LARGE EDDY SIMULATIONS OF A BACK-STEP TURBULENT FLOW AND PRELIMINARY ASSESSMENT OF MACHINE LEARNING FOR REDUCED ORDER TURBULENCE MODEL DEVELOPMENT

by

Biswaranjan Pati

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science

School of Aeronautics and Astronautics
West Lafayette, Indiana
August 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Haifeng Wang, Chair

School of Aeronautics and Astronautics

Dr. Jun Chen

School of Mechanical Engineering

Dr. Timothée Pourpoint

School of Aeronautics and Astronautics

Approved by:

Dr. Gregory Blaisdell

I dedicate this thesis to my parents, Dr. and Mrs. Prasad Kumar Pati, whose unrelenting efforts are a reflection of my own life.

ACKNOWLEDGMENTS

My sincerest appreciation goes to all the people without whose support the completion of this thesis would not have been possible.

Firstly I would like to thank my parents whose unconditional love and support over the years prepared me to overcome all the challenges in my personal and academic life. My younger brother holds a special place in my heart, and I hope to become a role model to him as my parents were to me. In addition to my parents, I would also like to thank my late grandmother, Mrs. Subhasini Rath for being a constant source of inspiration my whole life.

My heartfelt thanks goes to my supervisor Dr. Haifeng Wang for accepting me into CEPL (Computational Energy and Propulsion Lab) and affording me the opportunity to work with him for the past two years. I'm deeply grateful to my thesis committee members Dr. Jun Chen and Dr. Timothée Pourpoint for agreeing to be a part of the thesis defense. I can say the same for all my research associates at the CEPL: Tianfang Xie, Jie Tao, Utsav Jain, Abhinand Ayyaswamy, Krutika Appaswamy, and Xinran Zhao. Their experience and constant support have been a key factor in the completion of this thesis.

Lastly a special thanks goes to Purdue University for their support in matters both computational and financial, which gave me an impetus to work hard towards my goals. All the computational simulations would not have been possible without the valuable contribution of the Research Computing Center (Super computing clusters) and Engineering Computer Network (ECN) at Purdue University.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

a_{ij}	anisotropic Reynolds stresses
u	velocity
k	turbulent kinetic energy
$R\mathrm{e}$	Reynolds number
U	large scale velocity
L	large scale length
T	large scale time
u_η	Kolmogorov's velocity scale
p	pressure
R_λ	Taylor microscale Reynolds number
$\mathbb{R}$	set of Real numbers
$\mathbb{E}$	expectation
g	gravitational acceleration
$T_{k\mathrm{ij}}$	Reynolds stress-flux
$\mathcal{P}_{\mathrm{ij}}$	production tensor
$\mathcal{R}_{\mathrm{ij}}$	pressure strain-rate tensor
S_{ij}	strain rate deformation
C_s	Smagorinsky model constant
Δ_g	grid spacing
X_r	re-circulation length
C_p	coefficient of pressure
C_f	coefficient of skin friction
N_x	number of grid points in x-direction
N_y	number of grid points in y-direction
N_z	number of grid points in z-direction
N_{xyz}	total number of grid points in the mesh
$\Delta y_{min}/h$	minimum grid spacing normalized by channel height

Gr*ee*kAlphab*e*ts

ν	kinematic viscosity
δ_{ij}	kronecker delta
ϵ	rate of dissipation
η	Kolmogorov's length scale
τ_η	Kolmogorov's time scale
ρ	density
ϵ_{ij}	dissipation tensor
τ_{ij}	residual stress tensor (SGS stress tensor)
μ_t	eddy viscosity

ABBREVIATIONS

AI	Artificial Intelligence
APG	Adverse Pressure Gradient
CDF	Cumulative Density Function
CFD	Computational Fluid Dynamics
CV	Cross-Validation
DNS	Direct Numerical Simulations
ER	Expansion Ratio
FLOPS	Floating Point Operations
GCI	Grid Convergence Index
JPDF	Joint Probability Density Function
LES	Large Eddy Simulations
LEVM	Linear Eddy Viscosity Model
ML	Machine Learning
NS	Navier-Stokes
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PDF	Probability Density Function
RANS	Reynolds-Averaged Navier Stokes
RF	Random Forest
SGS	Subgrid-Scale
WALE	Wall-Adapting Local Eddy-Viscosity

ABSTRACT

Accuracy in turbulence modeling remains a hurdle in the widespread use of *Computational Fluid Dynamics* (CFD) as a tool for furthering fluids dynamics research. Meanwhile, computational power remains a significant concern for solving real-life wall-bounded flows, which portray a wide range of length and time scales. The tools for turbulence analysis at our disposal, in the decreasing order of their accuracy, include *Direct Numerical Simulation* (DNS), *Large Eddy Simulation* (LES), and *Reynolds-Averaged Navier Stokes* (RANS) based models. While DNS and LES would remain exorbitantly expensive options for simulating high Reynolds number flows for the foreseeable future, RANS is and continues to be a viable option utilized in commercial and academic endeavors. In the first part of the present work, flow over the back-step test case was solved, and parametric studies for various parameters such as re-circulation length (X_r), coefficient of pressure (C_p), and coefficient of skin friction (C_f) are presented and validated with experimental results. The back-step setup was chosen as the test case as turbulent modeling of flow past backward-facing step has been pivotal to understand separated flows better. Turbulence modeling is done on the test case using RANS (k-ϵ and k-ω models), and LES modeling, for different values of Reynolds number ($Re \in \{2, 2.5, 3, 3.5\} \times 10^4$) and expansion ratios ($ER \in \{1.5, 2, 2.5, 3\}$). The LES results show good agreement with experimental results, and the discrepancy between the RANS results and experimental data was highlighted. The results obtained in the first part reveal a pattern of under-prediction noticed with using RANS-based models to analyze canonical setups such as the backward-facing step. The LES results show close proximity to experimental data, as mentioned above, which makes it an excellent source of training data for the machine learning analysis outlined in the second part. The highlighted discrepancy and the inability of the RANS model to accurately predict significant flow properties create the need for a better model. The purpose of the second part of the present study is to make systematic efforts to minimize the error between flow properties from RANS modeling and experimental data, as seen in the first part. A machine learning model was constructed in the second part of the present study to predict the eddy viscosity parameter (μ_t) as a function of turbulent kinetic energy (TKE) and dissipation rate (ϵ) derived from LES data, effectively working

as an ad hoc eddy-viscosity based turbulence model. The machine learning model does not work well with the flow domain as a whole, but a zonal analysis reveals a better prediction of eddy viscosity than the whole domain. Among the zones, the area in the vicinity of the re-circulation zone gives the best result. The obtained results point towards the need for a zonal analysis for the better performance of the machine learning model, which will enable us to improve RANS predictions by developing a reduced order turbulence model.

1. INTRODUCTION

In the modern world, where computers dictate every aspect of our lives, the importance of computational work and data-driven physics are more relevant than ever. Computer simulations have replaced the proverbial *"Engineering handiwork"*, which shapes everyday life. High-fidelity simulations have played an essential role in understanding complex physical processes like turbulence and, with time, have proved to be an excellent tool for compounding theories for explaining such phenomena. This chapter presents an overview of turbulence and how to model it, emphasizing on data-driven turbulence modeling.

1.1 Overview of Data-Driven Turbulence Modeling

The discussion shown below gives a brief literature review of the past and emerging ideas in the field of data-driven turbulence modeling. The year 2015 was used as a natural breaking point between the sections as somewhere around that time, the existing norms changed, and new ideas began emerging; therefore, a separate section has been presented to highlight the more recent studies in this field.

1.1.1 Previous Work on Data-Driven Turbulence Modeling (Prior 2015)

The existence of a multitude of turbulence models is an indicator that no single model can explain the characteristics of all flow systems satisfactorily. Whether or not we can develop such a *"Universal Model"* is a discussion for another day. The current tools for turbulence modeling at our disposal, in the decreasing order of their accuracy, include *Direct Numerical Simulation* (DNS), *Large Eddy Simulation* (LES), and *Reynolds-Averaged Navier Stokes* (RANS) based models. A growing belief in the high-power computing community that LES was the next big thing, as the low pass filtering in the LES model is more accurate than RANS averaging. It is supposed to be a viable trade-off between DNS and RANS, presenting a computationally judicious option without compromising on accuracy.

It is not a generally believed notion that high-fidelity models are a better alternative to the eddy-viscosity models that have been in use for so long. They have not led to any noticeable improvements despite the theoretical advantages they offer over RANS modeling due to its empirical nature [1]. Over the years, many changes have been proposed to Eddy-viscosity models to increase their credibility. These are done to make the models more responsive towards the transition from laminar to turbulent flows [2], anisotropy due to near wall effects [3], and simple Galilean effects such as rotation and curvature [4] [5] [6]. Despite its many limitations RANS model continues to be the industry workhorse after many decades of its conception.

Meanwhile, data-driven physics has gained much popularity in the last decade and a half. Data science has meandered its way into every facet of human existence: image processing [7], advanced linguistics [8], and speech recognition [9]. The growth in the data science sector is unprecedented in terms of both number and application. The massive surge is indicative of a need for better classification and processing of the vast amount of data collected on the world wide web every day. The dissemination of ideas between the physical sciences and data science community has been a slow albeit fascinating problem-solving method. The gradual nature of the development can be primarily attributed to the fact that data science concepts have to be modified to respect the laws of physics.

Although high fidelity data exists for simple geometric turbulent flows at low Reynolds number, it has seldom been used to improve RANS models summarily. The high volume of available data and the massive growth in the data science sector has presented researchers with an excellent opportunity to improve turbulence models in an organized manner. The following section presents a brief literature review of the previous work done in the field of data-driven turbulence modeling. The process of using data-driven techniques to improve turbulence modeling has been underway for the past few decades now, with better strategies emerging over time. A case can be made that turbulence models have always headed towards data calibration due to their inherently empirical nature, as the theoretical approach cannot fill some big holes. For instance, some significant examples attest to this fact, such as predicting the coefficient of skin friction for zero pressure gradient for flow over a flat plate or predicting the RANS model constants by data calibration to better fit equations.

In 1998, Paraneix et al. [10] developed a comprehensive study using DNS datasets to target the improvement of second-moment closure equations. They conducted the a priori testing using DNS data to solve the transport equations for one isolated component of the Reynolds stress tensor. The posteriori testing involves modifying the model equations to fit the variable values obtained from their DNS dataset. Raiesi et al. [11] attempted to improve the accuracy of one and two-equation models using LES and DNS results for turbulent kinetic energy and dissipation rate.

In the early 2000s, Neural Networks (NNs) were fast becoming the primary tool for predicting the near-wall behavior in channel flows. Unlike unbounded and isotropic flows, many researchers run into a precarious situation when predicting the near-wall behavior in channel flows. This is due to the formation of boundary layer and the mainstream turbulent models failing to capture the wall effects. In 2002, Milano and Koumoutsakos [12] used Neural Networks to understand the near-wall effects in a better way. They recreated their wall model with second-order discretization by using wall quantities to express the higher-order terms. Several studies came forward between 2011 to 2014 [13] [14] [15] [16] using Probability Density Function (PDF) and Joint PDF of model parameters to reduce inaccuracies in models. Edeling et al. [14] quantified the statistical error in quantities such as velocity and coefficient of skin friction by gathering data from numerous experiments on boundary layer development. The above discussion points to the deficiency of a priori testing and the absence of model form discrepancies.

Even though data-driven modeling was heavily integrated into the traditional turbulence analysis paradigm by the 2010s, research about systematically improving eddy-viscosity models was still scarce. Dow and Wang [17] [18] developed an NN to understand eddy viscosity's structural uncertainties better using the velocity and pressure data from direct numerical simulations. This served as a motivation for the current research, an effort to use high-fidelity results obtained from LES modeling to improve results obtained from RANS-based analysis.

1.1.2 New Ideas Emerging Post-2015

Machine learning algorithms in improving turbulence analysis can be considered a recent endeavor as it all but started two decades ago. However, somewhere around 2015, the existing norms changed, and new ideas began emerging; therefore, a separate section has been presented to highlight the more recent studies in this field.

Xiao et al. [19] [20] used DNS data to calculate the spatial arrangement of perturbations in aij. The Anisotropic tensor, a_{ij}, is related to the velocity fluctuations and turbulent kinetic energy in the following way [21]:

$$a_{\mathrm{ij}} = \langle u_{\mathrm{i}}' u_{\mathrm{j}}' \rangle - \frac{2}{3} k \delta_{\mathrm{ij}}, \tag{1.1}$$

where $\langle u_{\mathrm{i}}' u_{\mathrm{j}}' \rangle$ denote the Reynolds stress, k denotes the turbulent kinetic energy, and δ_{ij} denotes the kronecker delta.

The stresses resulting from the induced perturbations are calculated by transforming the eigenvalues of the a_{ij} into Barycentric coordinates. They remodeled the perturbations in the Cartesian coordinate system using machine learning. This machine learning model was combined with the RANS model to predict improved results for a different flow setup. Weatheritt [22] from the University of Southampton had an interesting take on algebraic modeling using machine learning for his graduate dissertation. He used DNS data to construct an algebraic stress-strain expression for RANS equations using an evolutionary machine learning algorithm to understand the anisotropic tensor better.

A natural precursor to the current work was presented by Matai et al. [23] in 2018 where he used LES modeling to simulate flow over a set of parametric bumps and used Artificial Neural Network (ANN) to predict the drag over the bumps. The current work is based on a similar line, to construct a machine learning (ML) model using Random Forest algorithm to predict the eddy viscosity, μ_t, in flow over a back-step. The ML model is intended to minimize the error observed between flow properties from RANS modeling and experimental data. The experimental results were used to validate the LES modeling making it an excellent source of training data for the machine learning analysis outlined in the following sections. A further zonal analysis is performed to isolate the specific zones of the flow domain where the model performs the best, which will enable us to improve RANS predictions by developing a reduced order turbulence model.

1.2 Overview of Turbulence in Fluid Flow

The discussion shown below gives a brief literature review of the past and present norms in turbulence modeling. The section contains a clear picture of the problem statement that is addressed further in this study.

1.2.1 Characteristics of Turbulence

Turbulence is the common trait for all chaotic and seemingly random phenomena occurring in nature. Although defining turbulence explicitly is a challenging endeavor, but most of the literature found on this subject use the following metrics to characterize turbulence [21] [24]:

- Randomness in flow characteristics
- Increasingly diffusive flow
- Increasingly dissipative flow
- 3-D fluctuations

As turbulence is a characteristic trait of the flow and not the fluid, various flow parameters must be taken into account to quantify turbulence accurately. One such dimensionless quantity, Reynolds number, Re, is a quantifiable measure of the state of the flow: laminar, turbulent, or a transitional state between the two. Re can be defined as the ratio of inertial forces to the viscous forces and is written as:

$$R\text{e} = \frac{UL}{\nu}, \tag{1.2}$$

where U is the characteristic velocity, L is the characteristic length, and ν is the kinematic viscosity. A flow with high Re is often characterized by a high degree of turbulence which indicates energy is injected into smaller scales of motion through cascading. In highly turbulent flows, the largest eddies can be orders of magnitude more prominent than the smallest ones. As the turbulence increases, the range of scales of motion also tends to increase, and hence solving the set equations characterizing the flow for all the scales becomes a gargantuan task. As most flows of interest in the aerospace community are high Re flows, therefore containing a high degree of turbulence, it is essential to understand the turbulent scales of motion.

1.2.2 Scales of Turbulent Motion

In 1941, A.N. Kolmogorov [25] put forward a comprehensive analysis about the smallest scales in a turbulent flow. In this article, he compounded the Kolmogorov Hypotheses, which state, "*At sufficiently high Reynolds number, the small-scale turbulent motions are statistically isotropic. In every turbulent flow at sufficiently high Reynolds number, the statistics of the small-scale motions have a universal form that is uniquely determined by the kinematic viscosity, ν, and the specific dissipation, ϵ.*" Given the two parameters, ν, and ϵ, Kolmogorov formed unique length, velocity, and time scales collectively known as the Kolmogorov's microscales.

$$\eta \equiv \left(\frac{\nu^3}{\epsilon}\right)^{\frac{1}{4}}, \quad u_\eta \equiv (\epsilon\nu)^{\frac{1}{4}}, \quad \tau_\eta \equiv \left(\frac{\nu}{\epsilon}\right)^{\frac{1}{2}}, \tag{1.3}$$

where η, μ_η, and τ_η are Kolmogorov's length, velocity and time scale respectively.

Kolmogorov's hypotheses proved to be the first comprehensive work on linking the small and large scale properties. As mentioned before, the energy cascades from the larger scales into the smaller scales of flow by the continual breaking of larger eddies into smaller ones. For simplicity of analysis, we assume that during cascading, there is energy transfer without any loss. Using the scaling for specific dissipation, $\epsilon \sim U^3/L$, we can obtain the following relations:

$$\eta/L \sim Re^{-\frac{3}{4}}, \quad u_\eta/U \sim Re^{-\frac{1}{4}}, \quad \tau_\eta/T \sim Re^{-\frac{1}{2}}, \tag{1.4}$$

where L is the large scale length, U is the large scale velocity, and T is the large scale time. The ratio of the smallest and largest scale quantities can be expressed as a function of Reynolds number.

1.2.3 Turbulence Modeling

The previous section highlighted the importance and ubiquity of turbulence in hydrodynamic flows. Therefore it is equally important to model turbulence experimentally and analytically to get a real sense of everyday phenomena. Analytical turbulence modeling involves solving mathematical equations in their discretized forms. Physical modeling involves complex mathematical equations to explain the nuances of a system, and more often than not, they are derived from first principles. For example, the Navier-Stokes (NS) equations, which are the governing equations of fluid dynamics, are a derivative of the conservation of mass, momentum, energy. Empirical models are on the other extreme, which involves fitting the model to a dataset. Most models currently in use are an amalgamation of both extremes, involving first principles and empirical modeling.

The key to the accuracy of simulations for explaining complex physical phenomenon lies in solving mathematical equations in discretized form. So, for a simulation to be accurate, the underlying mathematical model has to be equally accurate and the discretization errors, minimum. To sum it up, obtaining accurate results depends on using proper methods to solve the right set of equations. A quality solution to a set of discretized equations is a

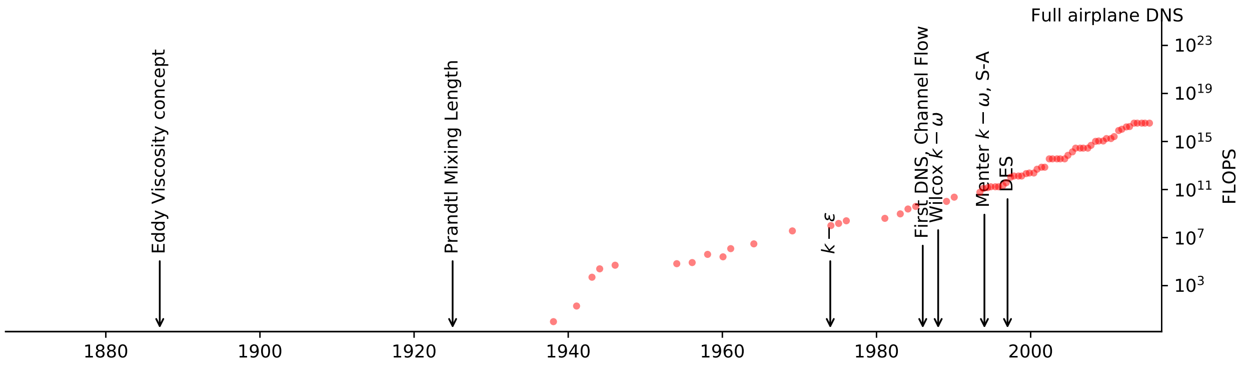


Figure 1.1. An approximate timeline of the graduation in turbulence simulations. The figure was taken from Singh [26].

careful balance between the computational time and accuracy - especially in turbulent flows, where resolving the smallest scales can be of utmost importance. A complicated flow where all scales of the flow are fully resolved is still a pipe dream as the current computational standards are insufficient to achieve such a result. Even when realizable, it will require months of computation time on the most powerful systems on earth, severely restricting its viability as a repeatable process.

In most applications of *Computational Fluid Dynamics* (CFD), the set of NS, continuity, and energy equations are sufficient to analyze all flow properties. The NS equations can take myriad forms, and the non-linearities present in the PDEs explain the flow complexities occurring between the various scales. The incompressible, unsteady form of NS equations can be expressed in the following way using Einstein's index notation:

$$\frac{\partial u_\mathrm{i}}{\partial x_\mathrm{i}} = 0, \tag{1.5}$$

$$\frac{\partial u_\mathrm{i}}{\partial t} + \frac{\partial u_\mathrm{i} u_\mathrm{j}}{\partial x_\mathrm{j}} = -\frac{1}{\rho}\frac{\partial p}{\partial x_\mathrm{i}} + \nu \frac{\partial^2 u_\mathrm{i}}{\partial x_\mathrm{i} \partial x_\mathrm{j}}, \tag{1.6}$$

where u_i is the velocity in the i^{th} direction, p is the pressure and ρ is the density.

Turbulence modeling involves discretizing and solving the NS equations for all flow scales to get an accurate solution of flow properties. If the discretization schemes allow us to solve the NS equation for all flow scales, i.e., the flow is *fully resolved*, then the solution can be deemed equally accurate to the analytical solution[1]. Such flow solutions are known as DNS [27] [28] and the high computational costs associated with it disqualify DNS as a practical candidate for running complex high Reynolds number simulations.

Figure 1.1 shows a plot for various breakthroughs in turbulent simulations with the increasing sophistication of computing units using Floating Point Operations (FLOPS) as a metric. The DNS of a full airplane is at the extreme end in terms of processing power, and that milestone is still a few decades away. Even with a system powerful enough to resolve all scales [29] [30], we are looking at years of run-time. Dealing with the considerable run-time poses a significant problem, especially when coupled with the viability of the whole simulation. Therefore, we depend on alternate turbulence models, which are a trade-off on resolving scales but accurate enough to be viable in terms of run-time.

As we have discussed the positives and negatives of using DNS in turbulence analysis, the current work focuses on the LES model to obtain high fidelity data without compromising numerical accuracy. Flow over the back-step test case was solved, and parametric studies for various parameters such as re-circulation length (X_r), coefficient of pressure (C_p), and coefficient of skin friction (C_f) are presented and validated with experimental results. The back-step setup was chosen as the test case as turbulent modeling of flow past backward-facing step has been pivotal to understand separated flows better. The validation of LES results with experimental data establishes a good agreement between the two and highlights the discrepancy between the RANS results and the experimental results. Thus, the LES data can be used to derive the training variables to feed into the ML model during the data-driven part of the analysis. The present study is intended to identify the critical need for a new turbulence model as the existing RANS models show limited closure. The work

[1] ↑This solution does not include the numerical errors due to finite precision of solvers.

will enable us to construct a reduced order turbulence model to which new features can be easily added, showing better predictions than RANS analysis.

1.3 Major Research Contributions

The major research contributions derived from the present work are listed below:

- The primary goal of this thesis is to analyze the turbulence for flow over a back-step test case and use a machine learning framework to improve upon the RANS analysis pre-existing in this arena.
- Problems related to canonical flow domains such as back-step are addressed, and sustained efforts are made to improve the analysis.
- A thorough comparison of RANS, LES, and experimental data is conducted, highlighting the limitations RANS-based models pose to solving canonical flow cases such as back-step.
- An ad hoc machine learning model is constructed using the LES model's training data to improve RANS results systematically. This represents a new framework for constructing a reduced order turbulence model and showing improved predictions over RANS analysis. This will enable our future work on adding and fine-tuning parameters to the preliminary model to further improve upon results.
- Based on the turbulence and machine learning analysis, significant conclusions are drawn about the positives and negatives of using machine learning in turbulence analysis.

1.4 Thesis Outline

The current thesis is divided into a total of six chapters. Chapter 1 is focused on presenting a brief introduction to the turbulence and machine learning analysis shown in the following chapters. In addition, a literature review showcasing the work of previous researchers in both fields has been presented.

Chapter 2 delineates the detailed machine learning approach used in the present study. Various facets of a machine learning model, including feature selection, normalization, cross-validation, and quality determination, have been explained in the chapter. In addition, a detailed account of the basic principles involved *Random Forest* method is also shown in this chapter.

Chapter 3 discusses the details of the computational methodology used for simulating the flow over a back-step and extracting usable training data from the LES model. Additionally, various details such as computational domain, grids, and turbulence modeling approach are also discussed in this chapter.

After detailed discussions regarding the flow parameters in the previous chapters, Chapter 4 shows multiple parametric studies conducted to analyze the flow behavior with respect to Reynolds number, expansion ratios, and wall functions in turbulence models. Other than that, a detailed grid convergence analysis is shown in this chapter to choose the optimum grid for accurate results.

Chapter 5 talks about the construction and application of the machine learning model in the present work. This chapter points out the limitations of RANS modeling in canonical cases like the flow over a back-step. The mathematical methods for training data extraction from the LES modeling are also discussed here. A further zonal analysis is elaborated to discuss the performance of the machine learning model in different zones of the flow domain.

