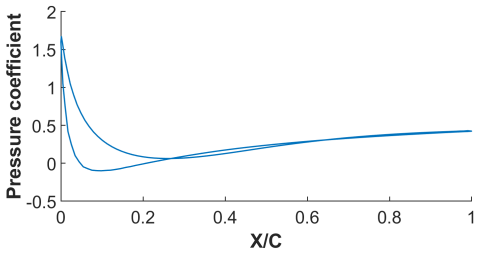


Figure 4.3. Pressure coefficient distribution on the surface of the airfoil in figure 4.1 for the specified boundary conditions.

4.3 ROM Creation and Application for Airfoil in Forward Problem.

The parametric space used to produce the airfoil geometry for POD snapshots in this study are as follows: i)the maximum camber is between 2% and 6% of the chord length ii)the location of maximum camber lies between 25% and 75% of the cord iii)the maximum thickness of the airfoil is greater than 12% and less than 30% of the chord length and iv)the range of AOA is between 0 and 10 degrees. Using random sampling of parameters, 200 snapshots are generated and used for POD. As pointed earlier, some of the POD modes can be excluded in the ROM without actually losing much accuracy. The cutoff mode size is determined by a rule that the approximated solution at each node for all the POD snapshots has absolute error of less than 1% (percentage calculated with respect to maximum pressure of each snapshot). The maximum percentage error and average rmse as a function of number of modes are shown in figure 4.4, from which 19 modes are selected for this study.

Separate feed-forward neural networks are trained for each of the 19 modal coefficients using Levenberg-Marquardt algorithm [72] and sigmoid transfer function. The number of hidden layers and units are automated such that the rmse on the validation data (5% of the total data) is minimum. Once the neural networks are fully trained, the POD coefficients can be easily determined for any parameter set in the training range. Finally,

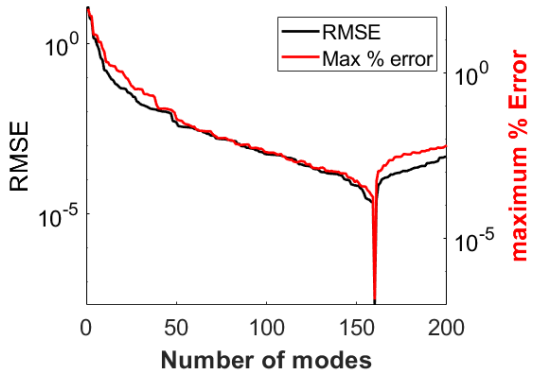


Figure 4.4. Average rmse and maximum percentage error in ROM approximation for different number of modes

the FOM solution is approximated using POD. Right sampling of the parameters is crucial to minimize the number of snapshots required in the ROM since the performance of POD and ANN depend on the sets of parameters chosen. Two widely used sampling techniques, random sampling and latin hypercube sampling (LHS), are tested against the author suggested new method, as described in chapter 2. In this new method, each parametric space is equally divided into the same number of points as the number of required samples. Then, each of the 4 parameters are randomly chosen from their respective set. Using each sampling technique, 3 ROMs are created with metrics outlined above and are used to approximate the surface pressure on 25 random airfoils and random AOAs. The box plot for maximum percentage errors in each test case are shown in figure 4.5. Clearly, the maximum percentage error in any of the 25 approximations do not exceed 7% with median error being less than 3% for any sampling technique. This means, the ROM is a robust predictor for this problem, with any sampling technique. Also, the results clearly show that the author suggested sampling technique is superior to random sampling and LHS as the maximum error is less than 4% with median error being less than 1.5%. Therefore, in the following sections, ROM built with author suggested sampling technique will be used.

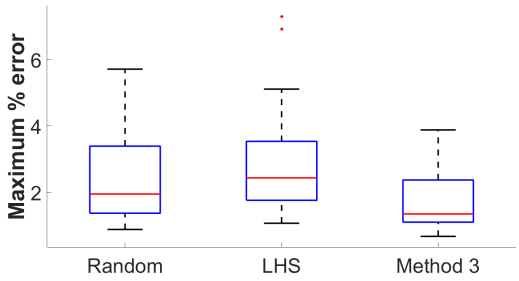


Figure 4.5. Box plot for maximum percent error on surface pressure for 25 random sets of parameters using ROM with different sampling techniques.

4.4 Inverse Approximation

The ROM built in the forward problem is further applied to recover the shape of the airfoil for a given target pressure on its surface. The objective function to minimize is the root mean squared distance between the target pressure and the inverse solution for a given set of parameters. 'fmincon' with 'sqp' algorithm is used in commercial software 'MATLAB' as the optimization tool. 'fmincon' is a non-linear gradient based optimization tool that works best for both linear and non-linear constrained problems. Sequential quadratic programming (sqp) [73] optimizes by setting up an optimal sub-problem which searches for next feasible point in the current iteration.