Based on the above chapters, significant conclusions are drawn, and the future scope of work is discussed in Chapter 6.

2. MACHINE LEARNING APPROACH FOR PHYSICAL MODELING

In this chapter, a detailed background and framework of the machine learning approach used in the present study are discussed. Section 2.1 gives a general overview of machine learning, and Sections 2.2 to 2.5 present the different factors contributing to forming a good ML model. Section 2.6 gives us some idea about *Random Forest* as tool for data analytics. Section 2.7 shows the usage of machine learning in physical modeling of real-life flow phenomena.

2.1 Machine Learning

Machine learning (ML), when simply put, is a mathematical tool used for data classification and efficient decision making. The meteoric rise in the data analytics sector is indicative of a need for innovative methods to manage the vast amount of data pouring into the world wide web every day. The amount of data channeling through the internet is staggering; for example, in a Digital Universe Study on Big Data sponsored by a corporation called the EMC2 [31], it was predicted that by the year 2020, every human on earth would generate 1.7 MB of data every second.

Three significant sub-classes of machine learning problems as shown in Figure 2.1 are supervised, unsupervised, and reinforcement learning. When we consider a pair of input-output variables, $N = [\mathbf{x_i}, \mathbf{f_i}]_{\mathrm{i}=1}^{N}$ the three categories vary in terms of the method used to map a correlation between them. Supervised learning is a method to plot a relationship between the inputs, $\mathbf{x_i}$, and the outputs, $\mathbf{f_i}$. the process of developing this mapping is known as **training**, and using the resulting model to predict output for a different set of id inputs is known as **testing**. Usually, the input variable is a vector, known as a feature set, and the output variable is a scalar. **Classification** a type of supervised learning problem where the output variable takes its values from a discreet set. In 2016, Ling et al. [32] published a study where they solved a classification problem by establishing a region of uncertainty in RANS simulation by using DNS data. On the other hand, when the output variable can

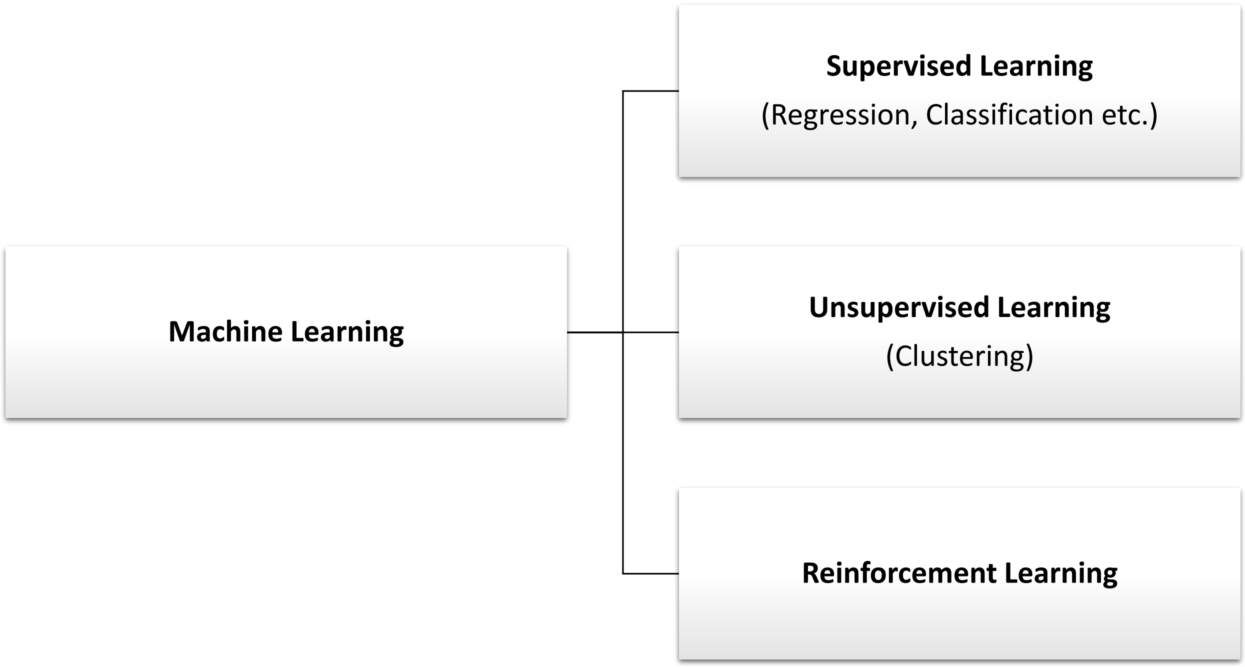


Figure 2.1. Types of sub-classes of Machine learning models: Supervised, Unsupervised, and Reinforcement learning.

take any real number as its value, the problem is referred to as **Regression**.

Unsupervised learning differs from supervised learning in the way that there is no fixed output variable to construct a mapping, and the algorithm solely utilizes the input data, $N = [\mathbf{x_i}]_{\mathrm{i=1}}^{N}$, to form patterns. Unsupervised learning is a beneficial tool to analyze gene clustering, social media analysis of a region or demographic, study market segmentation, and astronomical data analysis. Reinforcement learning is another sub-class of machine learning which deals with cumulative reward functions to simulate intelligent decision making. It is primarily used in Artificial Intelligence (AI) programs to mimic how intelligent beings make decisions in a particular environment.

Figure 2.2 demonstrates a simple flow chart to understand the working of supervised learning method. The present work deals mainly with supervised learning, using Random Forest to predict turbulent viscosity in fluid flows using k and ϵ from LES data. The following sections focus on the different aspects of the machine learning approach in the current work.

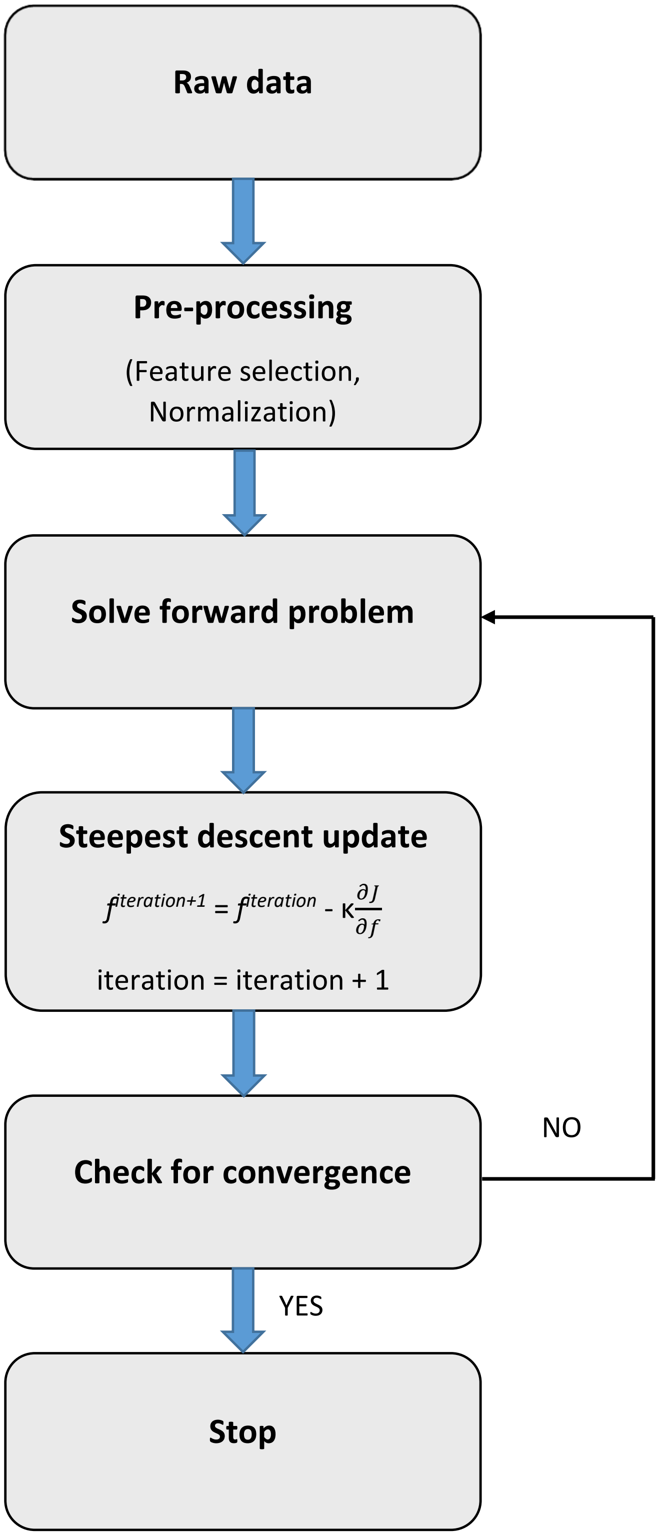


Figure 2.2. A simple flow-chart to demonstrate supervised machine learning.

2.2 Selection of Features

The input feature set, $\mathbf{x_i}$, in an ML model is an essential characteristic. It should represent the primary attributes of the raw data; the better the representation, the better the model works. Irrelevant or redundant features can negatively impact the performance of an ML model. Some advantages of good feature selection techniques are listed below:

- Good feature selection leads to reduction in over-fitting.
- There is overall minimization of misleading data, which increases the accuracy of solution.
- Good feature set reduces the complexity of training algorithm and the solution is obtained faster.

Desirable traits while selecting a feature set are univariate selection and the non-dimensionality of the features. Univariate selection refers to statistical tests to determine which feature has the most substantial relationship with the output variable. Combined with non-dimensionality, they help create a robust ML model capable of operating in a general setting. Standard practice is to use algorithms [33] to determine the importance of feature when considering a large feature set. But, it is equally important to use domain knowledge for feature selection.

2.3 Normalization

Normalization is a good practice when the input variable contains features of varying orders of magnitude, which is the case for the present work, as demonstrated in later sections. It also helps is faster and more accurate training of the ML model. The following formula is used for normalization:

$$x^{\mathrm{i}}_{normalized} = \frac{x^{\mathrm{i}} - \overline{x^{\mathrm{i}}}}{\sigma_{x^{\mathrm{i}}}}, \tag{2.1}$$

where $\overline{x^{\mathrm{i}}}$ is the mean, and $\sigma_{x^{\mathrm{i}}}$ is the standard deviation of the ith component of the feature vector $\mathbf{x}$. Normalization should also be used during testing of the ML model.

2.4 Cross-Validating

Most regression models in use today work well for dispersed data in a fluid domain, as in turbulent flows. However, highly flexible models tend to over-fit the raw data, which becomes an essential consideration for turbulence analysis. An ML model which works for a wide range of data is considered robust, but more often than not, it leads to inaccuracy in prediction due to over-fitting. While over-fitting may seem like a necessary evil, it can be prevented by following some simple steps. Cross-validation (CV) while training is one of the most effective methods used today [34].

The basic algorithm for CV consists of dividing the training data in to M folds and for each m$\in\{1,\ldots,M\}$, the training is done for all folds except the m^{th} and then tested on the m^{th}. The total error is determined by averaging the error within all folds and the final data predicted is also the averaged result of the prediction by all M folds. Figure 2.3 represents a simple schematic diagram to demonstrate the process of cross-validation (CV). The diagram shows the working of a 3-fold CV. As the number of folds increases, the accuracy of the CV increases as well, and in many cases, a clustering CV program is used to group training and testing folds. Cross-validation (CV) is also an effective strategy for selecting optimal features for training and testing.

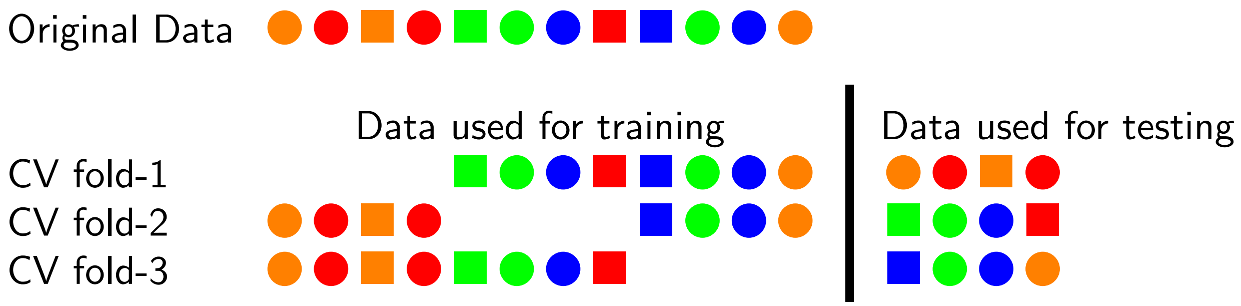


Figure 2.3. A figure to demonstrate the process of a three-fold cross-validation (CV). The figure is available at the url: https://tex.stackexchange.com/a/154121.

2.5 Quality of ML Model

Several factors can determine the quality of an ML model. The most common parameter to judge quality is called the coefficient of determination, or R^2. Consider a set of output data, $\{f_{1,true},\ f_{2,true},\ldots,f_{n,true}\}$, and the ML predicted output set, $\{f_{1,pred},\ f_{2,pred},\ldots,f_{n,pred}\}$, then R^2 is defined as,

$$R^2 \equiv 1 - \frac{SS_{res}}{SS_{tot}}, \tag{2.2}$$

where SS_{res} is the summation of the square of residuals, defined as,

$$SS_{res} = \sum_{\mathrm{i}} (f_{\mathrm{i},true} - f_{\mathrm{i},pred})^2. \tag{2.3}$$

SS_{tot} is a function of variance and is defined as,

$$SS_{tot} = \sum_{\mathrm{i}} (f_{\mathrm{i},true} - \overline{y})^2, \tag{2.4}$$

$$\overline{y} = \frac{1}{n} \sum_{\mathrm{i}} f_{\mathrm{i},true}. \tag{2.5}$$

Ideally, the value of $R^2 = 1$ implies a pristine ML model. But, any value of $R^2<1$ warrants human consideration to account for overall credibility of data and complexity of the ML model.

2.6 Random Forest (RF) Model

Random forest is a class of decision tree algorithm ideally suited for handling large data segments without compromising statistical efficiency. The random forest algorithm was devised originally by Breiman [33] in 2001, based on the earlier contributions of [35] [36] [37]. The basic principle of the algorithm is based on the following simple steps:

- Divide the large chunk of raw data into smaller sample sizes.

- Develop a randomized tree predictor for each sample size.

- Aggregate the tree predictors together.

The popularity of random forest is further fueled by its applicability and the need for tuning fewer parameters than other commercial regression software. The RF model has already been successfully integrated into various realistic scenarios, as evidenced by numerous studies including an EMC sponsored global data science hackathon on air-quality prediction[1]. To mention a few, Diaz-Uriarte and De Andres [38] were one of the first to introduce RF model into the field of bioinformatics for sample classification for their gene expression study. Prasad et al. [39] used RF for ecological prediction and Svetnik et al. [40] used it for data analysis in chemical engineering. On a theoretical note, the RF model is still a bit obscure as very little analysis is available on the mathematical formulation on the back-end. Figure 2.4 represents a simplified schematic of the network diagram of a Random Forest (RF) model.

[1] ↑https://www.kaggle.com/c/dsg-hackathon

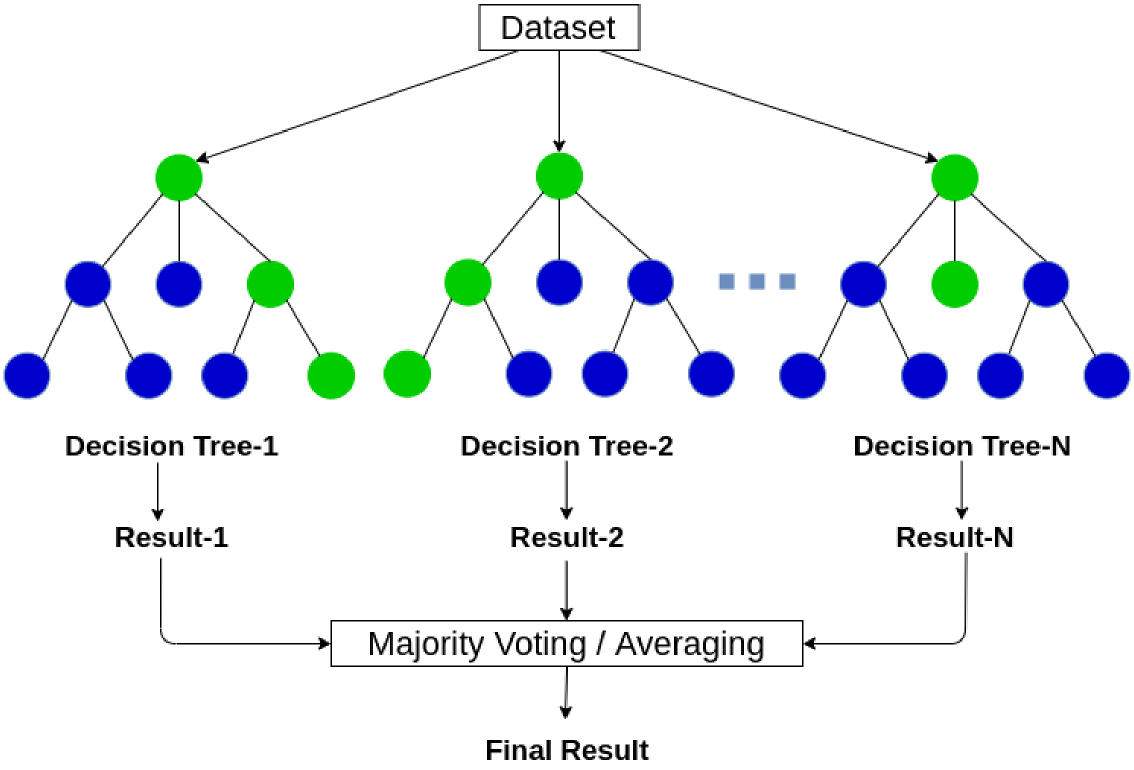
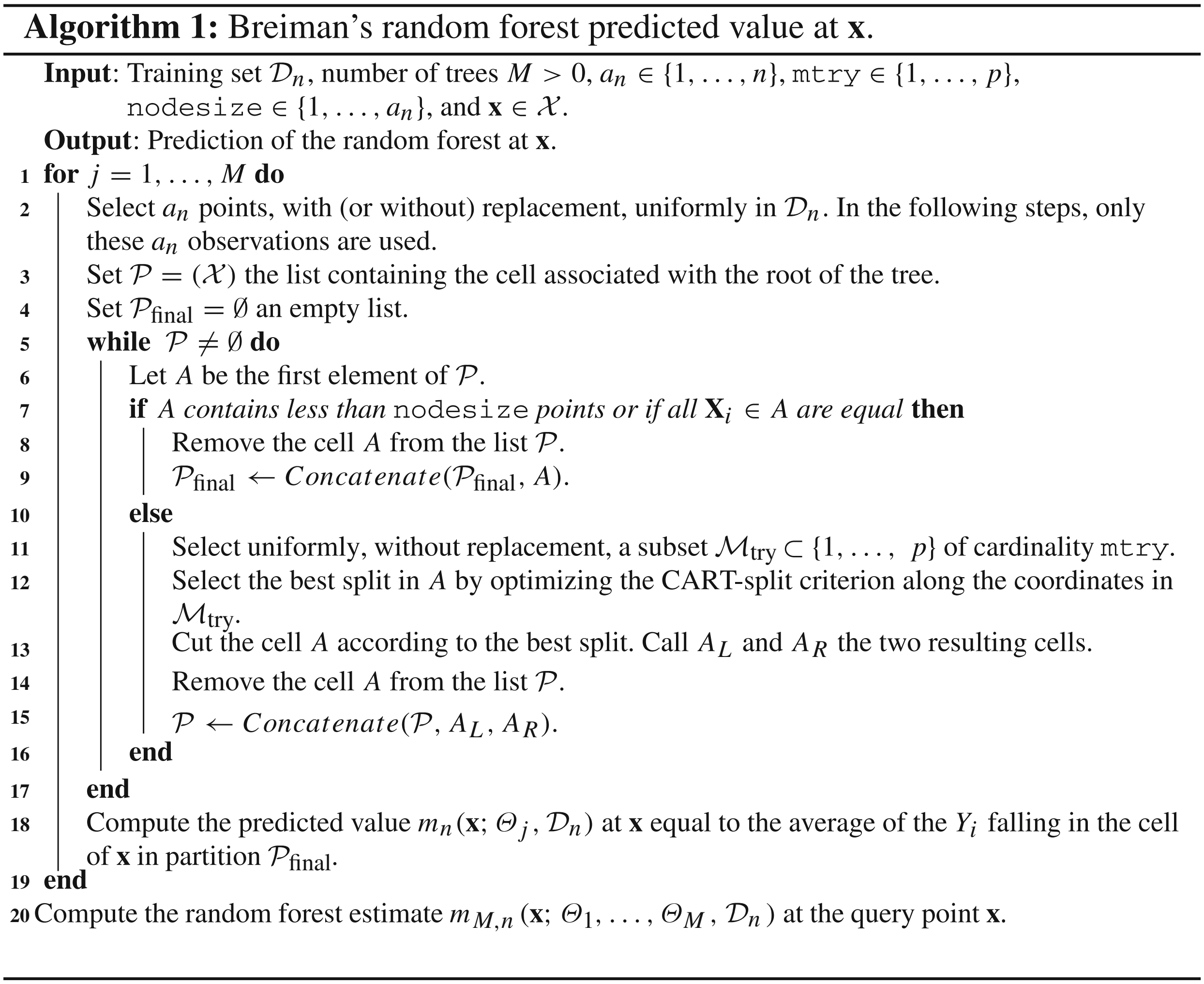


Figure 2.4. A figure depicting the network diagram for a simplified Random Forest model. The figure is adapted from this source.

The term *Random Forest* is a bit ambivalent as for some, it might mean both clustering decision trees, and for others, it might refer to Breiman's [33] original algorithm. In this section, we look at the mathematical definition of the Random Forest (RF) model put forth by Breiman in 2001 [41]. Consider the input vector $\mathbf{X} \in \mathcal{X} \subset \mathbb{R}^p$, and our goal is to predict the random output vector, $Y \in \mathbb{R}$ from the regression function $m(\mathbf{x}) = \mathbb{E}[Y \mid \mathbf{X} = \mathbf{x}]$. We have a training dataset $\mathcal{D}_n = ((\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n))$ which can be used to construct an estimate $m_n : \mathcal{X} \to \mathbb{R}$ such that $\mathbb{E}\left[m_n(\mathbf{X}) - m(\mathbf{X})\right]^2 \to 0$ as $n \to \infty$.



```
Algorithm 1: Breiman's random forest predicted value at $\mathbf{x}$.
Input: Training set $\mathcal{D}_n$, number of trees $M > 0$, $a_n \in \{1, \ldots, n\}$, mtry $\in \{1, \ldots, p\}$,
       nodesize $\in \{1, \ldots, a_n\}$, and $\mathbf{x} \in \mathcal{X}$.
Output: Prediction of the random forest at $\mathbf{x}$.
for $j = 1, \ldots, M$ do
    Select $a_n$ points, with (or without) replacement, uniformly in $\mathcal{D}_n$. In the following steps, only
    these $a_n$ observations are used.
    Set $\mathcal{P} = (\mathcal{X})$ the list containing the cell associated with the root of the tree.
    Set $\mathcal{P}_{\text{final}} = \emptyset$ an empty list.
    while $\mathcal{P} \neq \emptyset$ do
        Let $A$ be the first element of $\mathcal{P}$.
        if $A$ contains less than nodesize points or if all $\mathbf{X}_i \in A$ are equal then
            Remove the cell $A$ from the list $\mathcal{P}$.
            $\mathcal{P}_{\text{final}} \leftarrow Concatenate(\mathcal{P}_{\text{final}}, A)$.
        else
            Select uniformly, without replacement, a subset $\mathcal{M}_{\text{try}} \subset \{1, \ldots, p\}$ of cardinality mtry.
            Select the best split in $A$ by optimizing the CART-split criterion along the coordinates in
            $\mathcal{M}_{\text{try}}$.
            Cut the cell $A$ according to the best split. Call $A_L$ and $A_R$ the two resulting cells.
            Remove the cell $A$ from the list $\mathcal{P}$.
            $\mathcal{P} \leftarrow Concatenate(\mathcal{P}, A_L, A_R)$.
        end
    end
    Compute the predicted value $m_n(\mathbf{x}; \Theta_j, \mathcal{D}_n)$ at $\mathbf{x}$ equal to the average of the $Y_i$ falling in the cell
    of $\mathbf{x}$ in partition $\mathcal{P}_{\text{final}}$.
end
Compute the random forest estimate $m_{M,n}(\mathbf{x}; \Theta_1, \ldots, \Theta_M, \mathcal{D}_n)$ at the query point $\mathbf{x}$.
```

Figure 2.5. A figure to demonstrate Breiman's random forest algorithm devised in 2001 [41].

The random forest predictor comprises of $\mathcal{M}$ randomized trees where the jth tree takes the form,

$$m_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right)=\sum_{\mathrm{i}\in\mathcal{D}_n^*(\Theta_{\mathrm{j}})}\frac{\mathbf{1}\mathbf{x}_{\mathrm{i}}\in A_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right)Y_{\mathrm{i}}}{N_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right)}, \tag{2.6}$$

where $m_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right)$ is the predicted value at point $\mathbf{x}$, $\mathcal{D}_n^*$ is the set of points before the tree construction, $A_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right)$ is the cell containing $\mathbf{x}$, and $N_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right)$ are the points which fall into $A_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right)$.

The combinations of all trees take the (finite) form:

$$m_{M,n}\left(\mathbf{x};\Theta_1,\ldots,\Theta_M,\mathcal{D}_n\right)=\frac{1}{M}\sum_{\mathrm{j}=1}^{M}m_n\left(\mathbf{x};\Theta_{\mathrm{j}},\mathcal{D}_n\right). \tag{2.7}$$

When $\mathcal{M}\to\infty$ the combination takes the (infinite) form:

$$m_{\infty,n}\left(\mathbf{x};\mathcal{D}_n\right)=\mathbb{E}_\Theta\left[m_n\left(\mathbf{x};\Theta,\mathcal{D}_n\right)\right], \tag{2.8}$$

where $\mathbb{E}_\Theta$ is the calculated expectation with respect to Θ, conditional on $\mathcal{D}_n$.

Figure 2.5 represents the original algorithm proposed by Breiman in 2001 using R package to estimate RF predicted output for a regression problem. The three important parameters for the algorithm are described below:

- $a_n\in\{1,\ldots,n\}$: data points sampled per tree.
- `mtry` $\in\{1,\ldots,p\}$: permutations of all directions in which each node can be split at each tree.
- `nodesize` $\in\{1,\ldots,a_n\}$: number of samples in each cell below which the cell is not split.

2.7 Physical Modeling Using Machine Learning

Machine learning has been widely used for physical turbulence modeling, as is evident from the discussions presented above. Most hydrodynamic flows are physics-based, but choosing between a physics-based model and a data-driven model is problem-dependent. The problems can be classified into two categories:

- No direct analytical data is available on the system in question, but experimental data about its behavior exists.
- Mathematical description of the system is possible along with good theoretical understanding.

The problem investigated in the current work belongs to the second category. Combining data-driven modeling with system physics is an up-and-coming prospect. In this subsection, a simplifying flowchart is presented to clearly understand the processes involved in the supervised learning aspect of the current work.

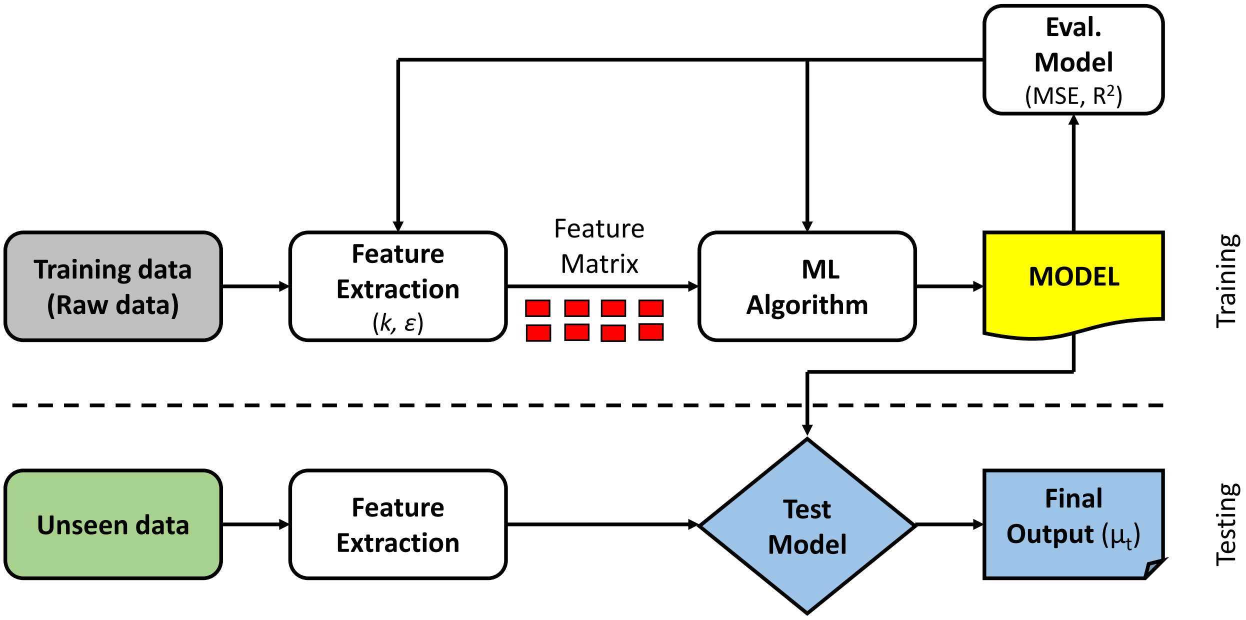


Figure 2.6. A schematic flow chart to explain the working of the supervised learning algorithm used in the current work. The diagram was inspired from the works of Nguyen et al. [42].

Figure 2.6 shows a typical flowchart for a supervised learning algorithm for regression. Firstly, the raw data is processed to identify relevant features which will aid the best algorithm to satisfy the dataset. Next, the extracted feature matrix (k and ϵ is the present case) is passed through the training model where ML algorithms construct a model that satisfactorily maps the input to output (eddy viscosity) values. The evaluation model often consists of an optimization algorithm that feedbacks into the feature extraction and learning models to minimize error as we move forward. This feedback process continues until a desired level of accuracy has been achieved. The constructed model is then used to predict output values for unseen (testing) data.

3. COMPUTATIONAL METHODOLOGY

This chapter presents a detailed account of the computational modeling approach used in the present work. Section 3.1 gives a general overview of the turbulence modeling approach used in the present work. Sections 3.2 and 3.3 show a detailed walkthrough of the mathematical aspect of the RANS-based and LES analysis, respectively. Section 3.4 gives a clear idea about the geometrical aspects of the test case and simulation details.

3.1 Overview of Modeling Approach

In this study, both RANS and LES models were employed to evaluate the fluid flow data and collect training data for the machine learning model. Both k-ϵ and k-ω models were used in RANS simulations, and the results were validated with the previous literature. For LES simulations, Smagorinsky-Lilly and Wall-Adapting Local Eddy-Viscosity (WALE) model were employed, and the results were validated using experimental data. The LES velocity and pressure data are used to derive the turbulent kinetic energy (k), and the rate of dissipation (ϵ), which subsequently serve as the training data for the ML model to predict eddy viscosity (μ_t).

For a Newtonian fluid, the viscosity relation where the shear stress between fluid layers is linearly dependent on the velocity gradient is valid. In the case of a turbulent flow, Newton's constitutive relation for eddy viscosity (as shown in equation 3.1) implies that Reynolds stress is a linear function of the velocity gradient.

$$\langle u'_{\mathrm{i}} u'_{\mathrm{j}} \rangle = -2\nu_t S_{\mathrm{ij}} + \frac{2}{3}\delta_{\mathrm{ij}} k. \tag{3.1}$$

The constitutive relation is 'Boussinesq approximated[1]', and it uses eddy viscosity as the proportionality coefficient. Therefore these models are referred to as LEVMs or linear eddy viscosity models. The major challenge in modeling the Reynolds stress is the estimation of ν_t, and the following section addresses this challenge in various ways.

[1] ↑Boussinesq approximation ignores the variation in fluid properties other than density (ρ) and the density only appears when it is multiplied by the gravitational acceleration (g).

3.2 Ryenolds-Averaged Navier Stokes

RANS-based CFD tools are some of the most popular commercially used turbulence models available in today's date. They are computationally cheap and require less technical expertise than DNS or LES analysis. These qualities have significantly contributed towards RANS as an industry workhorse many decades after its conception.

In Reynolds decomposition, the flow variables are expressed as a superposition of two flow: the mean flow[2], and the fluctuation. Reynolds decomposition using using Pope's notation [21] is shown below:

$$\phi = \langle \phi \rangle + \phi', \tag{3.2}$$

$$\langle \phi' \rangle = 0, \tag{3.3}$$

where $\langle \square \rangle$ represents mean flow, and $\square'$ represents fluctuation. When we apply Reynolds decomposition to flow properties in the NS equations we get the following form:

$$\frac{\partial \langle u_{\mathrm{i}} \rangle}{\partial x_{\mathrm{i}}} = 0, \tag{3.4}$$

$$\frac{\partial \langle u_{\mathrm{i}} \rangle}{\partial t} + \langle u_{\mathrm{j}} \rangle \frac{\partial \langle u_{\mathrm{i}} \rangle}{\partial x_{\mathrm{j}}} = -\frac{1}{\rho}\frac{\partial \langle p \rangle}{\partial x_{\mathrm{i}}} + \nu \frac{\partial^2 \langle u_{\mathrm{i}} \rangle}{\partial x_{\mathrm{j}} \partial x_{\mathrm{j}}} - \frac{\partial \langle u'_{\mathrm{i}} u'_{\mathrm{j}} \rangle}{\partial x_{\mathrm{j}}}. \tag{3.5}$$

The term $\langle u'_{\mathrm{i}} u'_{\mathrm{j}} \rangle$ also known as the Reynolds stress tensor leads to a closure problem as the number of unknowns are greater than the number of available equations. The unclosed set of equations can not be solved unless the Reynolds stress term is determined in terms of averaged quantities. The Reynolds stress is similar to the viscous stress and therefore it is possible to model the transport relations for the Reynolds stress tensor starting with Navier-Stokes equations. However, such derivations lead to further unclosed equations requiring higher order correlations (Section 3.2.1).

[2] ↑Also known as Reynolds averaged or Ensemble averaged flow.

3.2.1 Reynolds Stress Closure

The explicit transport equations of Reynolds stresses can be obtained by simplifying and taking moments of NS equations, as shown below [21]:

$$\frac{\mathrm{D}}{\mathrm{D}t}\left\langle u_{\mathrm{i}}u_{\mathrm{j}}\right\rangle + \frac{\partial}{\partial x_k}T_{k\mathrm{ij}} = \mathcal{P}_{\mathrm{ij}} + \mathcal{R}_{\mathrm{ij}} - \epsilon_{\mathrm{ij}}. \tag{3.6}$$

To get an exact idea of the magnitude of the problem we are dealing with, the set of closed and unclosed parameters are shown in Table 3.1.

Table 3.1. Form of various parameters in the Reynolds-stress transport equation.

Parameters	Definition	Form
$\frac{\mathrm{D}}{\mathrm{D}t}\left\langle u_{\mathrm{i}}u_{\mathrm{j}}\right\rangle$	Mean-flow convection	Closed
$\mathcal{P}_{\mathrm{ij}}$	Production tensor	Closed
$T_{k\mathrm{ij}}$	Reynolds stress-flux	Unclosed
$\mathcal{R}_{\mathrm{ij}}$	Pressure strain-rate tensor	Unclosed
ϵ_{ij}	Dissipation tensor	Unclosed

The unclosed parameters require closure with the aid of additional transport equations in the form of ϵ which lead to a total of 7 transport equations. The Reynolds-stress models are equipped to capture the mean rotation or curvature in a flow. They are also suitable for characterizing secondary flow characteristics due to their anisotropic nature. The various closure models used in the present work have been outlined in the following sections.

3.2.2 Two-Equation Closure Models

Majority of models in this category generally involve modeling the turbulent kinetic energy (k) and a second adjunct parameter. The two most popular models of this kind are the k-ϵ and k-ω models.

k-ϵ Model

The k-ϵ model is the most versatile and widely used model in CFD to simulate turbulent flows among the two-equation closure models. The original motivation for developing the k-ϵ model was the improvement of mixing length models and better prediction of algebraic length scales in moderately complex turbulent flows. The description of turbulent properties of the flow is achieved by two transport equations (PDEs):

- The first transport equation is for the turbulent kinetic energy (k).
- The second transport equation is for the dissipate rate of the turbulent kinetic energy or ϵ.

The original k equation is shown below:

$$\frac{Dk}{Dt} = -\nabla \cdot T + P - \epsilon. \tag{3.7}$$

where the term $T_{\mathrm{i}} = \frac{1}{2}\langle u'_{\mathrm{i}}u'_{\mathrm{j}}u'_{\mathrm{j}}\rangle + \langle u'_{\mathrm{i}}p'\rangle/\rho - 2\nu\langle u'_{\mathrm{i}}S'_{\mathrm{ij}}\rangle$ remains unclosed and are replaced by the gradient transport model shown in Equations 3.8 and 3.9. The standard k-ϵ model by first introduced by Jones and Launder [43] in 1972 to overcome the many uncertainties in the exact transport equations of k and ϵ.

$$\frac{\mathrm{D}k}{\mathrm{D}t} = 2\nu_t|S|^2 - \epsilon + \nabla \cdot (\nu + \nu_t)\nabla k, \tag{3.8}$$

$$\frac{\mathrm{D}\epsilon}{\mathrm{D}t} = 2c_{\epsilon 1}\frac{\epsilon}{k}|S|^2 - c_{\epsilon 2}\frac{\epsilon^2}{k} + \nabla \cdot (\nu + \sigma_\epsilon \nu_t)\nabla \epsilon. \tag{3.9}$$

The eddy viscosity is related to k and ϵ in the following way,

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}, \tag{3.10}$$

where $C_{\epsilon 1} = 1.44$, $C_{\epsilon 2} = 1.92$, and $C_\mu = 0.09$. These constants have been evaluated by numerous iterations of data-fitting.

The k-ϵ model generally works well for planar flows with shear layer formation and recirculating flows. In addition, it is instrumental in flows where there is formation of a free-shear layer with a low-pressure gradient and channel flows where Reynolds stresses dictate matter. In today's date, it is the most robust turbulence model yet simple enough where we supply the initial and boundary conditions to simulate flow.

However, it requires the solution of two extra PDEs and hence computationally more expensive than mixing length models. It is generally unsuitable for flows with a significant adverse pressure gradient (APG), such as compressors and pumps. The k-ϵ model also shows poor results near the walls for channel flows which can be improved by implementing enhanced wall treatment while simulating flows.

k-ω Model

Like its other counterpart under the RANS umbrella, the k-ω model is a standard two-equation model used in CFD for turbulence analysis. The model was first independently put forth by Kolmogorov [44] and later by Saffman [45]. Wilcox [46] [47] continued to refine the model for many decades, revisiting the model in 2008 [48] to publish his final remarks, at which point it had reached widespread utility in the CFD community.

The transport PDEs solved in the k-ω model are for the turbulent kinetic energy (k), and specific rate of dissipation (ω). The equations are shown below:

$$\frac{\mathrm{D}k}{\mathrm{D}t} = 2\nu_t|S|^2 - C_\mu k\omega + \nabla \cdot (\nu + \nu_t)\,\nabla k, \tag{3.11}$$

$$\frac{\mathrm{D}\omega}{\mathrm{D}t} = 2c_{\omega 1}|S|^2 - c_{\omega 2}\omega^2 + \nabla \cdot (\nu + \sigma_\omega k/\omega)\,\nabla\omega + (\sigma_d/\omega)\,\nabla k \cdot \nabla\omega. \tag{3.12}$$

The eddy viscosity relation is given as:

$$\mu_t = \rho\frac{k}{\omega}. \tag{3.13}$$

The parameters are modified by substituting $\epsilon = C_{\mu} k\omega$ which makes the ω equation adhoc. The last term of the ω transport equation, $(\sigma_d/\omega)\, \nabla k \cdot \nabla \omega$ is called the cross-diffusion term which aids in reducing the free-stream susceptibility of the model [48].

3.3 Large Eddy Simulation

Large-eddy simulation is a trade-off between the DNS and RANS modeling in terms of accuracy, where the larger scales are filtered, and the minor scales are modeled according to the filtering operation employed. Similar to RANS decomposition, the LES filtering operation is shown below:

$$\phi = \tilde{\phi} + \phi'', \tag{3.14}$$

where $\tilde{\square}$ represents the filtered (or resolved) component, and $\square''$ represents the residual (or subgrid-scale, SGS) component. The filtering operation is inherently mesh-dependent, as the grid resolution decides the smallest scales of motion.

The filtering operation is applied to the NS equations to derive the velocity field, which gives us the filtered momentum and continuity equations.

$$\frac{\partial \tilde{u}_{\mathrm{i}}}{\partial x_{\mathrm{i}}} = 0, \tag{3.15}$$

$$\frac{\partial \tilde{u}_{\mathrm{i}}}{\partial t} + \tilde{u}_{\mathrm{j}} \frac{\partial \tilde{u}_{\mathrm{i}}}{\partial x_{\mathrm{j}}} = -\frac{1}{\rho} \frac{\partial \tilde{p}}{\partial x_{\mathrm{i}}} + \nu \frac{\partial^2 \tilde{u}_{\mathrm{i}}}{\partial x_{\mathrm{j}} \partial . x_{\mathrm{j}}} - \frac{\partial \tau_{\mathrm{ij}}}{\partial x_{\mathrm{j}}} \tag{3.16}$$

The momentum equation contains the the term, τ_{ij}, otherwise known as the residual stress tensor (or SGS stress tensor) which is an unclosed equation due to the unresolved terms arising from residual motion. This closure problem can be resolved by an eddy-viscosity model proposed by Smagorinsky, which calculates the SGS stress tensor as shown in Equation 3.17.

$$\tau_{\mathrm{ij}} = -2\nu_{\mathrm{t}} \bar{S}_{\mathrm{ij}}, \tag{3.17}$$

where $\bar{S}_{\mathrm{ij}} = \frac{1}{2}\left(\frac{\partial \bar{u}_{\mathrm{i}}}{\partial x_{\mathrm{j}}} + \frac{\partial \bar{u}_{\mathrm{j}}}{\partial x_{\mathrm{i}}}\right)$ is called the strain rate deformation tensor. The eddy viscosity term is modeled as function of grid resolution as shown below:

$$\mu_{\mathrm{t}} = \rho \left(C_s \Delta_g\right)^2 \sqrt{2\bar{S}_{\mathrm{ij}}\bar{S}_{\mathrm{ij}}}, \tag{3.18}$$

where C_s is known as the Smagorinsky model constant and Δ_g is the grid spacing. Near the wall-region, the required grid points for solving LES increases as $\sim Re^{1.8}$ [49]. This makes LES an impractical candidate for high Re channel flows. Implementing *wall models* to relax the resolution sensitivity in the near wall region is a viable plan.

3.4 Back-Step Test Case and Simulation Details

Figure 3.1 shows the schematic diagram for computational domain of the back-step used for fluid flow and preliminary machine learning analysis. The geometry of the test case was adopted from the experimental setup of Armaly et al. [50] as shown in Figure 3.1. The expansion ratio (ER) is defined as $ER = H/h$, where H is the total channel height, h is the inlet height, and S is the step height.

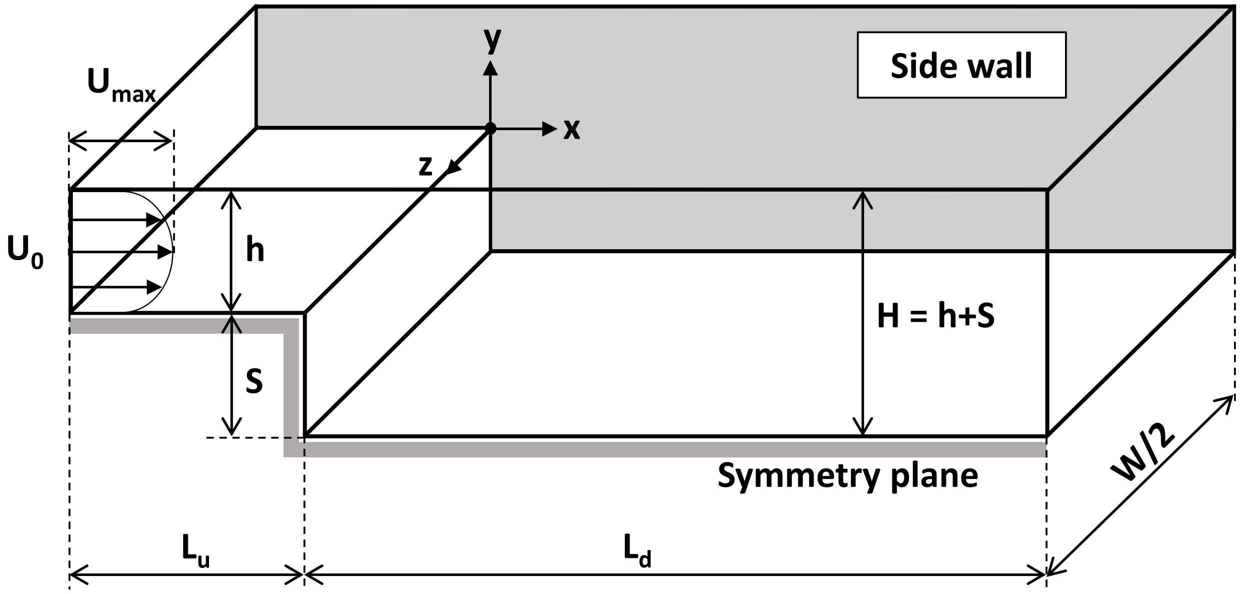


Figure 3.1. A schematic diagram of the back-step test case used for turbulence and machine learning analysis.

The simulations were carried out for four values of ER, i.e., 1.5, 2, 2.5, and 3 in accordance with the experimental setup by Armaly et al. [50]. Note that all domain measurements have been scaled into dimensionless quantities on the basis of inlet height, h. For a 3-D model, the span-wise width, W/h is set as 10, whereas Figure 3.1 shows half of the total span-wise thickness bisected by a plane of symmetry. For two-dimensional simulations, the symmetric plane was used as a domain. Additional simulations carefully checked the assumption of span-wise symmetry in the 3-D flow based on periodic boundary conditions in the span-wise direction. The computed results for flow properties seem to vary negligibly in the span-wise direction, e.g., the error percentage in the re-circulation length with and without the span-wise width was <0.01%.

The upstream and downstream channel lengths are L_u, and L_d respectively. The upstream length, $L_u \geqslant 5h$ does not affect the flow predictions, therefore, $L_u = 5h$ was chosen as the suitable limit. A short downstream distance has the disadvantage of hampering flow characteristics and preventing it from becoming full-developed. It was observed that $L_d = 15h$ was a suitable measurement to allow the flow to become fully-developed at the outlet. The inlet channel height, h is used for calculating the flow Reynolds number, $R\text{e} = \frac{U_b h}{\nu}$, where U_b denotes the bulk velocity at the inlet [51].

Figure 3.2 shows the mesh resolution for the computational domain used in the present work. The grid points are particularly refined in the vicinity of inlet and the re-circulation zone to capture the accurate essence of the re-circulation length. To present a reliable view of the grid, the area near the step (where re-circulation occurs) is zoomed in for better clarity.

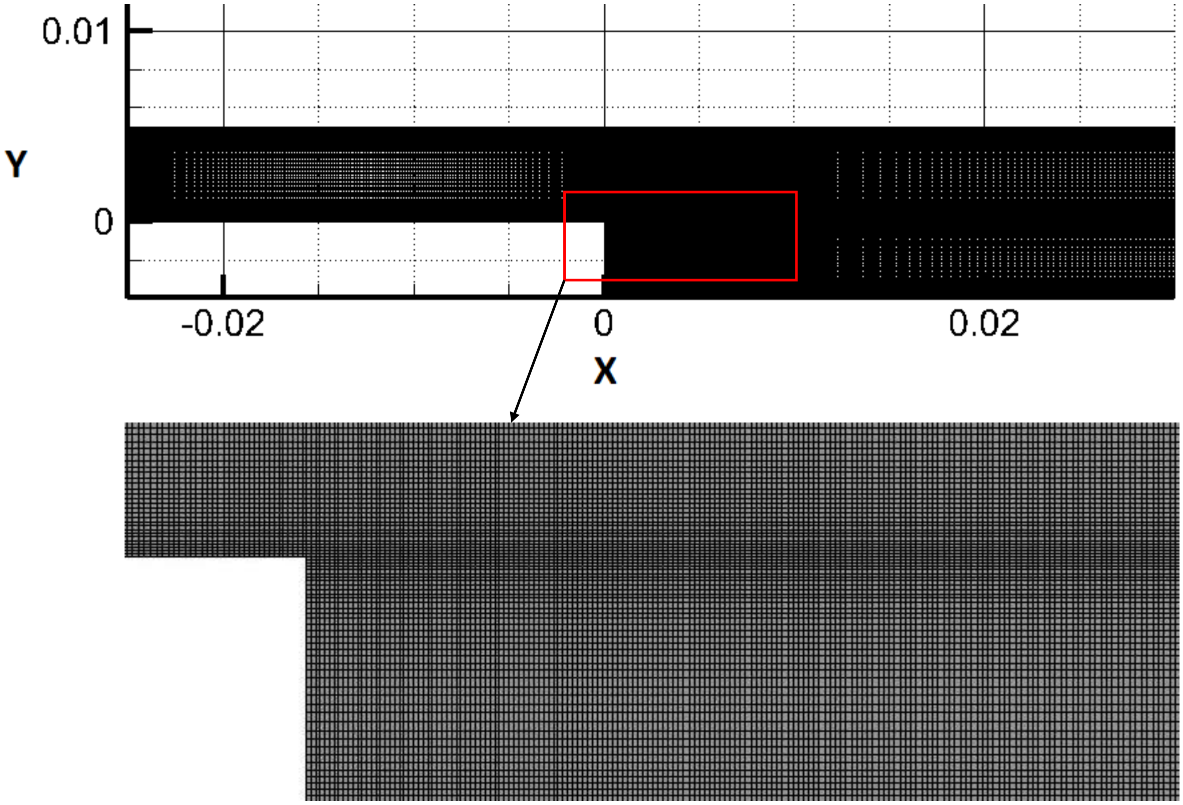


Figure 3.2. The grid resolution used for 2-D simulations with a zoomed view of the re-circulation zone.