We tried to recover the airfoil shape of table 4.1 for both 1 degree AOA and 9 degrees AOA with no noise and also with Gaussian noise whose maximum value is 5% of the

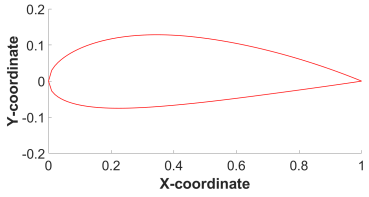
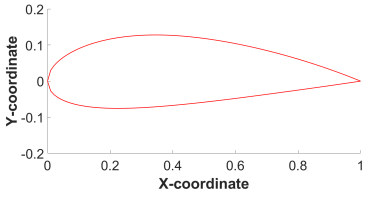
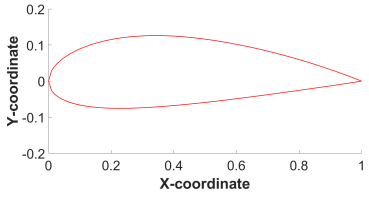
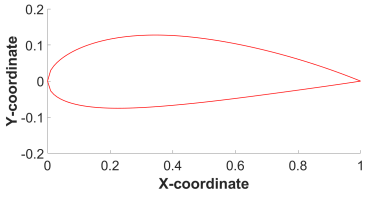
Table 4.2. Recovered parameters of table 4.1 and their % error using ROM for 1° AOA.

	Recovered parameters				percent error			
	m	p	t	AOA	m	p	t	AOA
0 noise	0.0306	0.4531	0.2007	0.9855	0.07	0.31	0.07	0.15
5% noise	0.0297	0.4639	0.2012	0.7678	0.03	1.39	0.12	2.32

Table 4.3. Recovered parameters of table 4.1 and their % error using ROM for 9° AOA.

	Recovered parameters				percent error			
	m	p	t	AOA	m	p	t	AOA
0 noise	0.0301	0.4559	0.2006	9.1917	0.01	0.59	0.06	1.91
5% noise	0.0312	0.4441	0.1964	9.0928	0.01	0.95	0.36	0.92

Table 4.4. Recovered airfoil shapes of figure 4.1 using ROM for different AOA and noise percentage.

	Recovered shape	
	1 ° AOA	9 ° AOA
0 noise		
5% noise		

maximum value of the target pressure. Table 4.2 shows the recovered parameters of table 4.1 and its percentage error in approximation for 1 degree AOA case. Similar results are shown in table 4.3 for 9 degrees AOA. The results show that this method is capable of recovering parameters with excellent accuracy in inverse problems even in the presence of large Gaussian noise. Note that the percentage errors calculated are with respect to chord length for errors in location and maximum AOA(10°) for errors in AOA. Visual representations of the recovered airfoil shapes for table 4.2 and table 4.3 are shown in table 4.4.

4.5 Computational Efficiency and Multiple Local Minima

In creating the ROM, 200 FOM solutions were required because of which the computational advantage of using this method over FOM iteration in inverse approximation is unknown. Therefore, we attempted to recover the parameters from table 4.1 and 1 degree AOA using same tools, algorithms and objective function as in ROM in the previous section but with the inverse solution being the FOM solution. From the recovered parameters, it was observed that the optimizer was stuck very close to the starting point for all 20 sts of initial guesses chosen, which clearly tells that the objective function has many local minima. To make sure that such observance is not a result of non optimal optimization algorithm, the value of the objective function was calculated for different values of camber and maximum camber location for a fixed value of maximum thickness(20% cord) and 1 degree AOA. The contour plot and the surface plot in figure 4.6 clearly shows that the objective function has many local minima. Note the very small range of parameters used in the plots, as such, it can be safely concluded that the number of local minima in full parameter space and for all 4 parameters is realistically too many to be handled computationally when used with FOM in inverse approximation. The inverse problem that practically seems impossible to solve using FOM with current technology is solved with high accuracy with only 200 FOM solutions when applied with POD-ANN ROM. In situations where better accuracies are desired, it can be safely concluded that the computational cost of increasing few snapshots in ROM is very minimal in comparison to the number of FOM iterations that would otherwise be required when used with FOM solution as inverse solution.

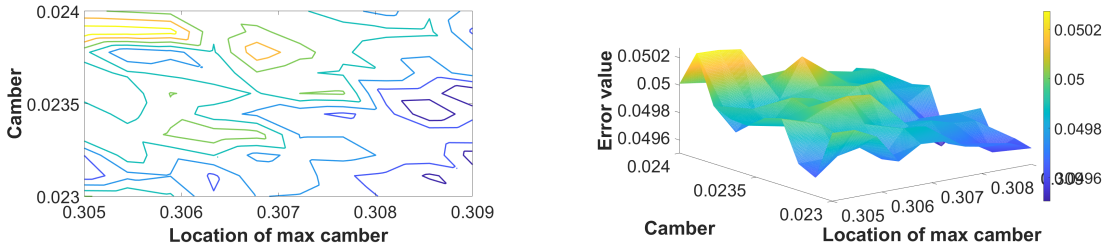


Figure 4.6. Contour plot(left) and surface plot(right) for the error objective function as a function of camber and location of maximum camber when thickness and AOA are fixed.

4.6 Conclusion

It is shown that a ROM that combines POD and ANN is a very robust approach in estimating the solution in forward as well as inverse problems for a non-linear deforming mesh fluid flow problem of inversely identifying the airfoil shapes. The dimensionality of the FOM was reduced to a model of 19 unknowns using POD. Then the parameters and their unknowns (modal coefficients) were trained using feed-forward neural networks so that a quick approximation of the unknowns was possible in real time for any given parameter set in the training range. This ROM was applied to predict the surface pressure on NACA 4 digit airfoils for different AOA cases. The model predicted the surface pressure very accurately with maximum error being less than 4% on all 25 test cases for the forward problem. In comparing random sampling and Latin hypercube sampling techniques with the author's suggested method, it was demonstrated that the author's approach outperforms these alternatives for selecting samples in ROMs. The same framework was also used in the inverse approximation to recover the NACA airfoil parameters and the AOA for a given target surface pressure distribution. The predictions were very accurate that the recovered parameters and AOA were within 2.5% of the actual values even in the presence of large Gaussian noise (5% of the highest pressure value). The key discovery in this

paper is that the model effectively removes the small local minima(or noise) in the error objective function. This achievement enables the solution of an otherwise computationally very extensive (or practically infeasible) problem using the FOM solution in an inverse problem.

CHAPTER 5

Concluding Remarks

In this thesis, three distinct applications of the POD-ANN ROM were explored in the field of computational dynamics for fluid flow and heat transfer.

The first study investigates the applicability of this ROM for a 3D-linear heat conduction in a deforming mesh problem in a hollow pipe, with a random internal geometry. The internal surface of the pipe was parameterized by a set of 20 parameters and provided with a fixed temperature. The outside surface was exposed to a constant temperature, while the side walls were insulated. The ROM accurately predicted the full temperature field, with errors being less than 3.5% of the actual values. This chapter addressed various aspects of the ROM that are applicable across diverse scenarios. These encompassed considerations like sample size, mode size, sampling techniques, and sample size selection, which formed the foundational principles for the subsequent chapters. Using the temperature information on the surface of the pipe, the framework was applied with optimization tools for inverse parameter approximations, whose errors did not exceed more than 0.7% of the actual parameter values for all 20 parameters. This framework worked perfectly even with high Gaussian noise introduced in the target temperature in inverse approximation. This method was compared against other commonly used approaches that leverage directly on machine learning, such as direct training of ANNs with parameters and the whole temper-

ature field. Also, the errors between the temperature on the surface for actual parameters and the available surface temperature for other parameters were trained with their corresponding parameter sets, and we sought to find the parameters with minimum error. The POD-ANN ROM approach outperformed these approaches by far margins. Furthermore, the author suggested sampling technique was found to be more stable and accurate sampling technique than Latin Hypercube Sampling (LHS) or Random Sampling.

Secondly, the ROM was applied to an inverse non-invasive determination of detailed internal flow and temperature for a non-linear conjugate heat and mass transfer multiphysics problem inside a cylindrical channel with a spherical heat source at its longitudinal and axial center for different inlet flow rates. This axisymmetric model was converted into a 2D model for simplicity. Instead of parameterizing the case with the actual flow rates, three temperature sensors were installed on the surface of either the cylinder or the sphere. Such parameterizing was chosen to address the issue that the flow rate information may not always be available, such as in a nuclear power plant. The full temperature and absolute velocity were predicted for 50 sets of sensor temperatures, both with and without simulated noise in the sensor temperatures. With the sensor locations on the surface of the cylindrical channel, the accuracies were satisfactory, however, such errors grew exponentially even with very small Gaussian noise. The ROM was then updated with sensors on the surface of the sphere, where the range of temperature was comparatively higher for the given range of flowrates. This slight update in the ROM made huge differences since the predictions were very accurate (temperature within 0.5% and velocity within 2.5%), and showed very robust behaviour with very high levels of Gaussian noise. Furthermore, this framework showed an excellent ability to capture the relationship temperature and velocity, since only temperature information was utilized to predict the full velocity field.

Finally, the ROM was applied for a non-linear deforming mesh fluid flow problem to predict the pressure field over NACA airfoils for different angle of attacks, and to inversely

identify the airfoil shapes for a given surface pressure distribution. The model predicted the surface pressure very accurately with maximum error being less than 4% for all 25 test cases. Here too, the author developed sampling technique proved to be more robust technique than LHS and Random Sampling. The predictions demonstrated remarkable accuracy, with recovered parameters and AOA within 2.5% of actual values, even when confronted with substantial Gaussian noise (equivalent to 5% of the highest pressure value). AN important finding in this application is, this framework is capable of removing small local minima (or noise) in the error objective function. Such discovery allows the solution in inverse approximation with only few hundred FOM solutions, which would otherwise be substantially high(or even computationally infeasible).