3.4.1 Initial and Boundary Conditions

The initial and boundary conditions for the simulations in the present work are discussed below in Table 3.2:

Table 3.2. The list of initial and boundary conditions used in the current work.

Parameters	IC/BC
Turbulence models	k-ϵ, k-ω, and LES models
Reynolds number	Four values, $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$
Inlet	Parabolic velocity inlet, with a Turbulent intensity = 13%, and Length scale = 0.01 mm
Outlet	Pressure outlet
Walls	No-slip boundary condition

This chapter summarizes the details of the turbulence aspect of the present study, the overview of turbulence models, and the simulation details about the back-step test case used in the current work. This is a precursor to the turbulence results shown in Chapter 4 which establishes a comparative study between the experimental, LES, and RANS-based results. The comparison is crucial to correlate sound turbulence analysis with machine learning used to improve upon existing norms. This will enable us to construct a reduced order turbulence model to improve RANS predictions, which shows under-prediction for flow properties due to limited closure.

4. BACK-STEP TURBULENCE SIMULATION RESULTS AND DISCUSSION

This chapter presents detailed parametric studies for the properties of flow over a back-step. Section 4.1 shows a detailed grid convergence analysis, Sections 4.2 and 4.3 show the parametric studies to investigate effect of Reynolds number, expansion ratio, and wall functions on the flow domain. Sections 4.4 and 4.5 present a comparison of LES and RANS data to highlight the limitations of RANS modeling.

4.1 Grid Independence Study

A detailed grid-convergence study for the current domain is presented in Section 4.1.1. The convergence study is carried out for five different grids with increasing resolution, as shown in Figure 4.1, and a moderate grid is chosen to plots all results. The parameters chosen for the grid convergence test are the re-circulation length (X_r), coefficient of pressure (C_p), and coefficient of skin friction (C_f).

4.1.1 Grid Convergence Study

A total of five grids were used for the grid convergence test, and a multiplicative parameter P was used as the index to decide the grid resolution. The details of grid points are shown in Table 4.1.

Table 4.1. Grid parameters for the resolutions used for the grid independence study.

Grid (At $Re = 35,000$)	N_x	N_y	N_z	N_{xyz}	$\Delta y_{min}/h$
$P = 0.25$	15	13	10	1950	0.2
$P = 0.5$	31	25	20	15500	0.1
$P = 1$	62	50	40	124000	0.05
$P = 2$	125	100	80	1×10^6	0.025
$P = 4$	250	200	120	8×10^6	0.0125

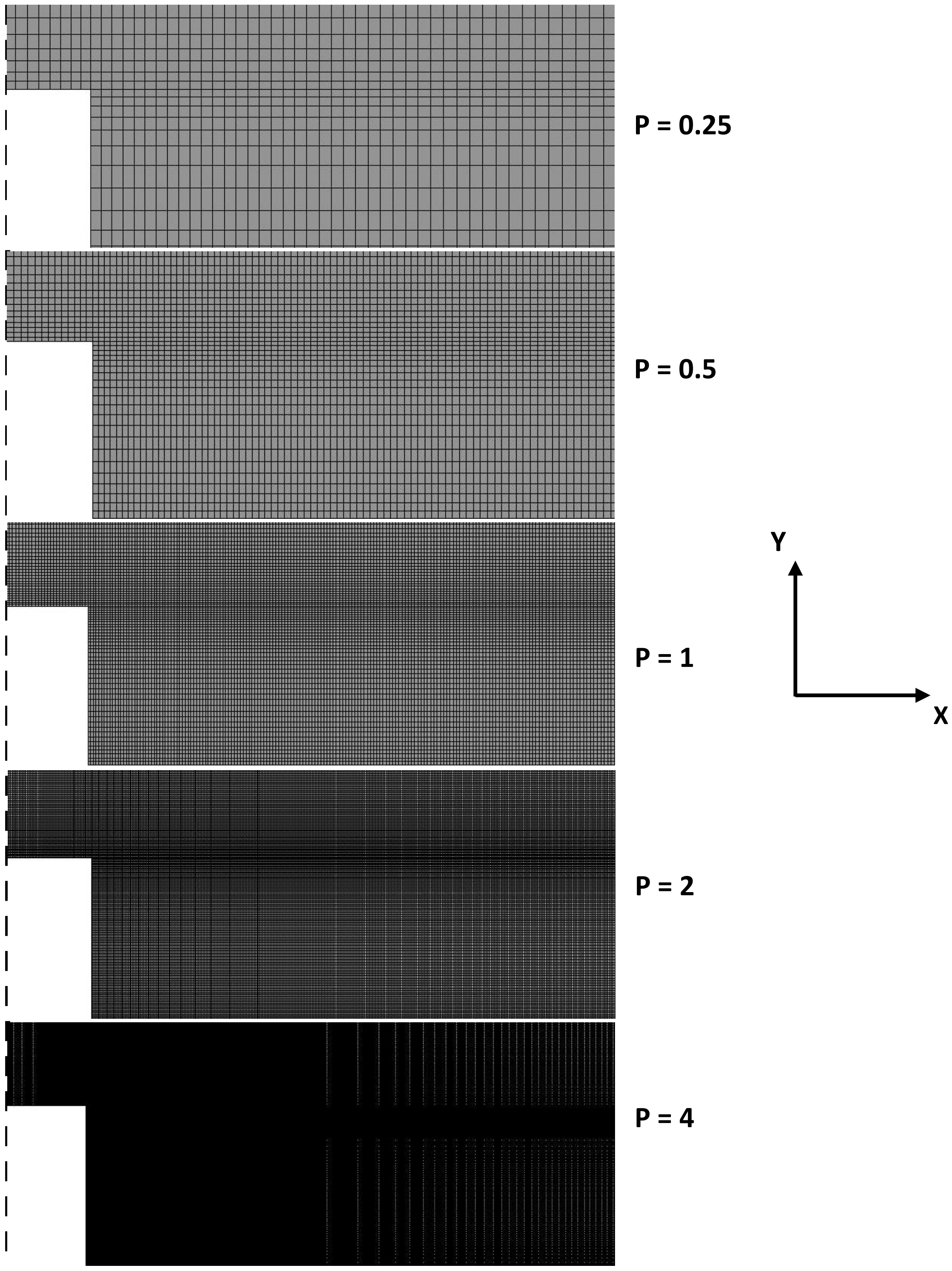


Figure 4.1. A schematic diagram of the different mesh resolutions used for the present analysis.

Figure 4.1 represents the schematic diagram of all five grid resolutions in and around the re-circulation zone used in the present work. The grids are presented in increasing order of grid resolution indicated by a multiplicative factor P. The grids $P = 0.25, 0.5$ represent coarse grids, $P = 1$ represents a medium grid and $P = 2, 4$ represent fine grid meshes. A detailed grid convergence analysis is shown in Section 4.1.2. The normalized re-circulation length (X_r/S) for $Re = 20,000$ and $ER = 1.5$ is the chosen parameter for this analysis.

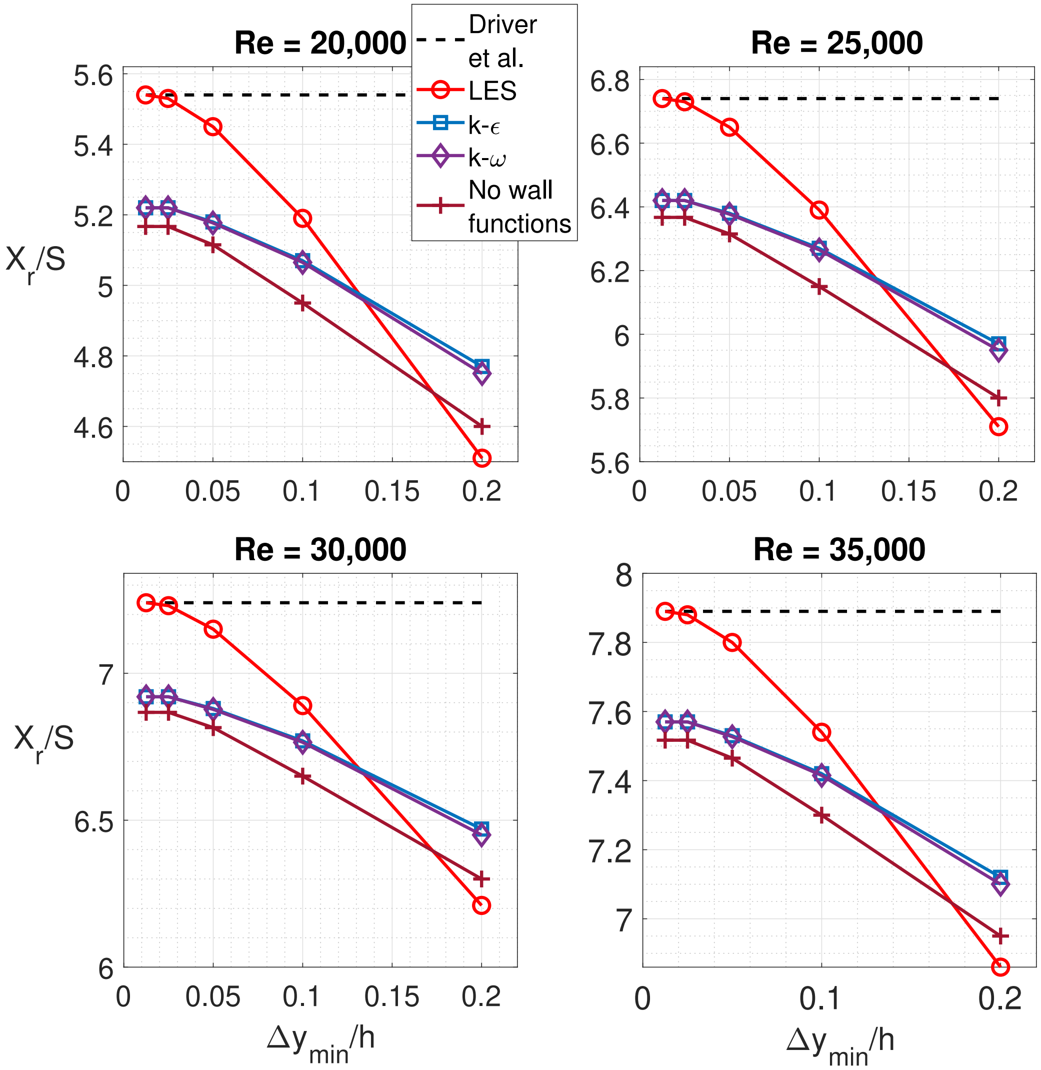


Figure 4.2. Grid convergence plot for re-circulation length (X_r) normalized by step height (S) for different Reynolds number (experimental data is obtained from Driver et al. [52]).

4.1.2 Grid Convergence Analysis

Table 4.2 shown below presents the grid data and parameter comparison for the preliminary grid convergence analysis.

Table 4.2. Normalized re-circulation length for various grid resolutions.

Normalized Grid	Grid spacing	Cell count	X_r/S
1	1	124000	5.48756
2	2	$1{\times}10^6$	5.52875
3	4	$8{\times}10^6$	5.53999

As we are doing the analysis for three grid resolutions, doubling the cell count in each axis, hence our refinement ratio, $r = 2$. We calculate the order of convergence, P, using the equation below:

$$P = \left| ln(\frac{P_3 - P_2}{P_2 - P_1})/ln(r) \right|. \tag{4.1}$$

P is obtained as,

$$P = \left| ln\left(\frac{5.53999 - 5.52875}{5.52875 - 5.48756}\right)/ln(2) \right| = 1.87392. \tag{4.2}$$

Technically, a second order solver would have $P = 2$, but the difference can be chalked off to grid-stretching, non-linearities in the discretization, and boundary conditions defined for the specific problem.

The Grid convergence index (GCI) is calculated by using the formula:

$$GCI = \frac{F_s|\mathrm{e}|}{r^P - 1}, \tag{4.3}$$

where e is the error between the two grids and F_s is a safety factor.

By choosing a Wilcox safety factor of $F_s = 1.25$, the coarse and medium GCI are calculated as follows:

$$GCI_{12} = \frac{1.25|(5.48756 - 5.52875)/5.48756| \times 100\%}{2^{1.87392} - 1} = 0.352.\% \tag{4.4}$$

$$GCI_{23} = \frac{1.25|(5.52875 - 5.53999)/5.52875| \times 100\%}{2^{1.87392} - 1} = 0.095.\% \tag{4.5}$$

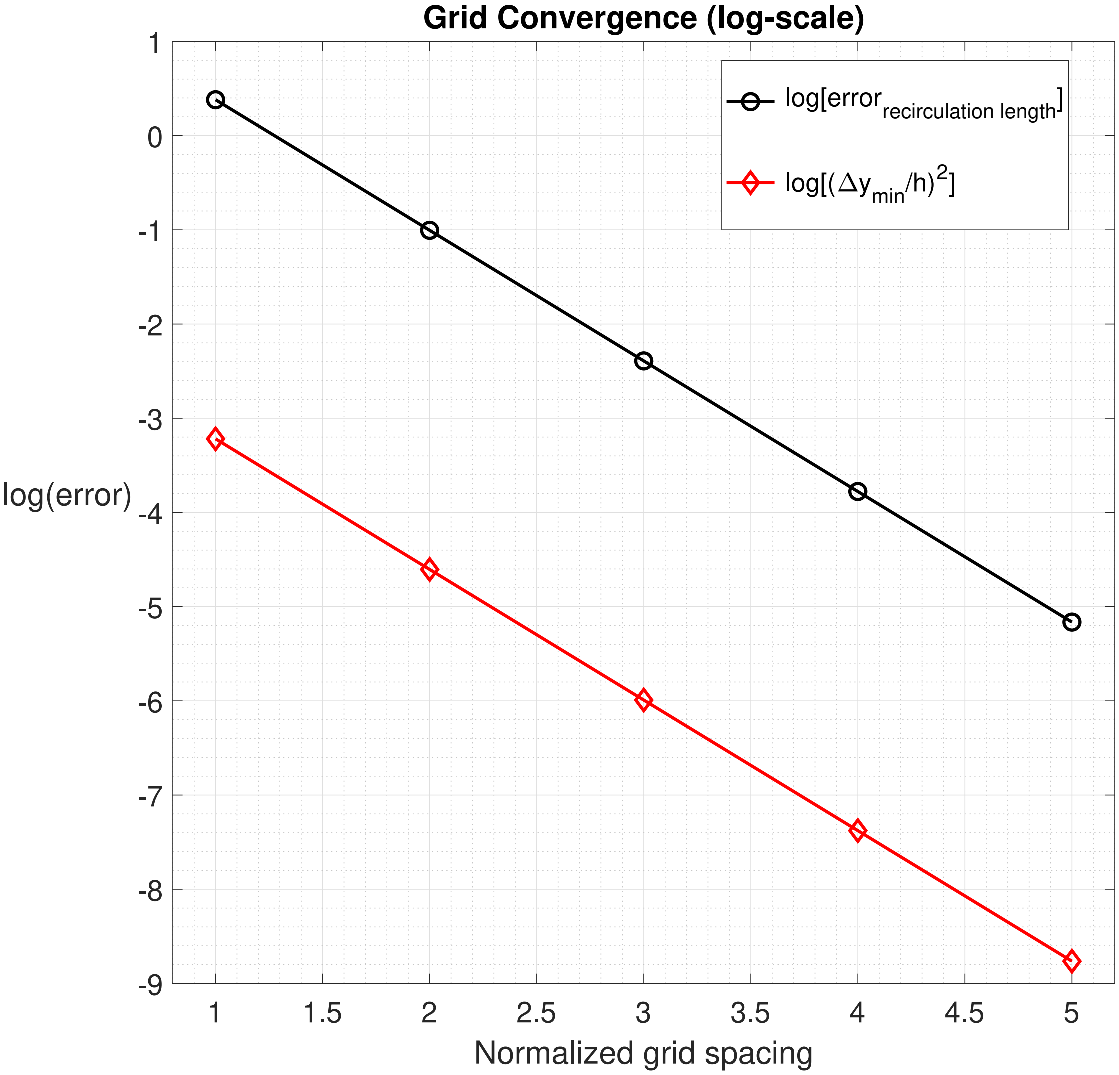


Figure 4.3. Log-scale plot of re-circulation length error and square of minimum grid spacing.

Figure 4.2 represents the grid convergence plot for re-circulation length (X_r) normalized by step height (S) for $ER = 1.5$ and different Reynolds number. The experimental data validation for the re-circulation length is obtained from Adams et al.[53]. We observe an increase in re-circulation length with increasing Reynolds number for the same expansion ratio. Along with the LES, the k-ϵ, k-ω, and k-ω model without wall functions are shown in Figure 4.2. As evident from the figure, the RANS models (with and without wall functions) show a sizeable difference for the re-circulation length data, even for the finest grid resolutions.

Figure 4.3 shows the log-scale plot of re-circulation length error and square of minimum grid spacing. As we simulate the flows using a second-order discretization scheme, the log plot of the X_r/s error should form a straight line with a $slope = 2$. To confirm this finding, we have also plotted the log-scale plot of the square of the minimum grid spacing in a normalized manner. As both the lines are parallel to each other, it can be confirmed that the grid has reached an asymptotic convergence in the second order.

Figures 4.4 and 4.5 represent the grid convergence plot of the coefficient of skin friction (C_f), and coefficient of pressure (C_p), respectively. All plots shown below are simulated using the LES model having an $ER = 1.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$. The experimental data for plot validation was obtained from Adams et al.[53]. For both the plots shown below, we can observe good convergence as we move from $P = 1$ to $P = 4$. The plots converge towards the experimental curve obtained from literature. $P = 4$ and $P = 2$ give equally good results for C_f and C_p hence $P = 2$ grid is used for all the results shown in the following sections. The grid convergence analysis enables us to choose the best grid for our analysis without running the risk of encountering errors in flow properties due to mesh-related discretization. A reliable grid will present accurate results for the high fidelity LES data, which further aid our reduced order turbulence model as a source of training data.

The experimental data used for the validation of the LES results are obtained from four different studies pertaining to different values of expansion ratios. Data from the works of Driver et al. [52], used to validate the LES results for $ER = 1.5$, shows the deployment of a backward-facing step in a subsonic wind tunnel setup. The static pressure on the wall near the step was measured using $0.2mm$ diameter slits placed along the test section centerline. the uncertainty in measuring the static pressure coefficient near the wall is assessed to be ±0.9% with a confidence limit of 95%. The skin friction drag is measured using an oil flow laser interferometer technique [52]. This technique produces an uncertainty of ±8% in the measurement of the skin friction coefficient of the step sidewall with a confidence limit of 95%. The re-circulation length is also measured using the same technique as the skin friction drag and shows an uncertainty of ±6%, these are represented by the error bars in Figure 4.17.

The data presented by Adams et al. [53] used to validate the LES results for $ER = 2$ also deploys a low-speed wind tunnel with a backward step channel to obtain experimental calculations. the static pressure coefficient near the step sidewall was measured by placing intermittent pressure tap sensors along the wall. These sensors project an uncertainty of ±0.5% at a reference velocity of $11m/s$. The skin friction drag was measured by employing a pulsed wire probe as described in the studies published by Westphal et al.[54]. The three wires present in the pulsed wire probe can easily measure both the magnitude and direction of skin friction and work especially well in regions of reversing flow. The uncertainty for skin friction is projected at ±5%. The re-circulation length is measured using a thermal tuft, which is not unlike a pulsed wire probe: the uncertainty for the re-circulation is calculated at $\pm 0.1S$ (where S is the step height).

The findings of Kim et al. [55] used for the validation of the LES results for $ER = 2.5$ employed a custom-made manometric transducer inside a subsonic quiet wind tunnel. The transducers strips were placed along the step sidewall to measure the static pressure coefficient with an uncertainty of $\pm 0.025mm$ of water. A combination of thermal tufts and oil flow visualization techniques were employed to observe the flow in the re-circulation zone as well as to measure the skin friction drag. The uncertainty in re-circulation length is projected as $X_r/S = 7 \pm 0.35$ and also represented in Figure 4.17. The data from Eaton et al. [56] used for validating the LES results for $ER = 3$ employ hot wire anemometers inside a subsonic wind tunnel to measure the general turbulence features for a backward-facing step. The uncertainty in the measure of re-circulation length is presented in Figure 4.17.

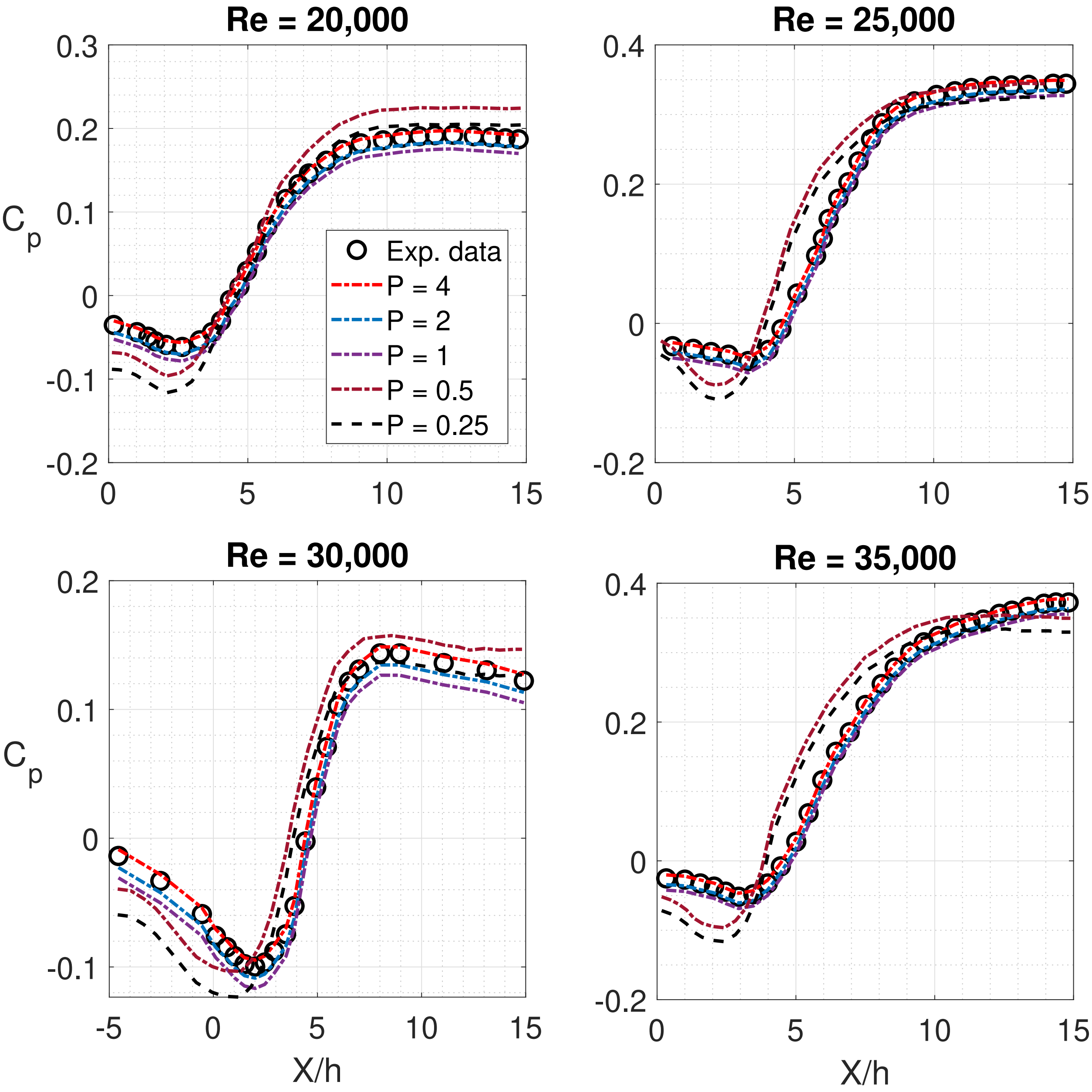


Figure 4.4. Grid convergence test for coefficient of pressure (C_p) for five different grid resolutions ($ER = 1.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$).