CHAPTER 6

Future Work

Through 3 different applications, the usefulness of POD-ANN ROM was shown for fluid flow and heat transfer problems. The first problem was a linear 3D heat conduction. The second application dealt with a multiphysics, non-linear axisymmetric problem, later converted to a 2D model. Ultimately, the third application was for a 2D non-linear deforming mesh problem. This framework's application and reliability are not tested for a 3D non-linear deforming mesh. Therefore, some future work will be dedicated to the applications of this novel concept to a wing optimization, a 3D-non linear deforming mesh application. Similarly, its suitability for addressing multiphysics problems warrants comprehensive exploration. Thus, further research in this direction will be conducted in the future.

We have shown the robustness of POD-ANN framework for diverse areas of applications. Even though we outlined the pros of using ANNs over other statistical method in the opening chapter, such comparisions are not quantified. In future research, a comparative analysis will be undertaken to assess the performance of POD-ANN against POD-Galerkin and other response surface methods like POD-kriging.

The ROM predicted solutions are very accurate for forward problems and inverse problems. Therefore, the ultimate future research is to develop an algorithm that will be

available for commercial software for general inverse design for predicting the initial parameters or initial solution for non-linear solvers.

Appendix

The Galerkin Projection methodology used in the determination of POD coefficients, initially devised by Dennis and colleagues, employs the Finite Element Method (FEM) to address heat transfer problems, as depicted in detail in their work [40]. This work minimizes the residual of the PDE in weak sense. The images presented below have been faithfully reproduced from the original work authored by Dennis and team.

The unknown coefficients are obtained using the Galerkin method. In this approach, $\vec{D}$ is determined such that the residual of the PDE is minimized in the weak sense. The Galerkin method uses a weight function W_R composed of the same basis functions used in the approximation function T_R though with a different set of unknown coefficients $\vec{F}$.

$$W_R = \vec{F}^T \vec{\phi} \tag{10}$$

A functional π is formed by integrating the residual of the PDE evaluated with T_R (denoted as $R(T_R)$) and weighted by W_R over the entire domain Ω.

$$\pi = \int_\Omega W_R R(T_R) d\Omega \tag{11}$$

The combined vectors $[\vec{u}]$ are multiplied by the vector of nodal basis functions $\vec{\psi}$ from the FOM to obtain a new vector of m basis functions $\vec{\phi}$.

$$\vec{\phi}^T = \vec{\psi}^T [\vec{u}] \tag{8}$$

These new basis functions can be used to form an approximation function for temperature, T_R, by multiplying the functions by a vector of unknown coefficients $\vec{D}$.

$$T_R = \vec{\phi}^T \vec{D} \tag{9}$$

Unlike nodal basis functions typically used in FEM, these coefficients do not represent the temperature at a specific nodes in a mesh. Rather, they are global coefficients with no immediate physical interpretation. In that sense, the functions $\vec{\phi}$ could be referred to as modal basis functions.

The unknown coefficients are obtained using the Galerkin method. In this approach, $\vec{D}$ is determined such that the residual of the PDE is minimized in the weak sense. The Galerkin method uses a weight function W_R composed of the same basis functions used in the approximation function T_R though with a different set of unknown coefficients $\vec{F}$.

$$W_R = \vec{F}^T \vec{\phi} \tag{10}$$

A functional π is formed by integrating the residual of the PDE evaluated with T_R (denoted as $R(T_R)$) and weighted by W_R over the entire domain Ω.

$$\pi = \int_\Omega W_R R(T_R) d\Omega \tag{11}$$

The functional is minimized with respect to the weight coefficients $\vec{F}$, which leads to a coupled system of m equations.

$$\frac{\partial \pi}{\partial \vec{F}} = \int_{\Omega} \vec{\phi} R(T_R) d\Omega = 0 \tag{12}$$

In our application, we are simulating heat conduction which is governed by the convection-diffusion equation

$$R(T_R) = \frac{\partial T_R}{\partial x} - \frac{1}{Pe} \nabla^2 T_R = 0 \tag{13}$$

Note that the time derivative is expressed as material derivative and contains the effect of the convection term. Substituting Eqn 13 into Eqn 12 and using the definition of T_R results in following system of linear equations.

$$\int_{\Omega} \left(\vec{\phi} \frac{\partial \vec{\phi}}{\partial x}^T - \frac{1}{Pe} \vec{\phi} \nabla^2 \vec{\phi}^T \right) \vec{D} d\Omega = 0 \tag{14}$$

The above equation can be expressed more compactly in the matrix form

$$[K_{ROM}] \vec{D} = 0 \tag{15}$$

where $[K_{ROM}]$ is the conduction matrix. This coupled system of m equations can be readily solved numerically by standard linear system techniques.

Forming the matrices in Eqn 15 requires the evaluation of the integrals in Eqn 14, which would be non-trivial for complex domain shapes.

In our approach, the integrals in Eqn 14 are evaluated numerically using the existing FOM finite element mesh. We show the formation of the conduction matrix below to illustrate how this is accomplished.

$$[K_{ROM}] = \int_{\Omega} \vec{\phi} R(\vec{\phi}^T) d\Omega = \int_{\Omega} [\bar{u}]^T \vec{\psi} R(\vec{\psi}^T)[\bar{u}] d\Omega =$$
$$[\bar{u}]^T \left(\sum \int_{\Omega_e} \vec{\psi}_e R(\vec{\psi}_e{}^T) d\Omega_e \right) [\bar{u}] = [\bar{u}]^T \left(\sum [K_e] \right) [\bar{u}] \quad (16)$$

Here Ω_e represents a single finite element, $\vec{\psi}_e$ is the elemental basis function vector, $[K_e]$ is the elemental conduction matrix, and $\sum [K_e]$ denotes the assembly of all elemental conduction matrices into a global matrix.

Equation 16 shows that $[K_{ROM}]$ can be formed directly from the FOM global conduction matrix using the POD derived vectors $[\bar{u}]$ as transformation matrices.

Thus, we can express the ROM system of equations in terms of the global FOM matrix $[K]$. We note that the matrix is readily available as it is formed when generating the snapshots.

$$[\vec{u}]^T[K][\vec{u}]\vec{D} = 0 \tag{17}$$

For a linear PDE with constant coefficients, the resulting ROM matrices can simply be formed once when the snapshots are generated and then stored for later use. Non-linear systems will require updating the matrices at each non-linear iteration if Newton-Raphson method is used.

This approach implicitly enforces essential boundary conditions in the ROM system when we work with constrained FOM global matrices. Natural boundary conditions are enforced using the global FOM heat load vector $\vec{Q}$

$$\vec{Q} = \sum \int_{\Gamma_e} \vec{\psi}_{\Gamma_e} q_n d\Gamma_e \tag{18}$$

Where Γ_e is the element boundary face, $\vec{\psi}_{\Gamma_e}$ are the boundary element shape functions, and q_n is the heat flux component normal to the boundary element.

The ROM heat load vector is formed using $[\vec{u}]$ as a transformation matrix operating on $\vec{Q}$.

$$\vec{Q}_{ROM} = [\vec{u}]^T\vec{Q} \tag{19}$$

The final ROM system of equations including natural boundary conditions is given below. We note that in this paper, we are concerned only with the steady-state solution. Therefore, $[C] = 0$

$$[\bar{u}]^T [K][\bar{u}]\vec{D} = [\bar{u}]^T \vec{Q} \tag{20}$$

In the case of a heat flux that varies parametrically, Eqns 18-19 are reevaluated for each parameter value. This requires reintegration and reassembly of boundary elements associated with the applied heat flux. Optimized code implementations of these steps are required for the ROM to be computationally efficient.

Once the coefficients $\vec{D}$ are obtained, the temperature function defined over the FOM mesh can be readily recovered as shown below.

$$T = \vec{\psi}^T \vec{T}_{FOM} = \vec{\psi}^T [\bar{u}]\vec{D} \tag{21}$$

The temperature at the nodal points on the FOM mesh are also easily computed with $\vec{T}_{FOM} = [\bar{u}]\vec{D}$. With this information, the ROM solution can be postprocessed on the FOM mesh and directly compare with the FOM solution on a node-by-node basis, if desired.

REFERENCES

[1] O. San, R. Maulik, and M. Ahmed, "An artificial neural network framework for reduced order modeling of transient flows," *Communications in Nonlinear Science and Numerical Simulation*, vol. 77, pp. 271–287, 2019.

[2] R. Swischuk, L. Mainini, B. Peherstorfer, and K. Willcox, "Projection-based model reduction: Formulations for physics-based machine learning," *Computers & Fluids*, vol. 179, pp. 704–717, 2019.

[3] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Dynamic mode decomposition with control," *SIAM Journal on Applied Dynamical Systems*, vol. 15, no. 1, pp. 142–161, 2016.

[4] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of fluid mechanics*, vol. 656, pp. 5–28, 2010.

[5] B. Haasdonk, "Reduced basis methods for parametrized pdes–a tutorial introduction for stationary and instationary problems," *Model reduction and approximation: theory and algorithms*, vol. 15, p. 65, 2017.

[6] P. German, M. Tano, J. C. Ragusa, and C. Fiorina, "Comparison of reduced-basis techniques for the model order reduction of parametric incompressible fluid flows," *Progress in Nuclear Energy*, vol. 130, p. 103551, 2020.

[7] R. Everson and L. Sirovich, "Karhunen-loeve procedure for gappy data," *Journal of the Optical Society of America A*, vol. 12, no. 8, pp. 1657–1664, 2000.

[8] S. K. Suman and A. Kumar, "Investigation and implementation of model order reduction technique for large scale dynamical systems," *Archives of Computational Methods in Engineering*, pp. 1–22, 2022.

[9] V. Simoncini and D. B. Szyld, "Recent computational developments in krylov subspace methods for linear systems," *Numerical Linear Algebra with Applications*, vol. 14, no. 1, pp. 1–59, 2007.

[10] F. Deluzet, G. Fubiani, L. Garrigues, C. Guillet, and J. Narski, "Sparse grid reconstructions for particle-in-cell methods," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 56, no. 5, pp. 1809–1841, 2022.