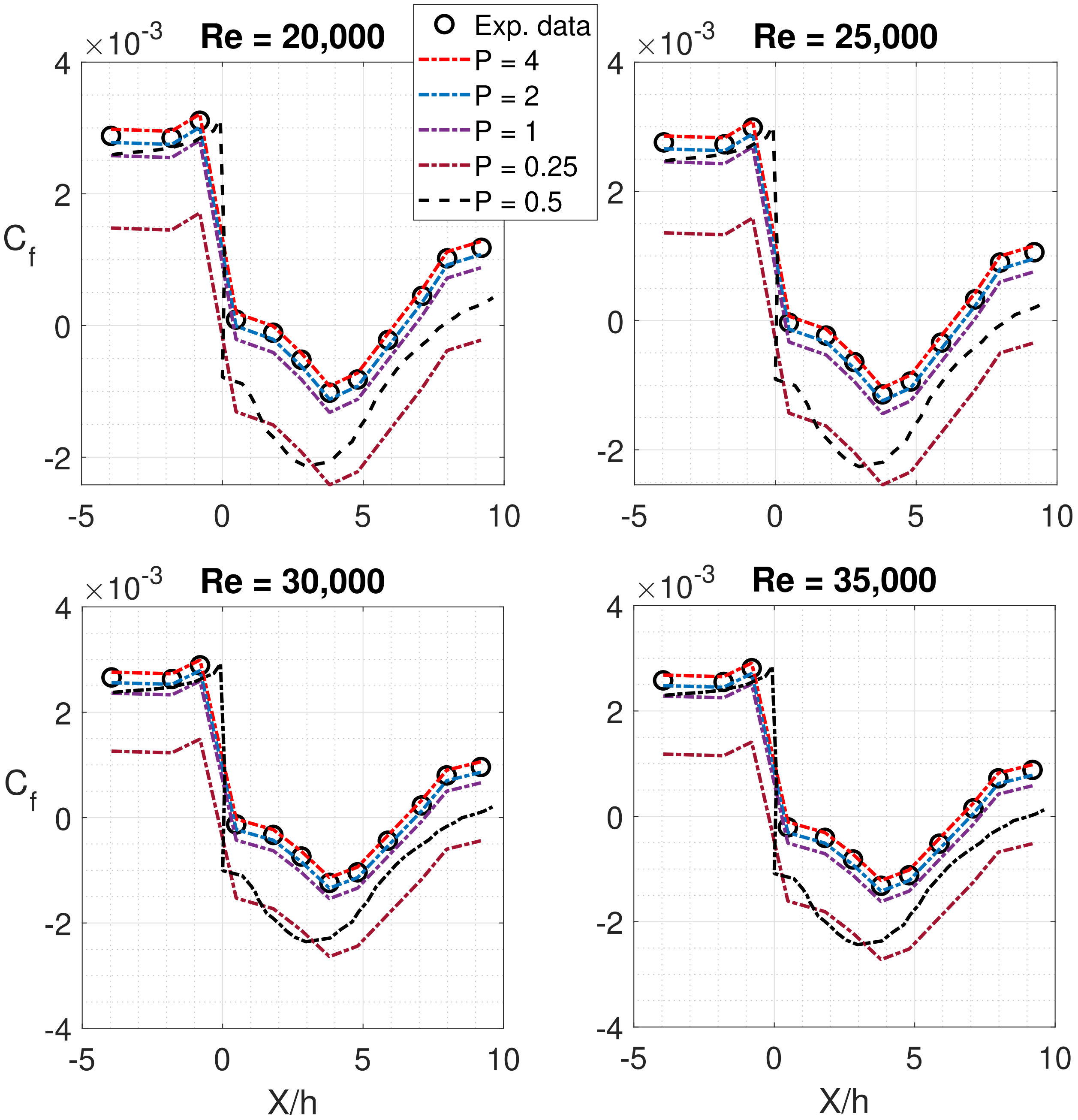


Figure 4.5. Grid convergence test for coefficient of skin friction (C_f) for five different grid resolutions ($ER = 1.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$).

4.2 Effect of Reynolds Number and Expansion Ratio

The coefficient of pressure at a point in the vicinity of a bluff body is generally independent of the body dimensions. As a result, a model of the actual body can be tested using a water or wind tunnel, and the C_p data can be indiscriminately used to predict the pressure near the critical points of a full-size engineering object. The response of a flow system towards change in geometrical flow domain and flow velocity is the most common approach towards determining the system's stability. The studies shown in this section consider two flow features, namely, coefficients of pressure and skin friction, to determine the system's sensitivity towards changes in Reynolds number and expansion ratio.

Figures 4.6-4.9 show the variation of coefficient of pressure (C_p) for $ER \in \{1.5, 2, 2.5, 3\}$ and $R\text{e} \in \{2, 2.5, 3, 3.5\} \times 10^4$. The experimental data for result validation was obtained from Adams et al.[53]. All the C_p plots shown below pertain to the lower wall of the flow domain, adjacent to the step. The plots show a positive bias towards an increase in expansion ratio, as the extreme value can be seen as increasing as the step height increases. The plots change characteristics as we increase the Reynolds number; they tend to broaden more towards the step vicinity are as we approach higher $R\text{e}$ values. The increase in the C_p value as we go past the step region in the downstream direction implies a re-circulation zone in that location.

The coefficient of skin friction (C_f) follows Prandtl's one-seventh-power law for turbulent flow systems and is related to the Reynolds number in the following way:

$$C_f = \frac{0.027}{R\text{e}_x^{1/7}}. \tag{4.6}$$

where x is the distance from the reference point of boundary layer formation.

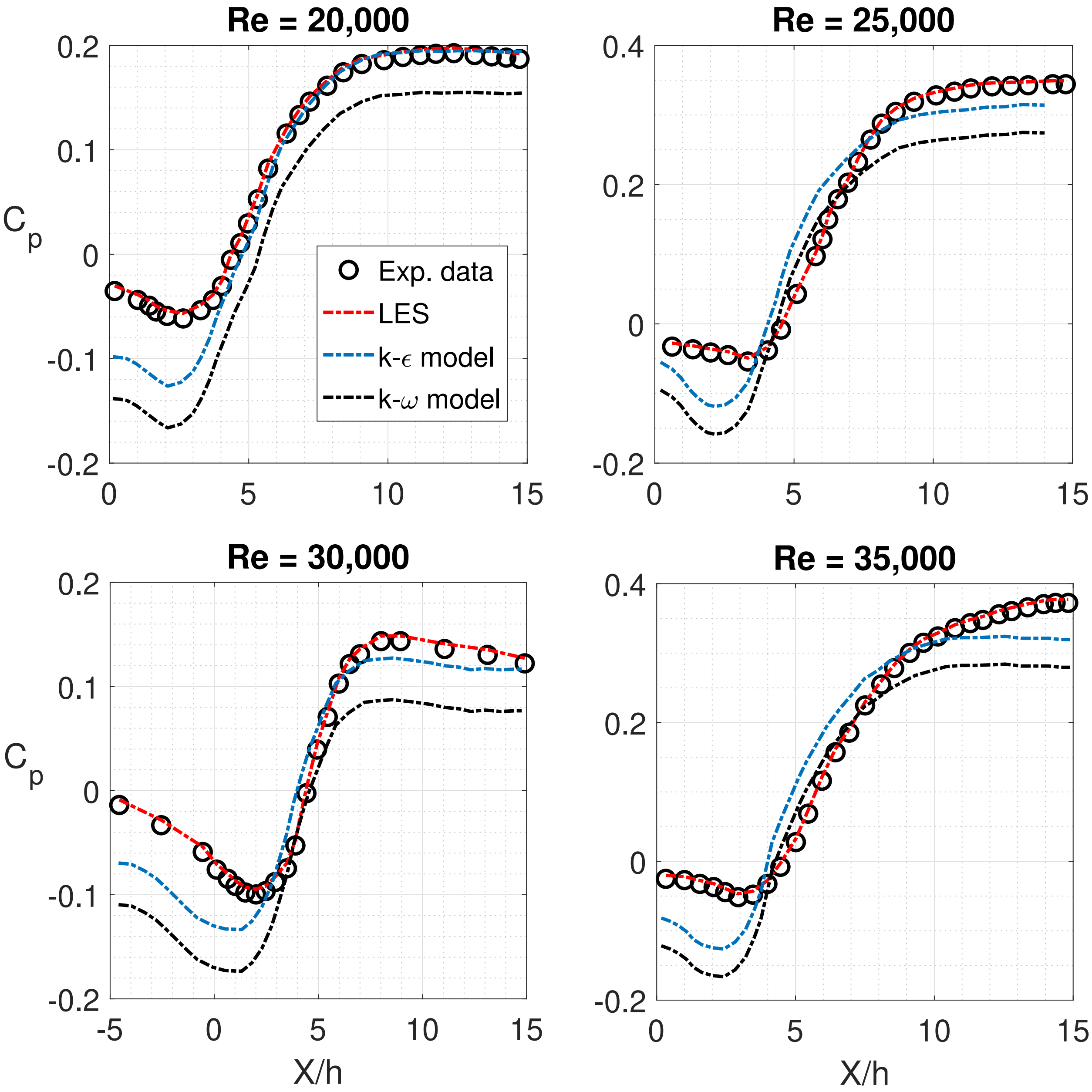


Figure 4.6. Variation of coefficient of pressure (C_p) for $ER = 1.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Driver et al. [52]).

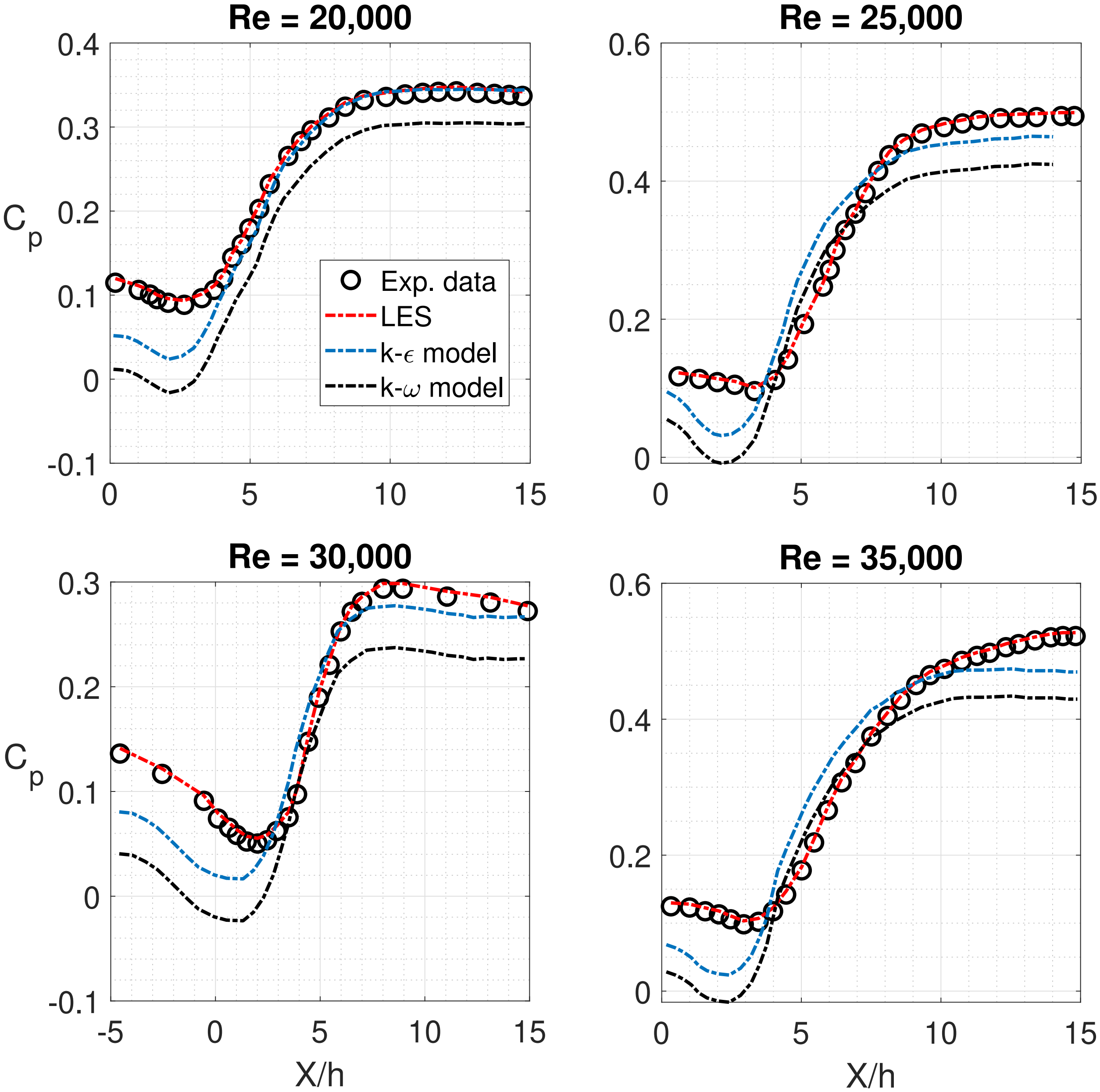


Figure 4.7. Variation of coefficient of pressure (C_p) for $ER = 2$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Adams et al. [53]).

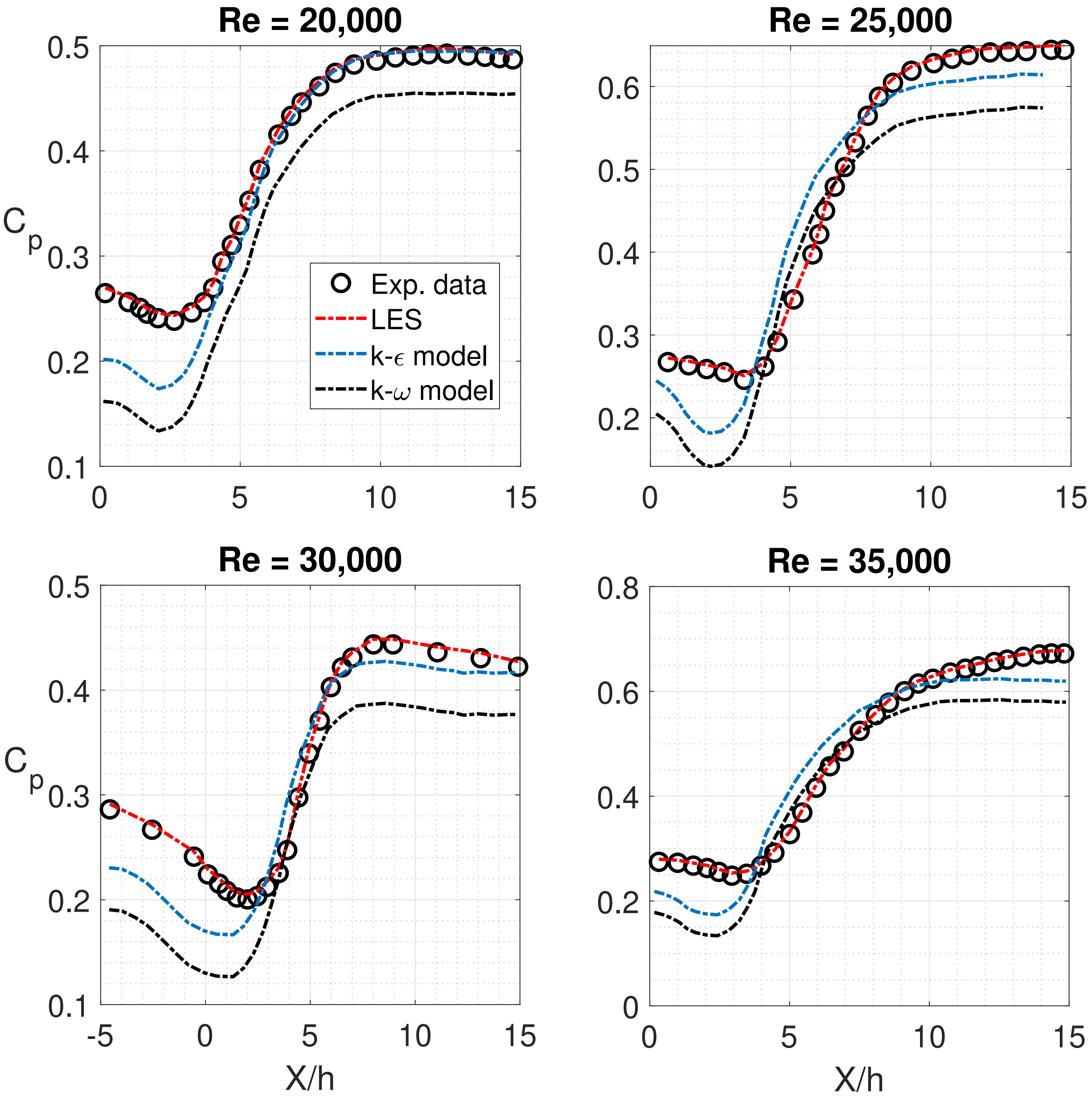


Figure 4.8. Variation of coefficient of pressure (C_p) for $ER = 2.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Kim et al. [55]).

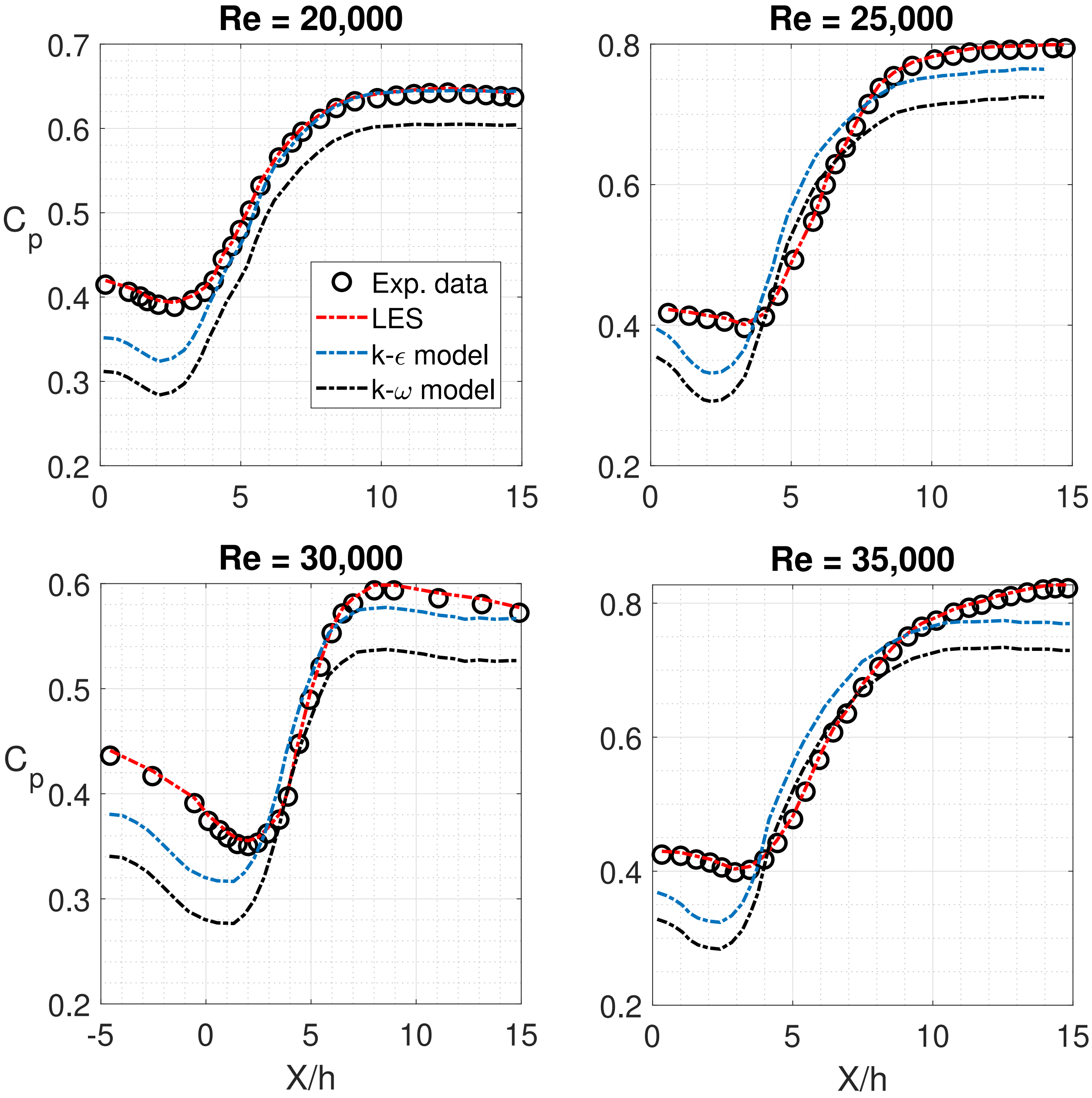


Figure 4.9. Variation of coefficient of pressure (C_p) for $ER = 3$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Eaton et al. [56]).

Skin friction drag is a component of viscous drag, caused by the fluid viscosity, and evolves from laminar to turbulent drag as the body moves through a fluid. Figures 4.10-4.13 show the variation of coefficient of skin friction (C_f) for $ER \in \{1.5, 2, 2.5, 3\}$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$. As shown in Equation 4.6, the coefficient of skin friction is inversely proportional to the seventh root of Re. We can observe a decrease in the extreme values of the C_f plot as we increase the Re, although there no change in the shape of the plots.

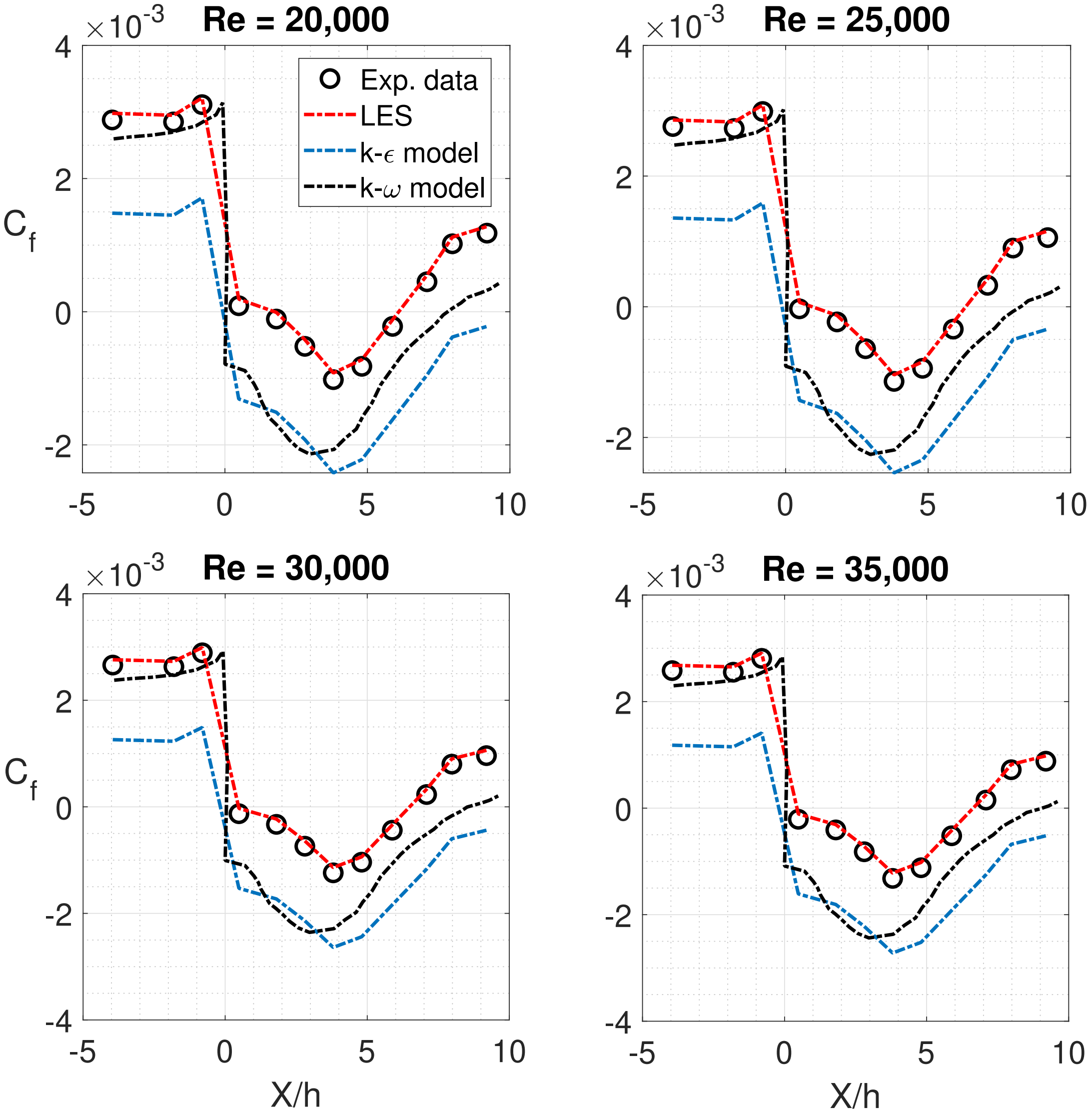


Figure 4.10. Variation of coefficient of skin friction (C_f) for $ER = 1.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Driver et al. [52]).

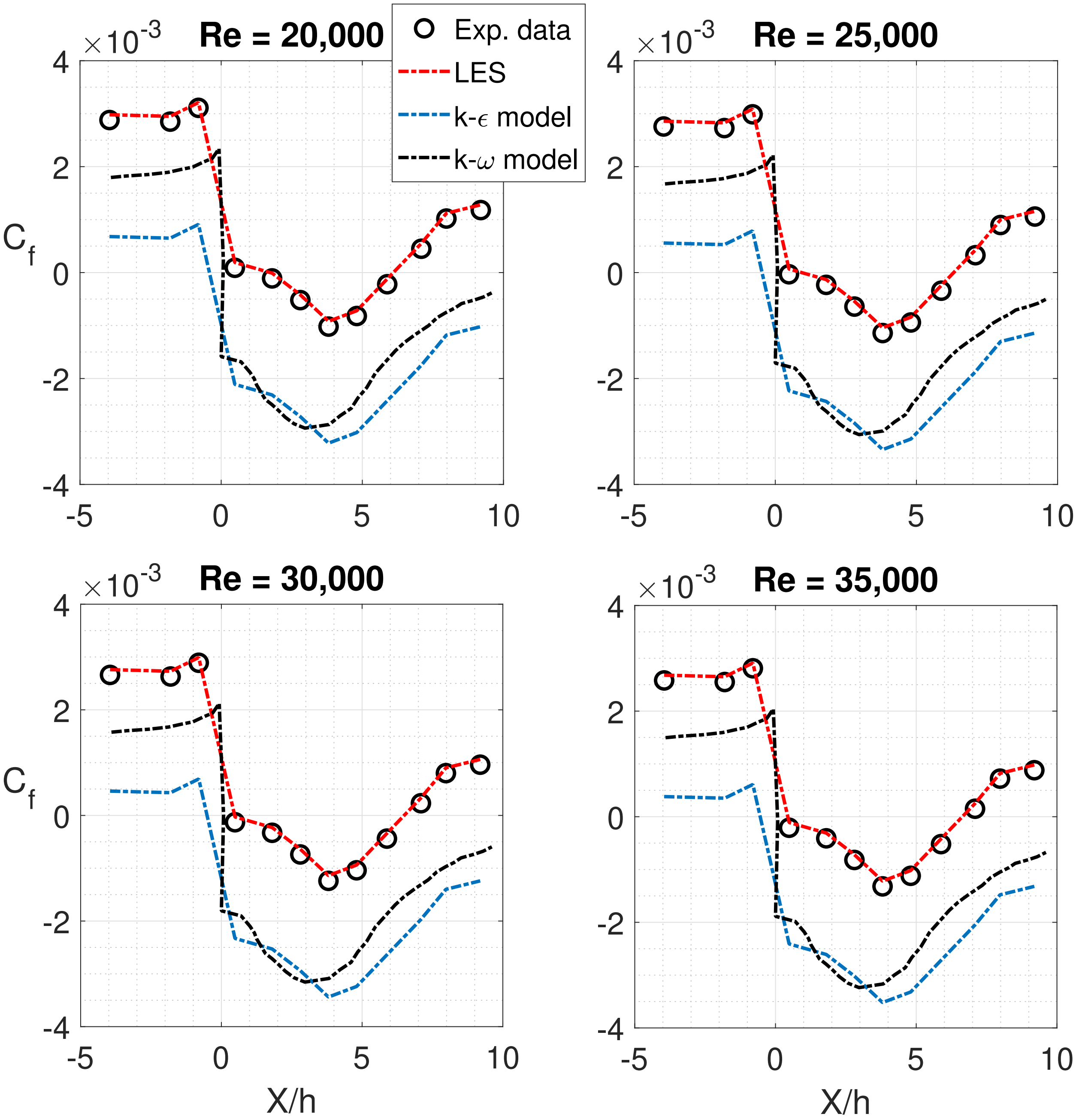


Figure 4.11. Variation of coefficient of skin friction (C_f) for $ER = 2$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Adams et al. [53]).

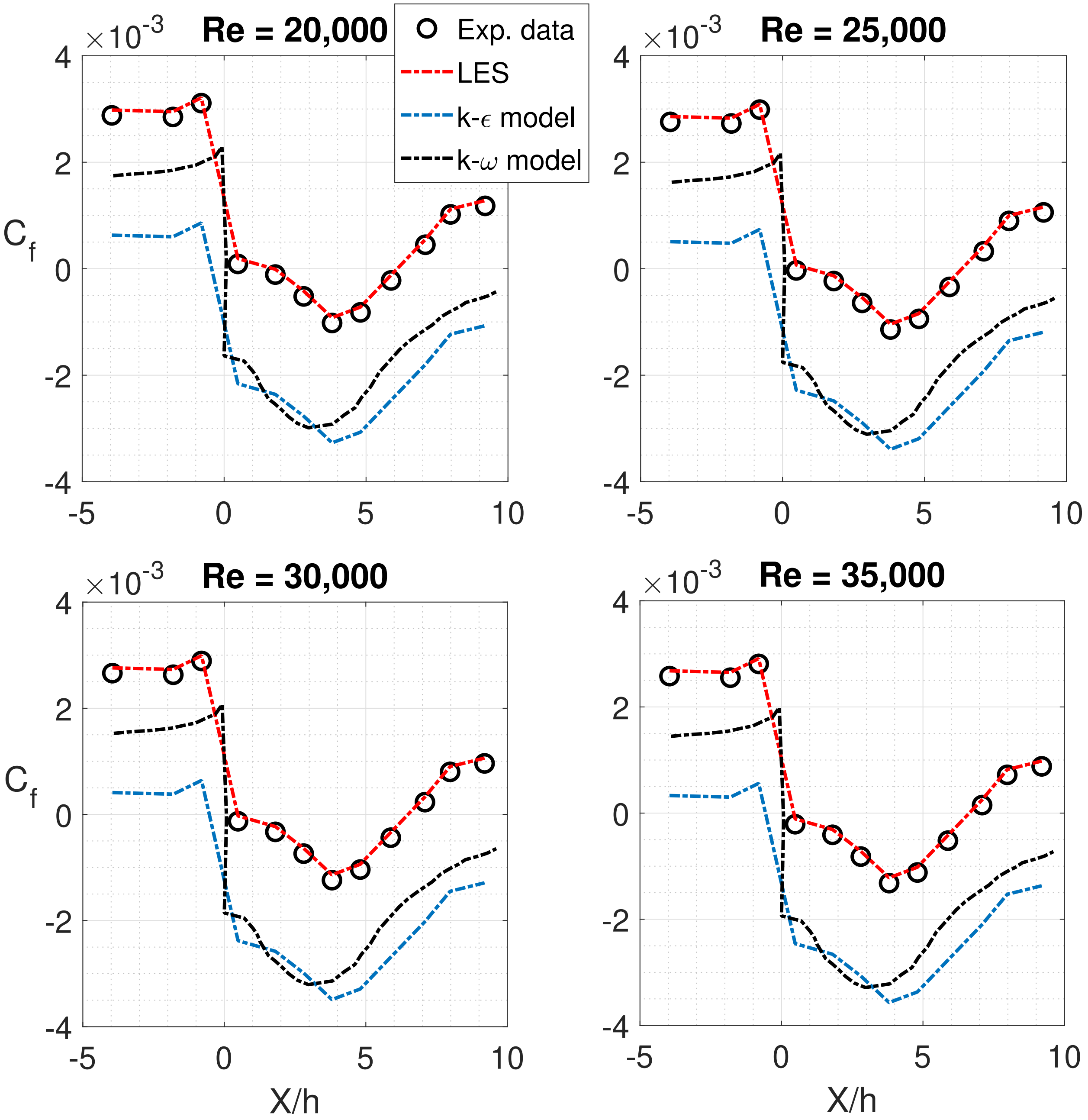


Figure 4.12. Variation of coefficient of skin friction (C_f) for $ER = 2.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Kim et al. [55]).

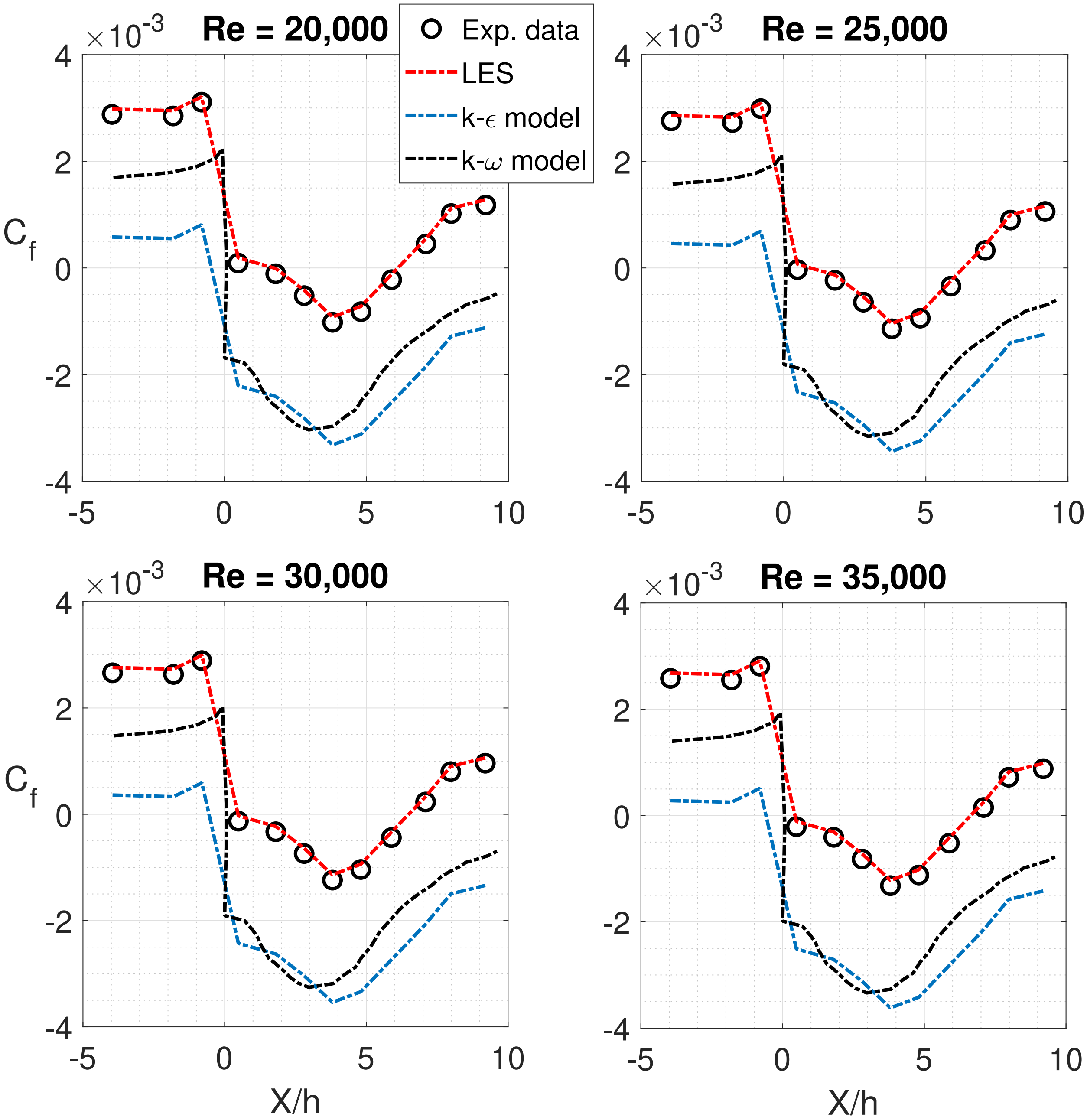


Figure 4.13. Variation of coefficient of skin friction (C_f) for $ER = 3$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Eaton et al. [56]).

The comparison shown above tests the response of a flow system towards change in geometrical flow domain and flow velocity. The studies shown in this section consider two flow features, namely, coefficients of pressure and skin friction, to determine the system's sensitivity towards changes in Reynolds number and expansion ratio. The significance of this parametric study lies in the fact that both coefficients of pressure and skin friction are highly susceptible to boundary conditions, and exploring those changes will help us select the best input parameters for the construction of our machine learning model. Section 4.3 presented below shows the effect of wall functions enabled in the turbulence model and how that can be parameterized to aid our analysis.

4.3 Effect of Wall Functions in RANS Modeling

Boundary layer is a thin region in the vicinity of the wall, where the velocity gradient normal to the wall is very high (as the velocity is zero at the walls and increases to the free-stream value at the end of the boundary layer). Boundary layer formation is a complex hydrodynamic phenomenon, and wall functions are necessary to predict flow behavior near the wall accurately. From a CFD standpoint, it is essential for turbulence models to predict the boundary layer behavior accurately. Ideally, the first grid cell should lie inside the thin viscous sub-layer to capture the boundary layer satisfactorily. Therefore, most models are retrofitted with an enhanced wall function feature for this very reason. This section discusses the effects of forced removal of wall functions from k-ω model, and the results are outlined below.

Figures 4.14 represent the normalized velocity and Reynolds stress plots at different domain locations respectively. As expected, the wall function disabled results show a discrepancy from the k-ω results near the wall. The C_p and C_f plots shown in Figures 4.15 and 4.16 demonstrate a similar trend near the wall, whereas the flow is relatively comparable in the rest of the flow domain.

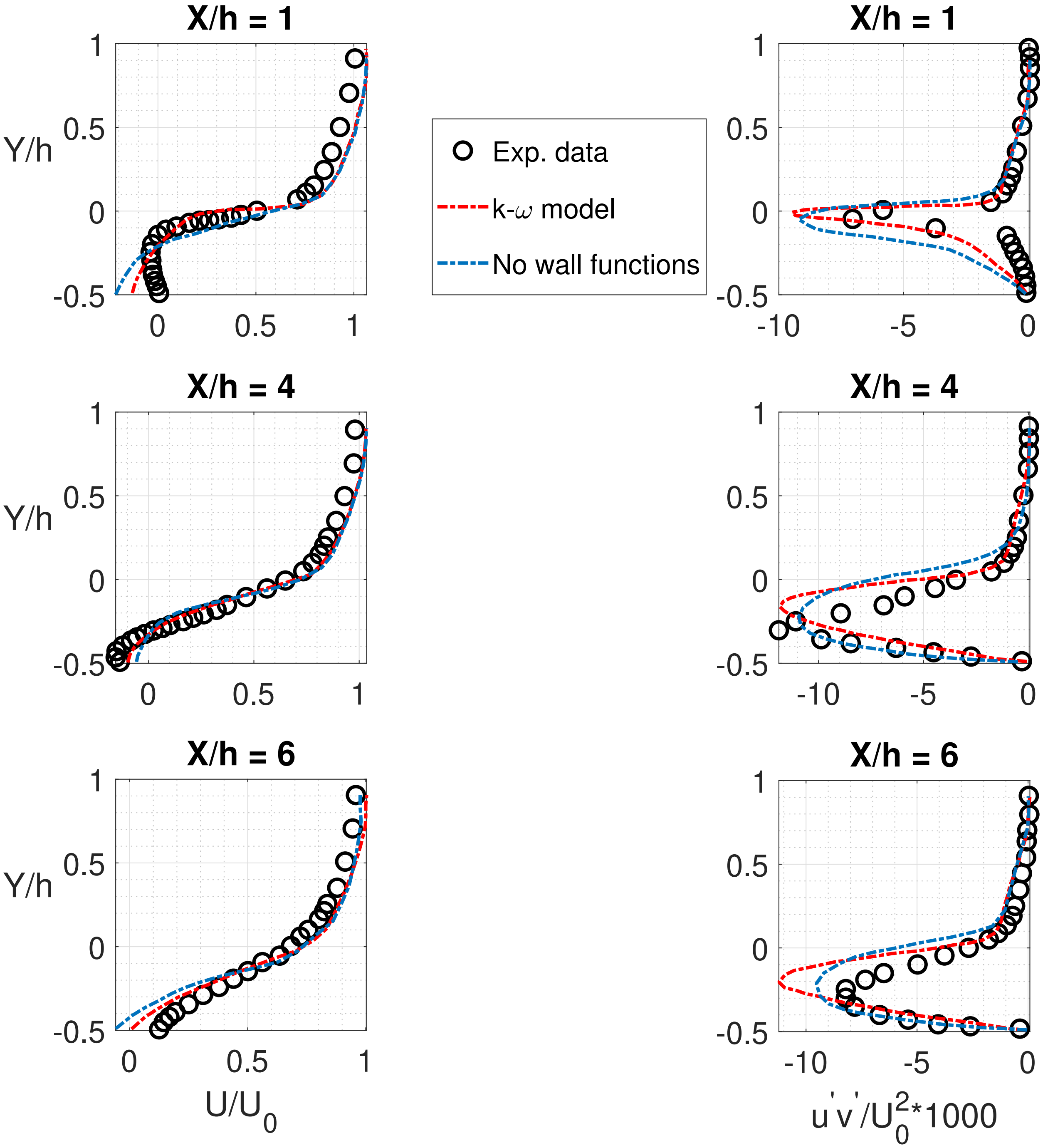


Figure 4.14. Normalized axial velocity and Reynolds stress comparison at different domain locations for $ER = 1.5$ and $Re = 20,000$ (experimental data is obtained from Driver et al. [52]).

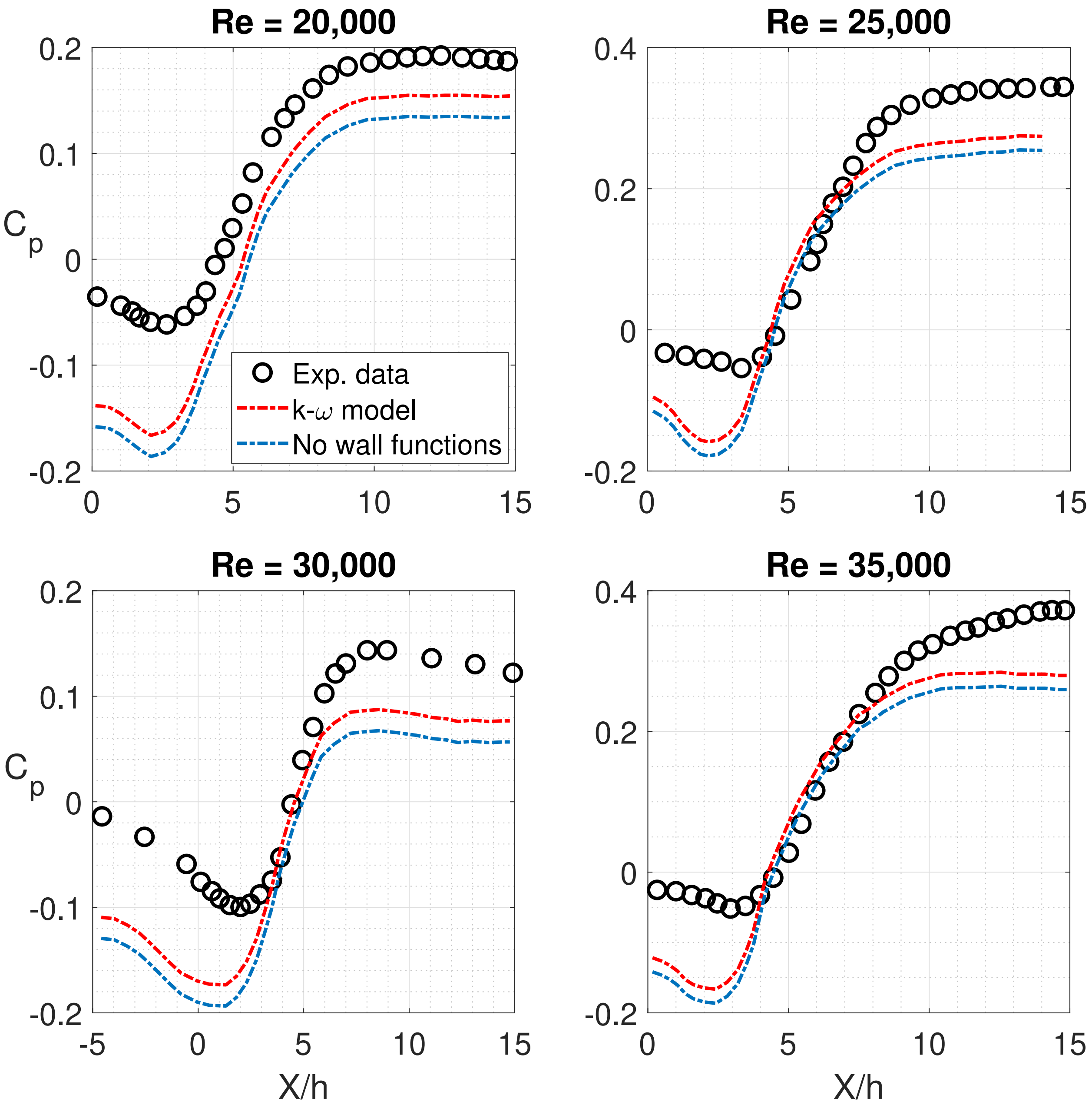


Figure 4.15. Coefficient of pressure (C_p) comparison at different domain locations for $ER = 1.5$ and $Re = 20,000$ (experimental data is obtained from Driver et al. [52]).

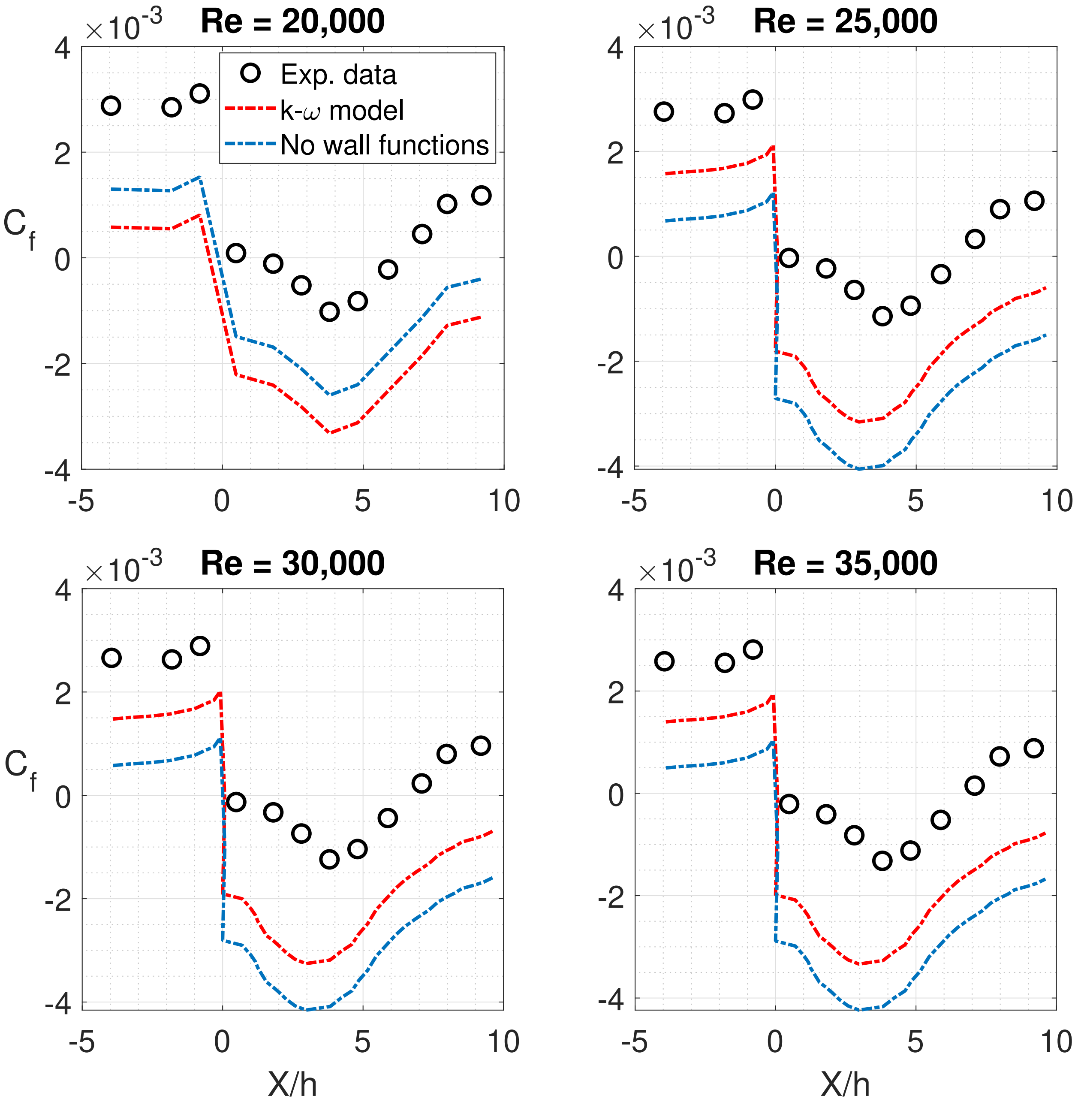


Figure 4.16. Coefficient of skin friction (C_f) comparison at different domain locations for $ER = 1.5$ and $Re = 20,000$ (experimental data is obtained from Driver et al. [52]).