[11] S. Volkwein and B. Haasdonk, "Proper orthogonal decomposition: Theory and reduced-order modeling," *Lecture Notes, University of Stuttgart, Germany*, vol. 1, no. 1, pp. 43–65, 2002.

[12] Y. LIANG, H. LEE, S. LIM, W. LIN, K. LEE, and C. WU, "Proper orthogonal decomposition and its applications—part i: Theory," *Journal of Sound and Vibration*, vol. 252, no. 3, pp. 527–544, 2002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022460X01940416

[13] G. Kerschen, J.-C. Golinval, A. VAKAKIS, and L. BERGMAN, "The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview," *Nonlinear Dynamics*, vol. 41, pp. 147–169, 08 2005.

[14] A. Fic, R. A. Białecki, and A. J. Kassab, "Solving transient nonlinear heat conduction problems by proper orthogonal decomposition and the finite-element method," *Numerical Heat Transfer, Part B: Fundamentals*, vol. 48, no. 2, pp. 103–124, 2005.

[15] N. Aubry, "On the hidden beauty of the proper orthogonal decomposition," *Theoretical and Computational Fluid Dynamics*, vol. 2, no. 5-6, pp. 339–352, 1991.

[16] T. R. Smith, J. Moehlis, and P. Holmes, "Low-dimensional modelling of turbulence using the proper orthogonal decomposition: a tutorial," *Nonlinear Dynamics*, vol. 41, pp. 275–307, 2005.

[17] N. Aubry, P. Holmes, J. L. Lumley, and E. Stone, "The dynamics of coherent structures in the wall region of a turbulent boundary layer," *Journal of fluid Mechanics*, vol. 192, pp. 115–173, 1988.

[18] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *Journal of fluid mechanics*, vol. 641, pp. 115–127, 2009.

[19] T. R. BEWLEY, P. MOIN, and R. TEMAM, "Dns-based predictive control of turbulence: an optimal benchmark for feedback algorithms," *Journal of Fluid Mechanics*, vol. 447, p. 179–225, 2001.

[20] H. V. Ly and H. T. Tran, "Modeling and control of physical processes using proper orthogonal decomposition," *Mathematical and Computer Modelling*, vol. 33, no. 1, pp. 223–236, 2001, computation and control VI proceedings of the sixth Bozeman conference. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0895717700002405

[21] M. Fogleman, J. Lumley, D. Rempfer, and D. Haworth, "Application of the proper orthogonal decomposition to datasets of internal combustion engine flows," *Journal of Turbulence*, vol. 5, no. 1, p. 023, jun 2004. [Online]. Available: https://dx.doi.org/10.1088/1468-5248/5/1/023

[22] K. Lu, Y. Jin, Y. Chen, Y. Yang, L. Hou, Z. Zhang, Z. Li, and C. Fu, "Review for order reduction based on proper orthogonal decomposition and outlooks of applications in mechanical systems," *Mechanical Systems and Signal Processing*, vol. 123, pp. 264–297, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0888327019300184

[23] S. HAN and B. FEENY, "Application of proper orthogonal decomposition to structural vibration analysis," *Mechanical Systems and Signal Processing*, vol. 17, no. 5, pp. 989–1001, 2003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0888327002915700

[24] F. Bamer and C. Bucher, "Application of the proper orthogonal decomposition for linear and nonlinear structures under transient excitations," *Acta Mechanica*, vol. 223, pp. 2549–2563, 2012.

[25] M. Thiene, M. Zaccariotto, and U. Galvanetto, "Application of proper orthogonal decomposition to damage detection in homogeneous plates and composite beams," *Journal of Engineering Mechanics*, vol. 139, no. 11, pp. 1539–1550, 2013.

[26] D. Kumar, C. Rai, and S. Kumar, "Principal component analysis for data compression and face recognition," *INFOCOMP Journal of Computer Science*, vol. 7, no. 4, pp. 48–59, 2008.

[27] L. S. Ting, D. Y. F. Weng, and N. B. A. Manap, "A novel approach for arbitrary-shape roi compression of medical images using principal component analysis (pca)," *Trends in Applied Sciences Research*, vol. 10, no. 1, p. 68, 2015.

[28] F. L. Gewers, G. R. Ferreira, H. F. D. Arruda, F. N. Silva, C. H. Comin, D. R. Amancio, and L. d. F. Costa, "Principal component analysis: A natural approach to data exploration," *ACM Computing Surveys (CSUR)*, vol. 54, no. 4, pp. 1–34, 2021.

[29] A. Bakhshinejad, A. Baghaie, A. Vali, D. Saloner, V. L. Rayz, and R. M. D'Souza, "Merging computational fluid dynamics and 4d flow mri using proper orthogonal decomposition and ridge regression," *Journal of biomechanics*, vol. 58, pp. 162–173, 2017.

[30] G. Janiga, "Novel feature-based visualization of the unsteady blood flow in intracranial aneurysms with the help of proper orthogonal decomposition (pod)," *Computerized Medical Imaging and Graphics*, vol. 73, pp. 30–38, 2019.

[31] M. Balasubramanian, C. Bowd, R. N. Weinreb, G. Vizzeri, L. M. Alencar, P. A. Sample, N. O'Leary, and L. M. Zangwill, "Clinical evaluation of the proper orthogonal decomposition framework for detecting glaucomatous changes in human subjects," *Investigative ophthalmology & visual science*, vol. 51, no. 1, pp. 264–271, 2010.

[32] E. Samadiani and Y. Joshi, "Reduced order thermal modeling of data centers via proper orthogonal decomposition: a review," *International Journal of Numerical Methods for Heat & Fluid Flow*, vol. 20, no. 5, pp. 529–550, 2010.

[33] F. Rulli, S. Fontanesi, A. d'Adamo, and F. Berni, "A critical review of flow field analysis methods involving proper orthogonal decomposition and quadruple proper orthogonal decomposition for internal combustion engines," *International Journal of Engine Research*, vol. 22, no. 1, pp. 222–242, 2021.

[34] R. Manthey, A. Knospe, C. Lange, D. Hennig, and A. Hurtado, "Reduced order modeling of a natural circulation system by proper orthogonal decomposition," *Progress in Nuclear Energy*, vol. 114, pp. 191–200, 2019.

[35] L. Vergari, A. Cammi, and S. Lorenzi, "Reduced order modeling for coupled thermal-hydraulics and reactor physics problems," *Progress in Nuclear Energy*, vol. 140, p. 103899, 2021.

[36] P. German, M. Tano, C. Fiorina, and J. C. Ragusa, "Gen-rom—an openfoam®-based multiphysics reduced-order modeling framework for the analysis of molten salt reactors," *Progress in Nuclear Energy*, vol. 146, p. 104148, 2022.

[37] D. J. Lucia, P. S. Beran, and W. A. Silva, "Reduced-order modeling: new approaches for computational physics," *Progress in aerospace sciences*, vol. 40, no. 1-2, pp. 51–117, 2004.

[38] R. Zimmermann, A. Vendl, and S. Görtz, "Reduced-order modeling of steady flows subject to aerodynamic constraints," *AIAA journal*, vol. 52, no. 2, pp. 255–266, 2014.

[39] C. J. Ruscher, J. F. Dannenhoffer III, and M. N. Glauser, "Repairing occluded data for a mach 0.6 jet via data fusion," *AIAA journal*, vol. 55, no. 1, pp. 255–264, 2017.

[40] B. H. Dennis, A. Akbariyeh, J. Michopoulos, F. Komninelli, and A. Iliopoulos, "Inverse determination of moving heat flux distributions using reduced order models," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 46285. American Society of Mechanical Engineers, 2014, p. V01AT02A026.

[41] X. Sun, X. Pan, and J.-I. Choi, "Non-intrusive framework of reduced-order modeling based on proper orthogonal decomposition and polynomial chaos expansion," *Journal of Computational and Applied Mathematics*, vol. 390, p. 113372, 2021.

[42] H. N. Najm, "Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics," *Annual review of fluid mechanics*, vol. 41, pp. 35–52, 2009.

[43] S. Abraham, M. Raisee, G. Ghorbaniasl, F. Contino, and C. Lacor, "A robust and efficient stepwise regression method for building sparse polynomial chaos expansions," *Journal of Computational Physics*, vol. 332, pp. 461–474, 2017.

[44] C. Audouze, F. De Vuyst, and P. B. Nair, "Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations," *Numerical Methods for Partial Differential Equations*, vol. 29, no. 5, pp. 1587–1628, 2013.

[45] M. Xiao, P. Breitkopf, R. Filomeno Coelho, C. Knopf-Lenoir, M. Sidorkiewicz, and P. Villon, "Model reduction by cpod and kriging: application to the shape optimization of an intake port," *Structural and multidisciplinary optimization*, vol. 41, pp. 555–574, 2010.

[46] K. Hoang, Y. Fu, and J. Song, "An hp-proper orthogonal decomposition–moving least squares approach for molecular dynamics simulation," *Computer Methods in Applied Mechanics and Engineering*, vol. 298, pp. 548–575, 2016.

[47] R. Everson and L. Sirovich, "Karhunen–loeve procedure for gappy data," *JOSA A*, vol. 12, no. 8, pp. 1657–1664, 1995.

[48] P. Astrid, S. Weiland, K. Willcox, and T. Backx, "Missing point estimation in models described by proper orthogonal decomposition," *IEEE Transactions on Automatic Control*, vol. 53, no. 10, pp. 2237–2251, 2008.

[49] D. Amsallem and C. Farhat, "An online method for interpolating linear parametric reduced-order models," *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2169–2198, 2011.

[50] J. S. Hesthaven, G. Rozza, and B. Stamm, "Certified reduced basis methods for parametrized partial differential equations," *Springer*, 2015.

[51] D. Huang, J. N. Fuhg, C. Weißenfels, and P. Wriggers, "A machine learning based plasticity model using proper orthogonal decomposition," *Computer Methods in Applied Mechanics and Engineering*, vol. 365, p. 113008, 2020.