The section presented above discusses the effects of forced removal of wall functions from a RANS-based model, and the results are outlined. The primary goal for doing a wall function analysis is to determine if the reduced order turbulence model should only be developed for the boundary layer near the wall or work for the whole flow domain. The results discussed above do not show a massive discrepancy between wall function enabled and disabled models. Therefore, it is prudent to develop a machine learning model that would improve the flow predictions in the whole flow domain. Section 4.4 shown below discusses the comparison between k-ϵ based results and LES results to establish the discrepancy between both, pointing out the limitation of RANS-based results, which can be overcome by developing a reduced order turbulence model using machine learning.

4.4 Analysis of k-ϵ Model Using LES Data

The parametric studies conducted in the Sections 4.1.2, 4.2, and 4.3 already highlight the discrepancy between RANS based results and experimental data for parameters such as the re-circulation length (X_r), the coefficient of pressure (C_p), and the coefficient of skin friction (C_f). The k-ϵ model under-predicts the normalized re-circulation length (X_r/S) by 15-20% for the finest grid simulations. The C_p and C_f plots show a deviation of 5-10% from experimental plots at the extreme points. This systematic discrepancy (consistent across the range of Reynolds number and expansion ratios used for the analysis) observed for RANS based models stems from either of two factors: numerical inaccuracies arising due to insufficient discretization, or, discrepancy in the prediction of flow features, stemming from inaccurate calculations of Reynolds stresses.

The LES model proves to be a handy substitute to RANS modeling, as evidenced in figures 4.17, 4.18, and 4.19. The LES results show little to no deviation from the experimental values as shown in the X_r, C_p, and C_f plots below. Figure 4.17 shows the normalized re-circulation length ($X_r/$) for $ER \in \{1.5, 2, 2.5, 3\}$ and $R\text{e} = 20,000$. The plots show a good agreement of LES results with experimental data and a visible under-prediction in case of k-ϵ model results. The error percentage is estimated to be around 15-20%. Figures 4.18

and 4.19 show the lower wall coefficient of skin friction (C_p) and coefficient of skin friction (C_f) respectively. The lower wall results are chosen due to all complicated flows mechanisms, such as the re-circulation zone forming near the step. As established in the previous sections, the plots show a good agreement with the LES plots and a systematic difference from the RANS-based models.

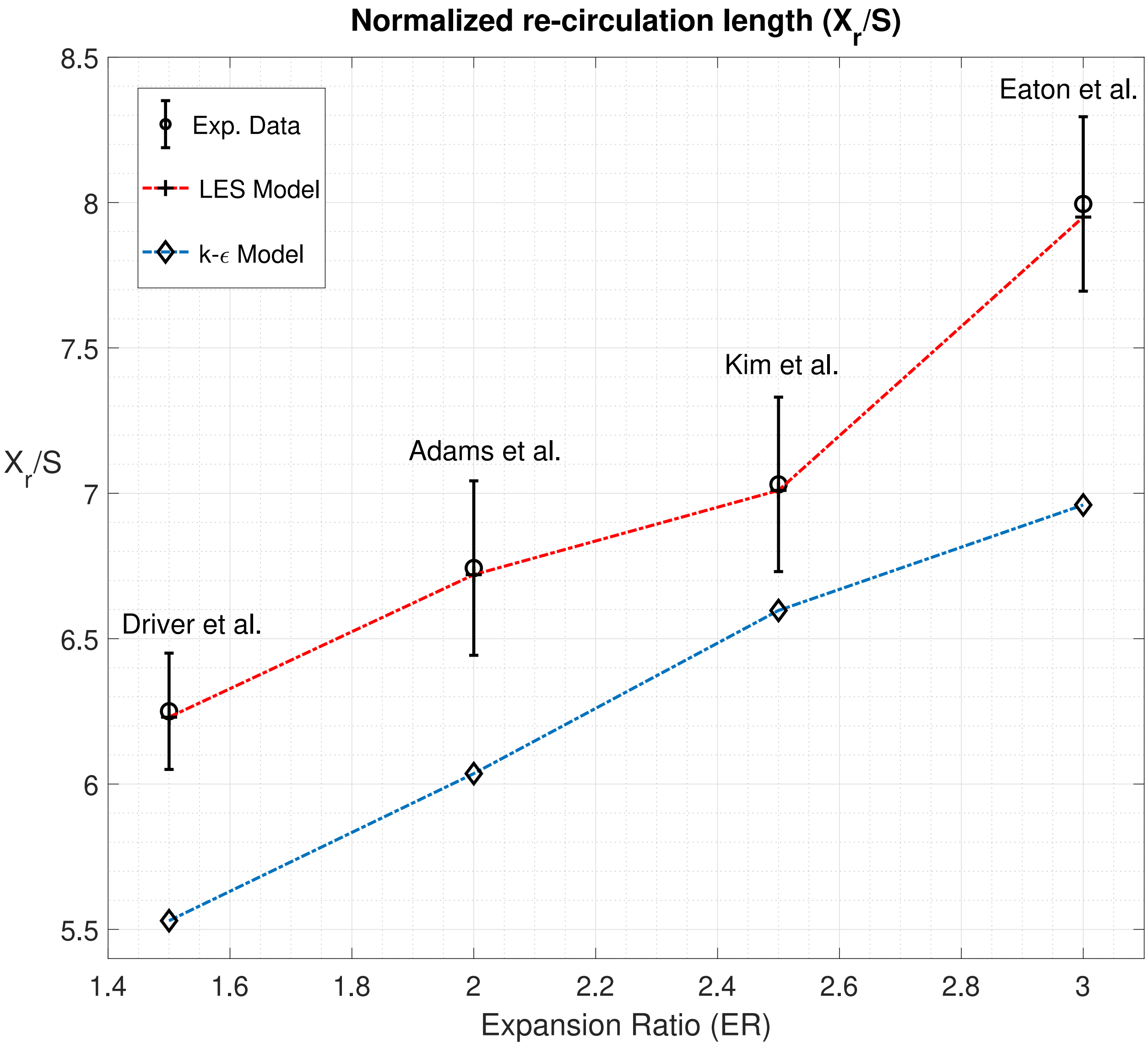


Figure 4.17. Comparison of Experimental, LES, and RANS normalized re-circulation length ($X_r/$) for $ER \in \{1.5, 2, 2.5, 3\}$ and $Re = 20,000$ (experimental data is obtained from Driver et al. [52], Adams et al. [53], Kim et al. [55], and Eaton et al. [56]).

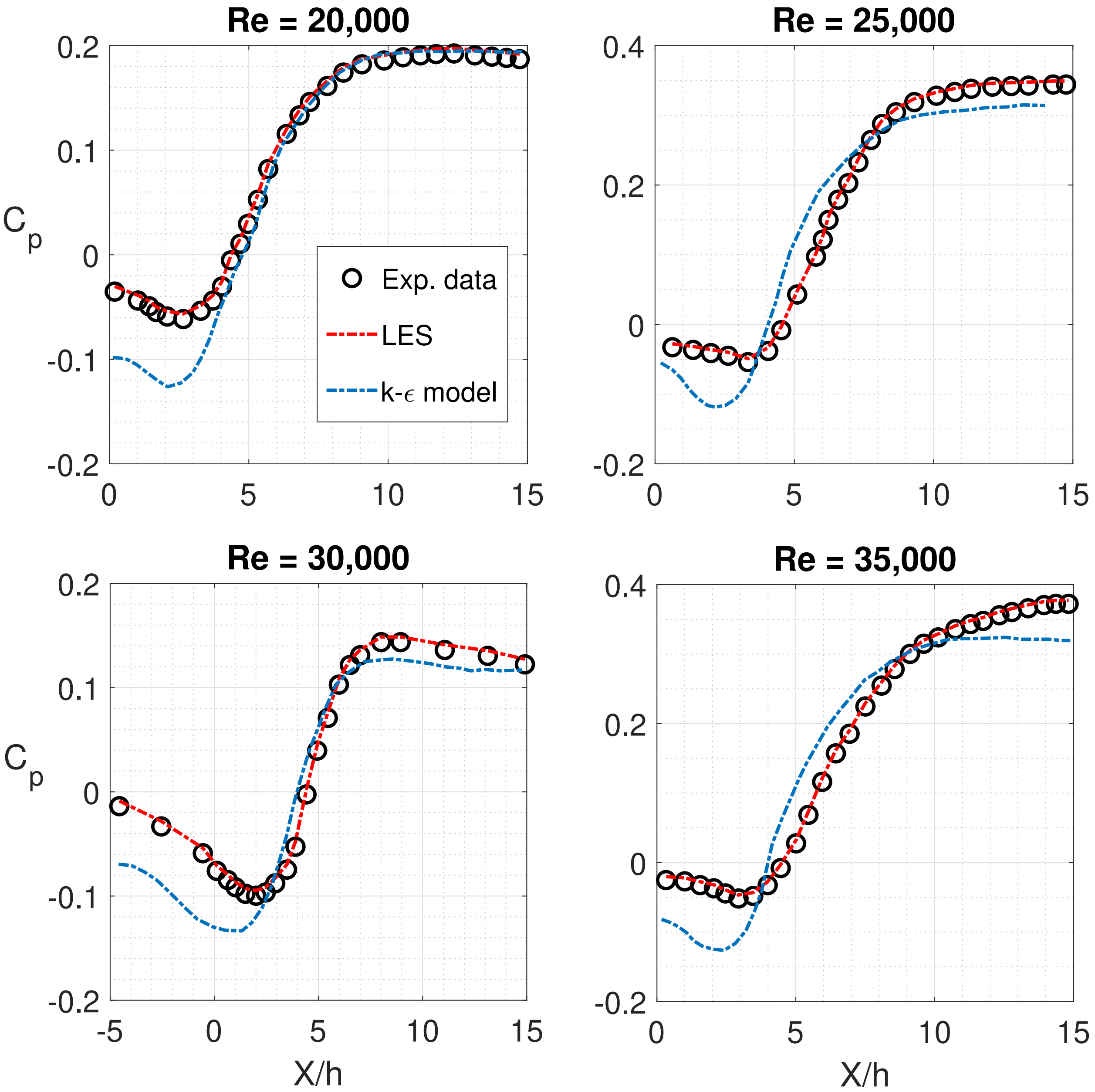


Figure 4.18. Comparison of Experimental, LES, and RANS coefficient of pressure (C_p) for $ER = 1.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Driver et al. [52]).

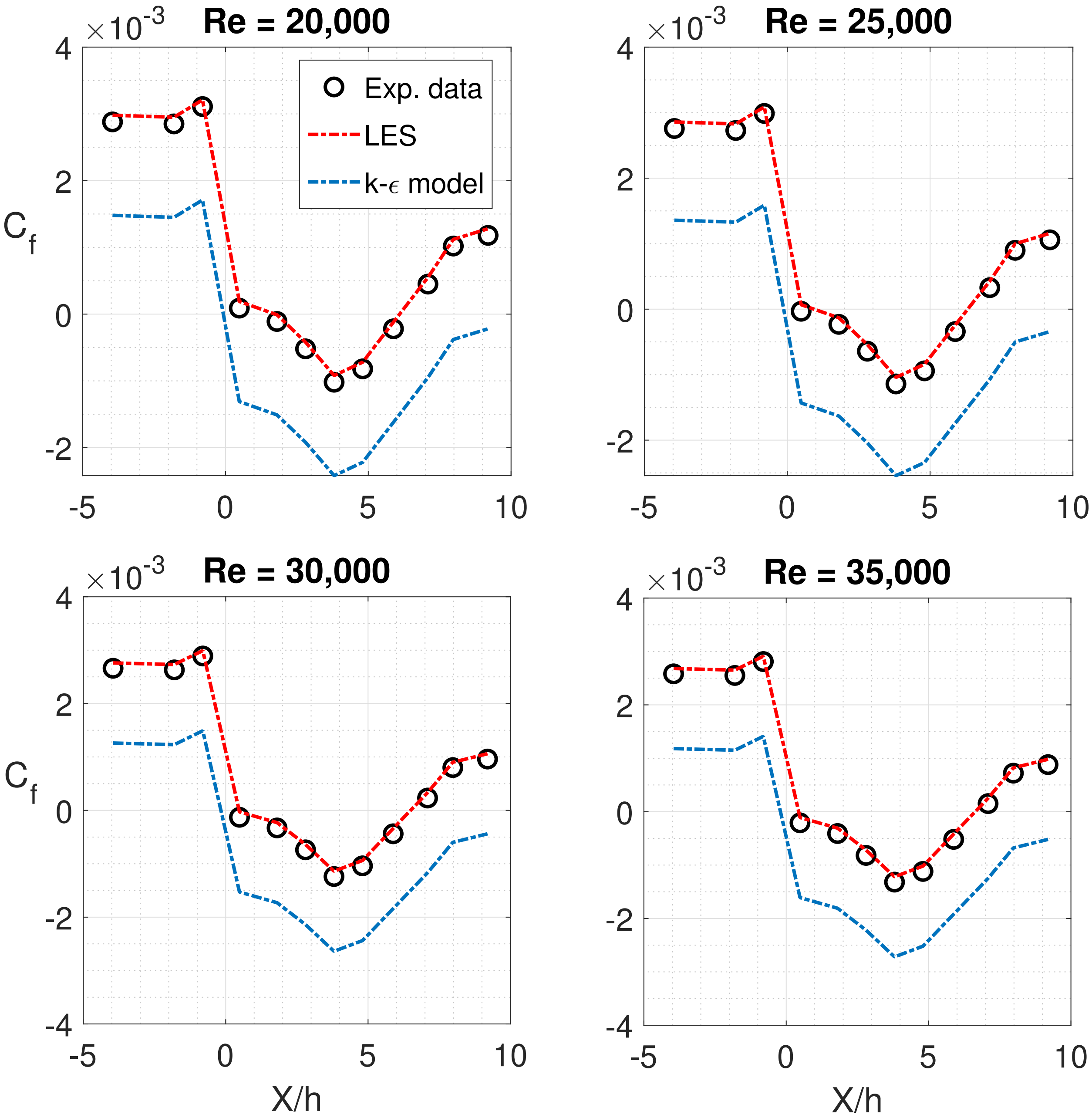


Figure 4.19. Comparison of Experimental, LES, and RANS coefficient of skin friction (C_f) for $ER = 1.5$ and $Re \in \{2, 2.5, 3, 3.5\} \times 10^4$ (experimental data is obtained from Driver et al. [52]).

4.5 Limitations of k-ϵ Model

The reason behind presenting this section shows the motivation behind the current work, in simple words, "*Systematically improving RANS results using machine learning.*" The analysis shown in the previous sections present a comprehensive view of highlighting the pros and con of using RANS-based models, and the limitations it poses in the current scenario. The parametric studies conducted in the Sections 4.1.2, 4.2, and 4.3 reflect the importance of changes to flow factors such as Reynolds number, geometrical factors such as expansion ratios, and wall functions or instead how the absence of it can cause a radical change in flow properties.

The following section outlines the use of machine learning analysis to improve the RANS results shown in the previous sections systematically. Section 5.1 highlights the mathematical procedures followed to obtain RANS-based parameters from LES data to be used as a training data set in the ML model. The original formulae for the extrinsic flow quantities such as turbulent kinetic energy, production ($\mathcal{P}$), strain rate tensor (S_{ij}), and eddy viscosity (μ_t) are obtained from Pope [21]. Section 5.2 depicts the working of the actual machine learning model and the parameters involved in the calculations. Although, the analysis is at a nascent stage, the results show promise of an independent eddy-viscosity ML model capable of predicting results at par with the LES model.

5. EXPLORATION OF MACHINE LEARNING FOR TURBULENCE MODEL DEVELOPMENT

This chapter discusses the construction and application of the machine learning model in the present work. Section 5.1 outlines the mathematical methods for training data extraction from the LES modeling. Section 5.2 shows the comparison of the results for the reduced order turbulence model using machine learning. The subsequent sections consist of a further zonal analysis to determine the working of the machine learning model at different locations of the flow domain.

5.1 Extraction of Training Data from LES Model

The machine learning analysis requires the model usable form of turbulent kinetic energy (k) and dissipation rate (ϵ), which have to be derived from LES data as they are not readily available. The subsections are shown below chalk out the mathematical formulation of the machine form usable training data from the LES model.

As we have established in the previous sections:

- RANS Decomposition: $\phi = \langle\phi\rangle + \phi$.
- LES filtering: $\phi = \tilde{\phi} + \phi''$.

5.1.1 Turbulent Kinetic Energy (k)

The original equation for turbulent kinetic energy is shown below [21]:

$$k_{tot} = \frac{1}{2}\langle {u'_{\mathrm{i}}}^2 \rangle = \frac{1}{2}\langle (u_{\mathrm{i}} - \langle u_{\mathrm{i}} \rangle)^2 \rangle. \tag{5.1}$$

To express turbulent kinetic energy in terms of LES quantities we have to decompose the velocity into LES filtered and sub-grid scale quantities:

$$k_{tot} = \frac{1}{2}\langle (\tilde{u}_{\mathrm{i}} + u''_{\mathrm{i}} - \langle \tilde{u}_{\mathrm{i}} \rangle)^2 \rangle. \tag{5.2}$$

Assumptions:

- The Reynolds average of sub-grid scale quantities are assumed to be zero for simplicity, i.e. $\langle u''_{\mathrm{i}} \rangle = 0$.
- The total TKE is obtained as a sum of filtered TKE and sub-grid scale TKE, i.e. $k_{tot} = k_{filtered} + k_{sgs}$.

After simplifying, the $k_{filtered}$ is obtained as shown below:

$$k_{filtered} = \frac{1}{2}[\langle \tilde{u}_{\mathrm{i}}\tilde{u}_{\mathrm{i}} \rangle - \langle \tilde{u}_{\mathrm{i}} \rangle \langle \tilde{u}_{\mathrm{i}} \rangle], \tag{5.3}$$

where the k_{sgs} is obtained from the solving its transport equation inherently in LES modeling using Fluent 19.0.

5.1.2 Production ($\mathcal{P}$)

Obtaining the dissipation rate data from LES quantities poses a problem as major contribution of the dissipation comes from the smaller scales, which when ignored would not give an accurate estimate of the calculated data. Therefore it is prudent to equate the production term with the dissipation term in the energy budget and estimate the production instead. The original formulation of production is shown below [21]:

$$P \equiv \epsilon = -\langle u'_{\mathrm{i}} u'_{\mathrm{j}} \rangle \frac{\partial \langle u_{\mathrm{i}} \rangle}{\partial x_{\mathrm{j}}} = (\langle u_{\mathrm{i}} \rangle \langle u_{\mathrm{j}} \rangle - \langle u_{\mathrm{i}} u_{\mathrm{j}} \rangle) \frac{\partial \langle u_{\mathrm{i}} \rangle}{\partial x_{\mathrm{j}}}. \tag{5.4}$$

After expanding and simplifying the relevant terms we get the final form,

$$P \equiv \epsilon = (\langle \tilde{u}_{\mathrm{i}} \rangle \langle \tilde{u}_{\mathrm{j}} \rangle - \langle \tilde{u}_{\mathrm{i}} \tilde{u}_{\mathrm{j}} \rangle) \frac{\partial \langle \tilde{u}_{\mathrm{i}} \rangle}{\partial x_{\mathrm{j}}}. \tag{5.5}$$

5.1.3 Strain rate tensor (S_{ij})

Using the original form of strain rate tensor, S_{ij}, as shown in Pope's book [21]:

$$\overline{S_{\mathrm{ij}}} = \frac{1}{2}\left[\frac{\partial \langle u_{\mathrm{i}} \rangle}{\partial x_{\mathrm{j}}} + \frac{\partial \langle u_{\mathrm{j}} \rangle}{\partial x_{\mathrm{i}}}\right], \tag{5.6}$$

$$\overline{S_{\mathrm{ij}}} = \frac{1}{2}\left[\frac{\partial\langle\tilde{u}_{\mathrm{i}} + u''_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} + \frac{\partial\langle\tilde{u}_{\mathrm{j}} + u''_{\mathrm{j}}\rangle}{\partial x_{\mathrm{i}}}\right]. \tag{5.7}$$

Expanding and simplifying using the assumptions stated in Section 5.1.1,

$$\overline{S_{\mathrm{ij}}} = \frac{1}{2}\left[\frac{\partial\langle\tilde{u}_{\mathrm{i}}\rangle}{\partial x_{\mathrm{j}}} + \frac{\partial\langle\tilde{u}_{\mathrm{j}}\rangle}{\partial x_{\mathrm{i}}}\right]. \tag{5.8}$$

5.1.4 Eddy viscosity (μ_t)

Using the original formula for eddy viscosity from Pope's book [21]:

$$\mu_t = \frac{\rho\left[\frac{1}{3}k_t\delta_{\mathrm{ij}} - 0.5(\langle u'_{\mathrm{i}}u'_{\mathrm{j}}\rangle)\right]}{\overline{S_{\mathrm{ij}}}}. \tag{5.9}$$

Modeling the Reynolds stress term as in the production term above:

$$\mu_t = \frac{\rho\left[\frac{1}{3}k_t\delta_{\mathrm{ij}} - 0.5(\langle\tilde{u}_{\mathrm{i}}\tilde{u}_{\mathrm{j}}\rangle + \langle u''_{\mathrm{i}}\tilde{u}_{\mathrm{j}}\rangle + \langle\tilde{u}_{\mathrm{i}}u''_{\mathrm{j}}\rangle + \langle u''_{\mathrm{i}}u''_{\mathrm{j}}\rangle - \langle\tilde{u}_{\mathrm{i}}\rangle\langle\tilde{u}_{\mathrm{j}}\rangle)\right]}{\overline{S_{\mathrm{ij}}}}. \tag{5.10}$$

Final form of the equation is shown below:

$$\mu_t = \frac{\rho\left[\frac{1}{3}k_t\delta_{\mathrm{ij}} - 0.5(\langle\tilde{u}_{\mathrm{i}}\tilde{u}_{\mathrm{j}}\rangle + \langle u''_{\mathrm{i}}u''_{\mathrm{j}}\rangle - \langle\tilde{u}_{\mathrm{i}}\rangle\langle\tilde{u}_{\mathrm{j}}\rangle)\right]}{\overline{S_{\mathrm{ij}}}}. \tag{5.11}$$

5.2 Reduced Order Turbulence Model Using Machine Learning

As mentioned previously, Random Forest (RF) model is used in the machine learning analysis in this section. Random forest is a class of decision tree algorithm ideally suited for handling large data segments without compromising statistical efficiency. The random forest algorithm was devised originally by Breiman [33] in 2001, based on the earlier contributions of [35] [36] [37]. The basic principle of the algorithm is based on the following simple steps:

- Divide the large chunk of raw data into smaller sample sizes.

- Develop a randomized tree predictor for each sample size.

- Aggregate the tree predictors together.

The validity of the machine learning model was tested using previously obtained k-ϵ eddy viscosity data. As previously stated, the eddy viscosity relation in the k-ϵ model is given as:

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}, \tag{5.12}$$

where $C_\mu = 0.09$ is a model constant. Figure 5.1 shows the training of the Random Forest ML model using previously obtained eddy viscosity data. As evidenced below, the predicted value of eddy viscosity shows a perfect match with the actual value in the training dataset. The value of the coefficient of determination, $R^2 \sim 1$, for the training plot which implies a pristine training algorithm. The figure also shows the testing of ML algorithm on unseen data, which shows good agreement for the most part proving the validity of our model.