[52] C. Ooi, Q. T. Le, M. H. Dao, V. B. Nguyen, H. H. Nguyen, and T. Ba, "Modeling transient fluid simulations with proper orthogonal decomposition and machine learning," *International Journal for Numerical Methods in Fluids*, vol. 93, no. 2, pp. 396–410, 2021.

[53] S. Chaturantabut and D. C. Sorensen, "Nonlinear model reduction via discrete empirical interpolation," *SIAM Journal on Scientific Computing*, vol. 32, no. 5, pp. 2737–2764, 2010.

[54] A. T. Mohan and D. V. Gaitonde, "A deep learning based approach to reduced order modeling for turbulent flow control using lstm neural networks," *arXiv preprint arXiv:1804.09269*, 2018.

[55] J. S. Hesthaven and S. Ubbiali, "Non-intrusive reduced order modeling of nonlinear problems using neural networks," *Journal of Computational Physics*, vol. 363, pp. 55–78, 2018.

[56] N. J. Falkiewicz and C. E. Cesnik, "Proper orthogonal decomposition for reduced-order thermal solution in hypersonic aerothermoelastic simulations," *AIAA journal*, vol. 49, no. 5, pp. 994–1009, 2011.

[57] C. A. Rogers, A. J. Kassab, E. A. Divo, Z. Ostrowski, and R. A. Bialecki, "An inverse pod-rbf network approach to parameter estimation in mechanics," *Inverse Problems in Science and Engineering*, vol. 20, no. 5, pp. 749–767, 2012.

[58] V. Huayamave, A. Ceballos, C. Barriento, H. Seigneur, S. Barkaszi, E. Divo, and A. Kassab, "Rbf-trained pod-accelerated cfd analysis of wind loads on pv systems," *International Journal of Numerical Methods for Heat & Fluid Flow*, vol. 27, no. 3, pp. 660–673, 2017.

[59] W. Lee, K. Jang, W. Han, and K. Y. Huh, "Model order reduction by proper orthogonal decomposition for a 500 mwe tangentially fired pulverized coal boiler," *Case Studies in Thermal Engineering*, vol. 28, p. 101414, 2021.

[60] Z. Ostrowski, R. Białecki, and A. J. Kassab, "Solving inverse heat conduction problems using trained pod-rbf network inverse method," *Inverse Problems in Science and Engineering*, vol. 16, no. 1, pp. 39–54, 2008.

[61] H. S. R. Rajula, G. Verlato, M. Manchia, N. Antonucci, and V. Fanos, "Comparison of conventional statistical methods with machine learning in medicine: diagnosis, drug development, and treatment," *Medicina*, vol. 56, no. 9, p. 455, 2020.

[62] A. Chatterjee, "An introduction to the proper orthogonal decomposition," *Current science*, pp. 808–817, 2000.

[63] S. Ferrari and R. F. Stengel, "Smooth function approximation using neural networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 24–38, 2005.

[64] S. Sapna, A. Tamilarasi, M. P. Kumar, *et al.*, "Backpropagation learning algorithm based on levenberg marquardt algorithm," *Comp Sci Inform Technol (CS and IT)*, vol. 2, pp. 393–398, 2012.

[65] F. A. Potra, "Interior-point methods florian a. potra and stephen j. wright," 2000.

[66] A. Ranganathan, "The levenberg-marquardt algorithm," *Tutoral on LM algorithm*, vol. 11, no. 1, pp. 101–110, 2004.

[67] G. Tortora, A. Concilio, and R. Pecora, "Airfoil shape morphing through a novel parameterization and fitting optimization method based on uniform non-rational b-spline functions," *Designs*, vol. 7, no. 1, 2023. [Online]. Available: https://www.mdpi.com/2411-9660/7/1/28

[68] W. Ma and J.-P. Kruth, "Parameterization of randomly measured points for least squares fitting of b-spline curves and surfaces," *Computer-Aided Design*, vol. 27, no. 9, pp. 663–675, 1995.

[69] A. Shikhar Jaiswal, "Shape parameterization of airfoil shapes using bezier curves," in *Innovative Design and Development Practices in Aerospace and Automotive Engineering: I-DAD, February 22-24, 2016*. Springer, 2016, pp. 79–85.

[70] C. L. Ladson and C. W. Brooks Jr, "Development of a computer program to obtain ordinates for naca 4-digit, 4-digit modified, 5-digit, and 16 series airfoils," Tech. Rep., 1975.

[71] A. Muftah, "Cfd simulation and optimization of 4-digit naca airfoils."

[72] J. J. Moré, "The levenberg-marquardt algorithm: implementation and theory," in *Numerical Analysis: Proceedings of the Biennial Conference Held at Dundee, June 28–July 1, 1977*. Springer, 2006, pp. 105–116.

[73] P. E. Gill, W. Murray, and M. A. Saunders, "Snopt: An sqp algorithm for large-scale constrained optimization," *SIAM review*, vol. 47, no. 1, pp. 99–131, 2005.