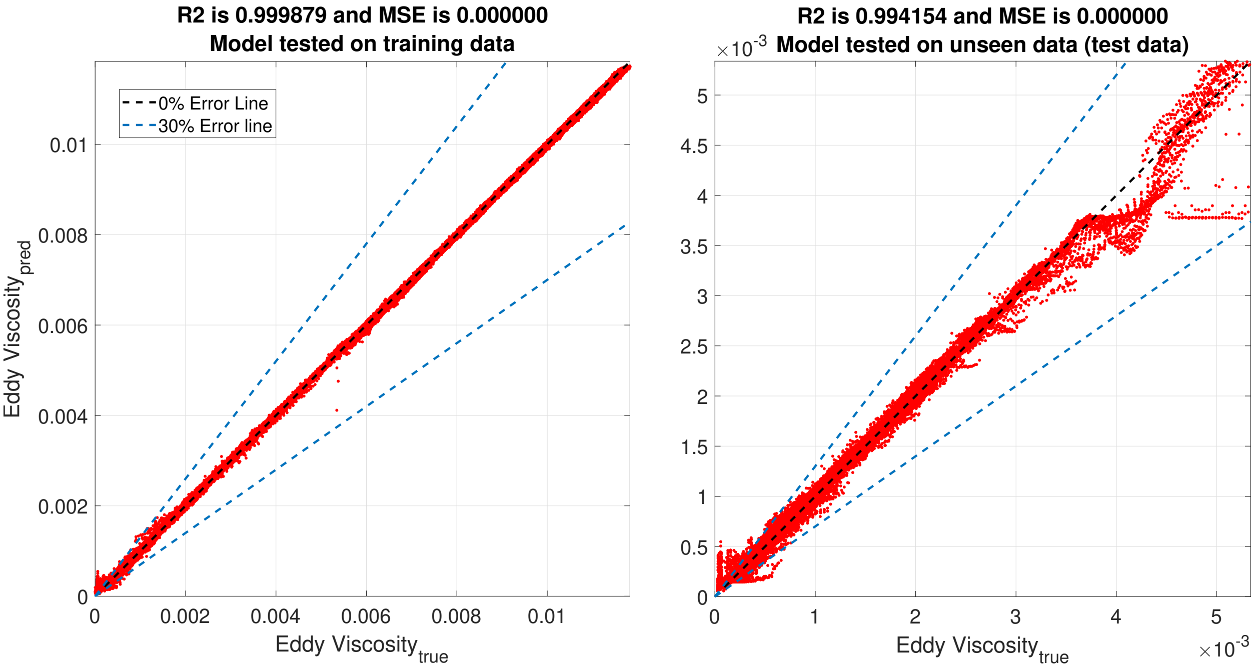


Figure 5.1. Training and testing of the machine learning model using training data obtained from k-ϵ model.

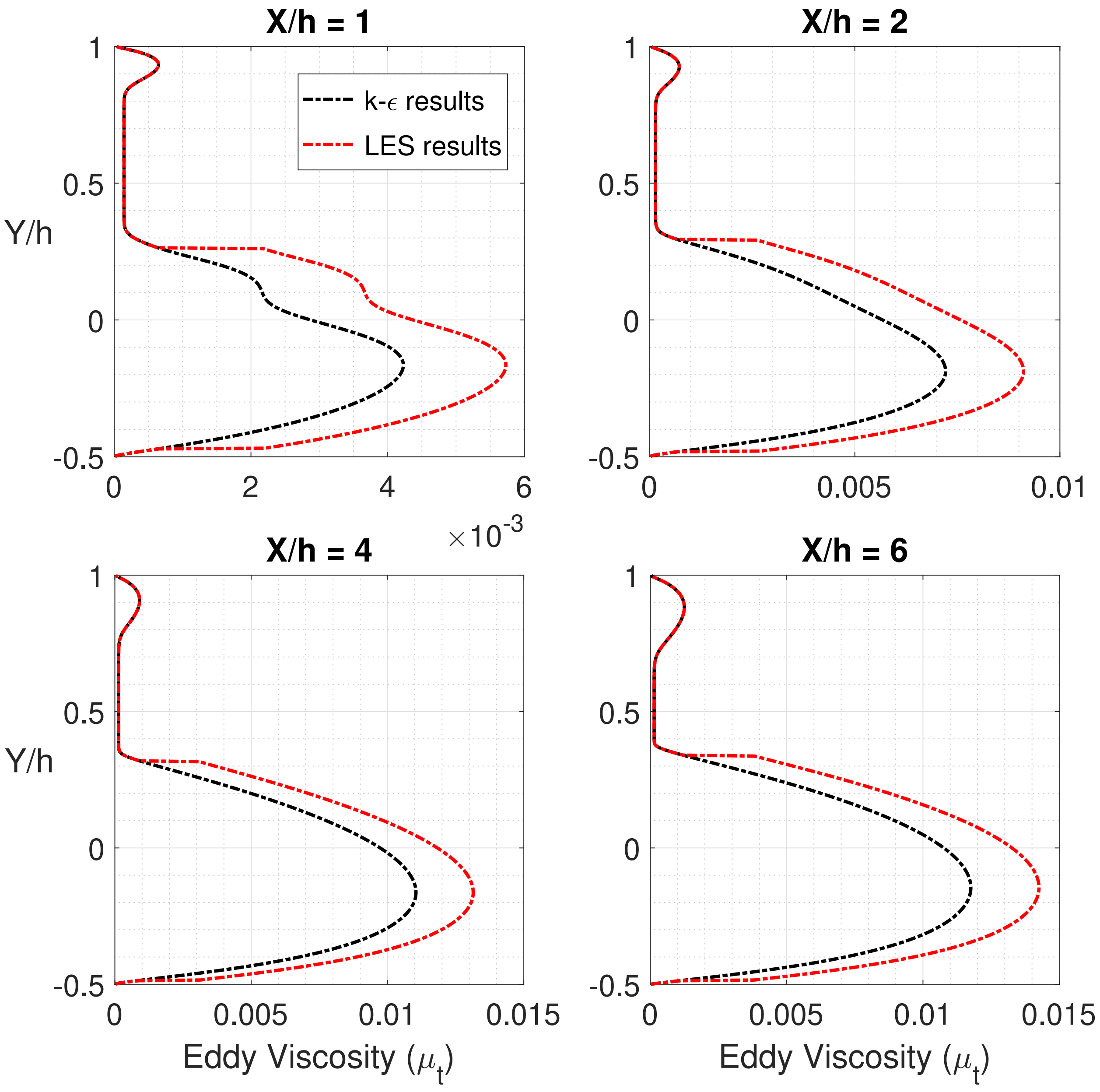


Figure 5.2. Comparison of LES and RANS eddy viscosity (μ_t) for $ER = 1.5$ and $Re = 20,000$ at different domain locations. The locations are chosen in and around the re-circulation zone to study the effect in the most complicated part of the flow domain.

Figure 5.2 shows the eddy viscosity plot comparison for the LES and RANS model at different domain locations, as shown below. As we have established before, the LES model shows close proximity to experimental results in terms of essential flow features. There is a visible discrepancy in the k-ϵ, and LES derived eddy viscosity values as expected. This serves as a motivation for developing the ML model, which essentially acts as an ad hoc k-ϵ model to predict eddy viscosity (μ_t), using turbulent kinetic energy, and dissipation rate (ϵ) as features.

Figure 5.3 shows the training of the ML model using the eddy viscosity data derived from the LES model. The training plot does not show a perfect match as in the case of the eddy viscosity data obtained form the k-ϵ model. This can be attributed to the fact that the sub-grid scale averaging was ignored while calculating the training data using LES variables, leading to imperfect data. A zonal analysis is carried out in this section to further analysis the machine learning results.

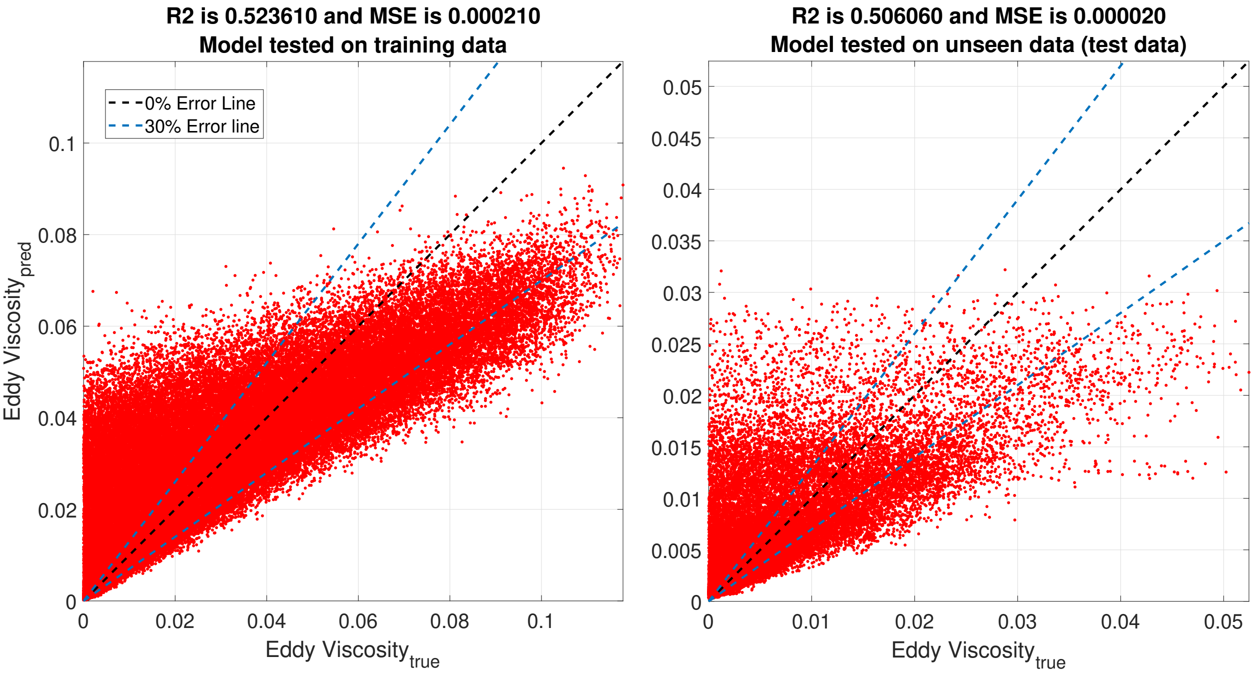


Figure 5.3. Training and testing of the machine learning model using μ_t obtained from LES model for the whole back-step domain.

5.3 Zonal Analysis Using Machine Learning

Figure 5.4 shows the schematic diagram of the zonal division of the back-step flow domain further to understand the implications of the machine learning analysis. The flow domain is divided into the *Inlet*, *Re-circulation*, and *Downstream* zones as shown below. Figures 5.5, 5.6, and 5.7 represent the training and testing of ML model at the inlet, re-circulation, and downstream zone respectively. The zonal analysis indicates a shift in prediction plots as the individual domain plots give better results than the domain taken as a whole. Among the zones, the ML model gives the best match in the re-circulation zone compared to the other two zones. This is a significant conclusion that potentially proves that the ML model gives us a good prediction when the training data from zonal fractions are passed through the model instead of the data obtained from the whole domain. The predictions in the re-circulation zone are encouraging, and further analysis is done in the following sections.

The zonal analysis's purpose lies in the need to make the machine learning model work in specific zones of the flow domain to determine the viability of the reduced order turbulence model. The predictions observed in the zonal analysis will set up a basis for our future work to develop a machine learning model for specific flow zone to obtain the best results.

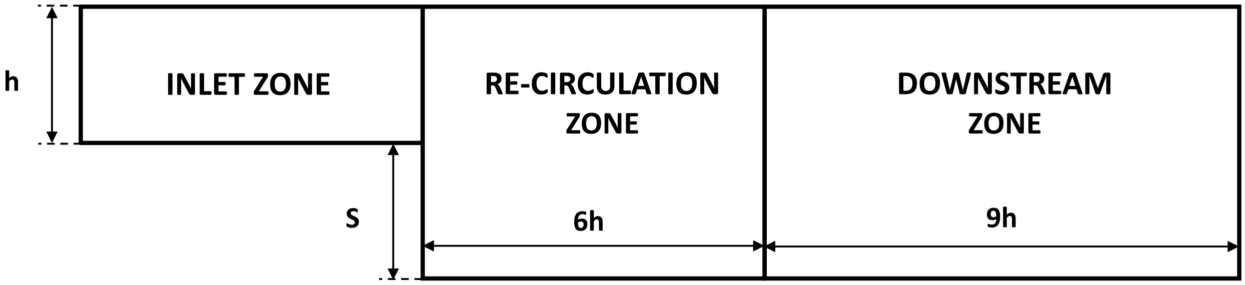


Figure 5.4. Schematic diagram of the zonal division of the back-step domain.

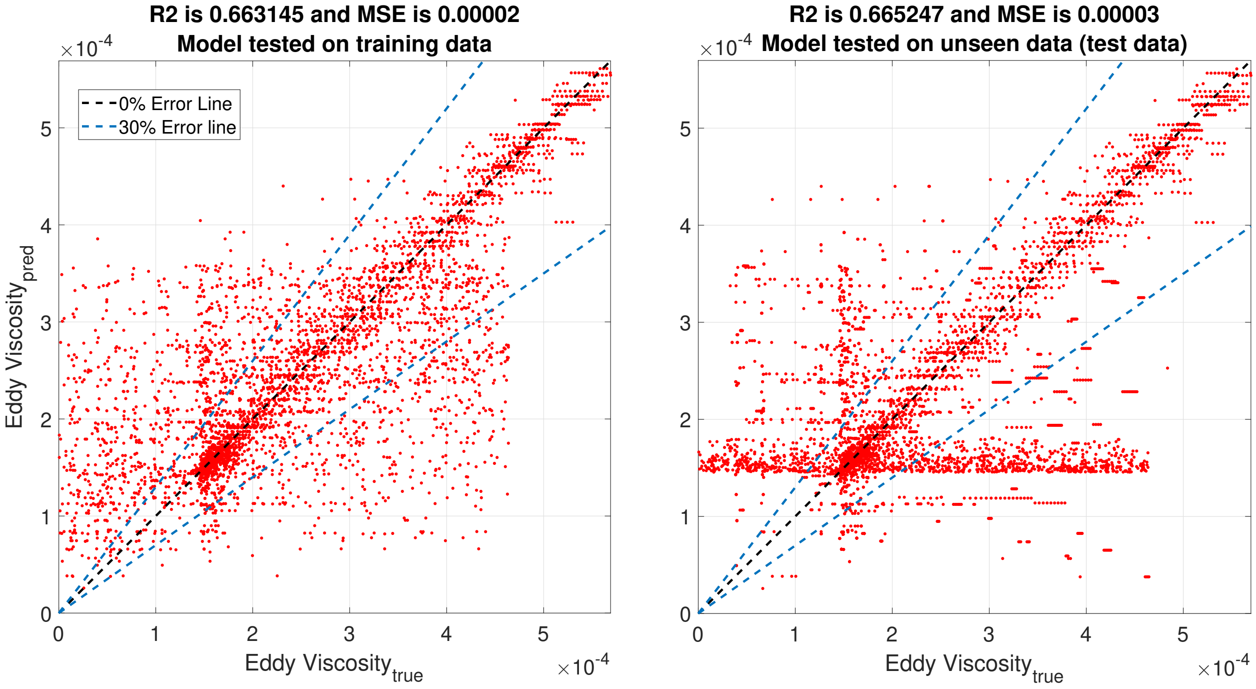


Figure 5.5. Training and testing of the machine learning model using μ_t obtained from LES model for the inlet zone as shown in Figure 5.2.

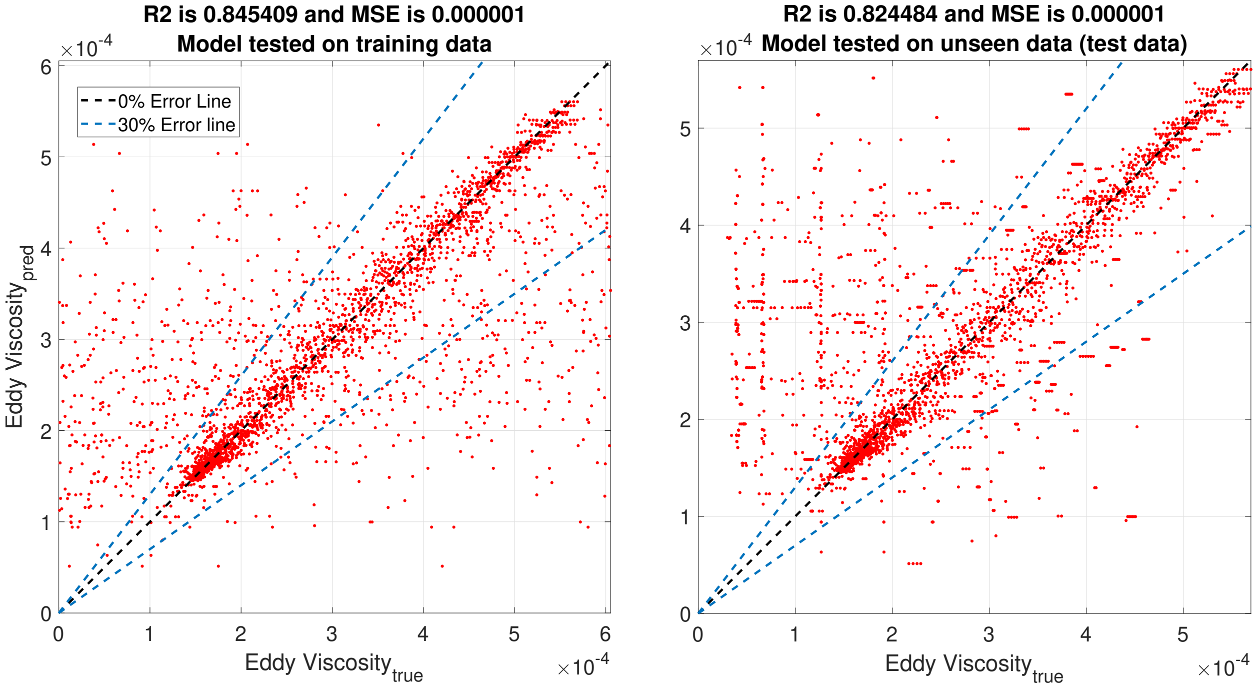


Figure 5.6. Training and testing of the machine learning model using μ_t obtained from LES model the re-circulation zone as shown in Figure 5.2.

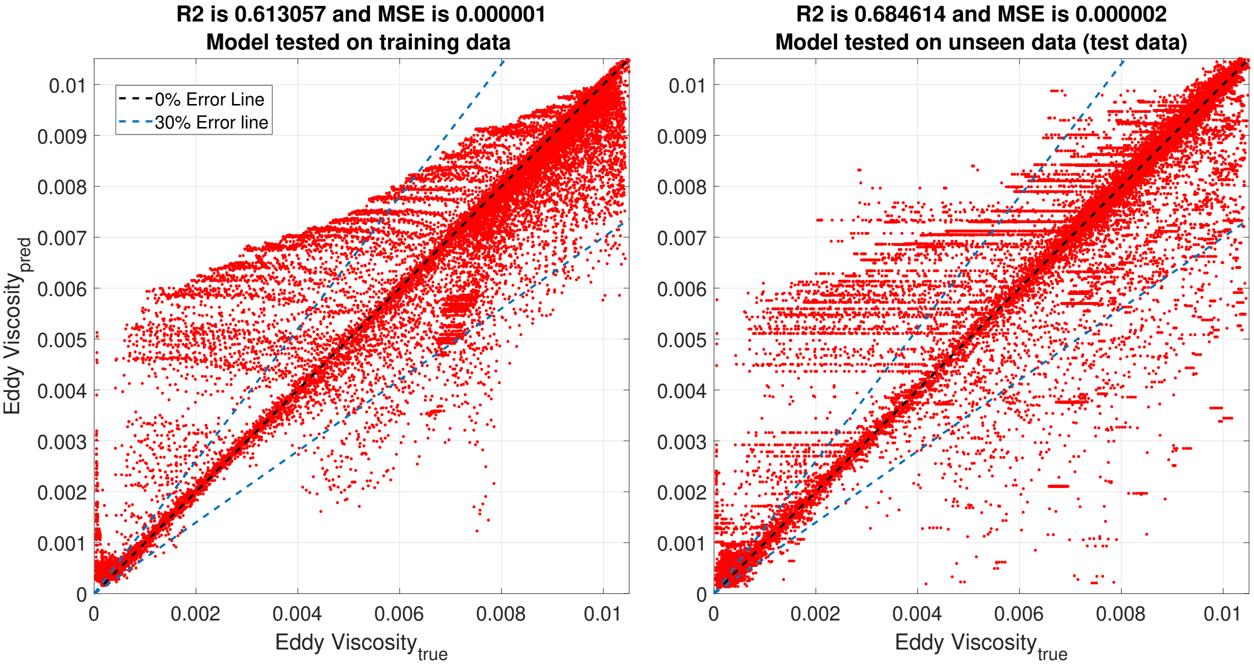


Figure 5.7. Training and testing of the machine learning model using μ_t obtained from LES model for the downstream zone as shown in Figure 5.2.

5.4 Eddy Viscosity (μ_t) Comparison

The aim of this section is to further analysis the potential ramifications of the zonal analysis using machine learning. Figure 5.8 shows the eddy viscosity plot comparison for the LES, RANS, and ML model at different domain locations. The locations are chosen in and around the re-circulation zone to study the effect in the most complicated part of the flow domain. The ML model results, although they show a slight improvement over RANS results, still deviate from the original LES-derived eddy viscosity values.

Figure 5.9 shows the eddy viscosity plot comparison for the LES, RANS, and ML model at different domain locations in and around the downstream zone to study the contrast between the eddy viscosity values in different parts of the flow domain. The ML model still shows intermediate results between the k-ϵ and LES values, but the discrepancy between the ML results and the k-ϵ data is slightly lower than the re-circulation zone.

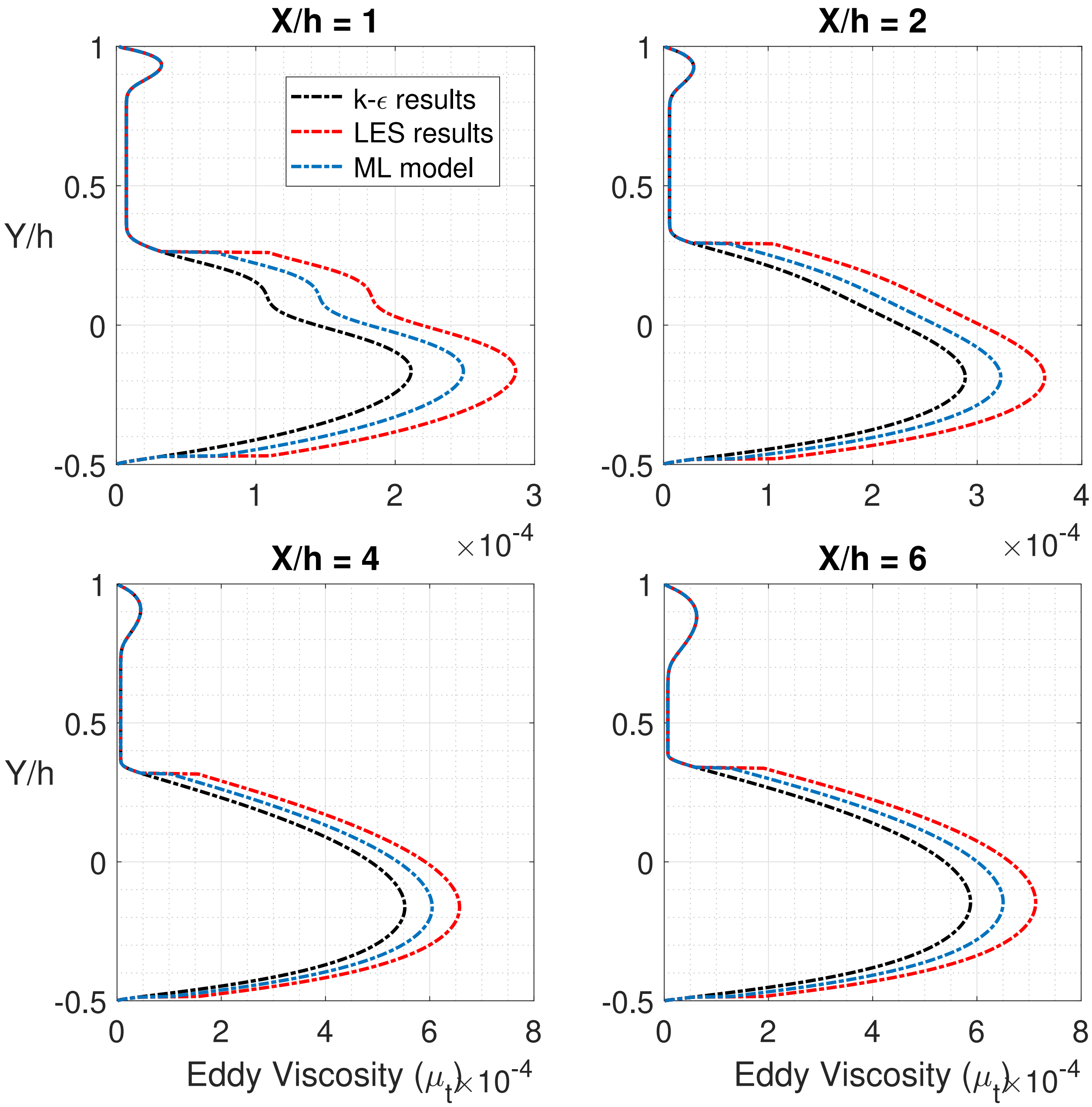


Figure 5.8. Comparison of LES, RANS, and ML model eddy viscosity (μ_t) for $ER = 1.5$ and $Re = 20,000$ at different domain locations. The locations are chosen in and around the re-circulation zone to study the effect in the most complicated part of the flow domain.

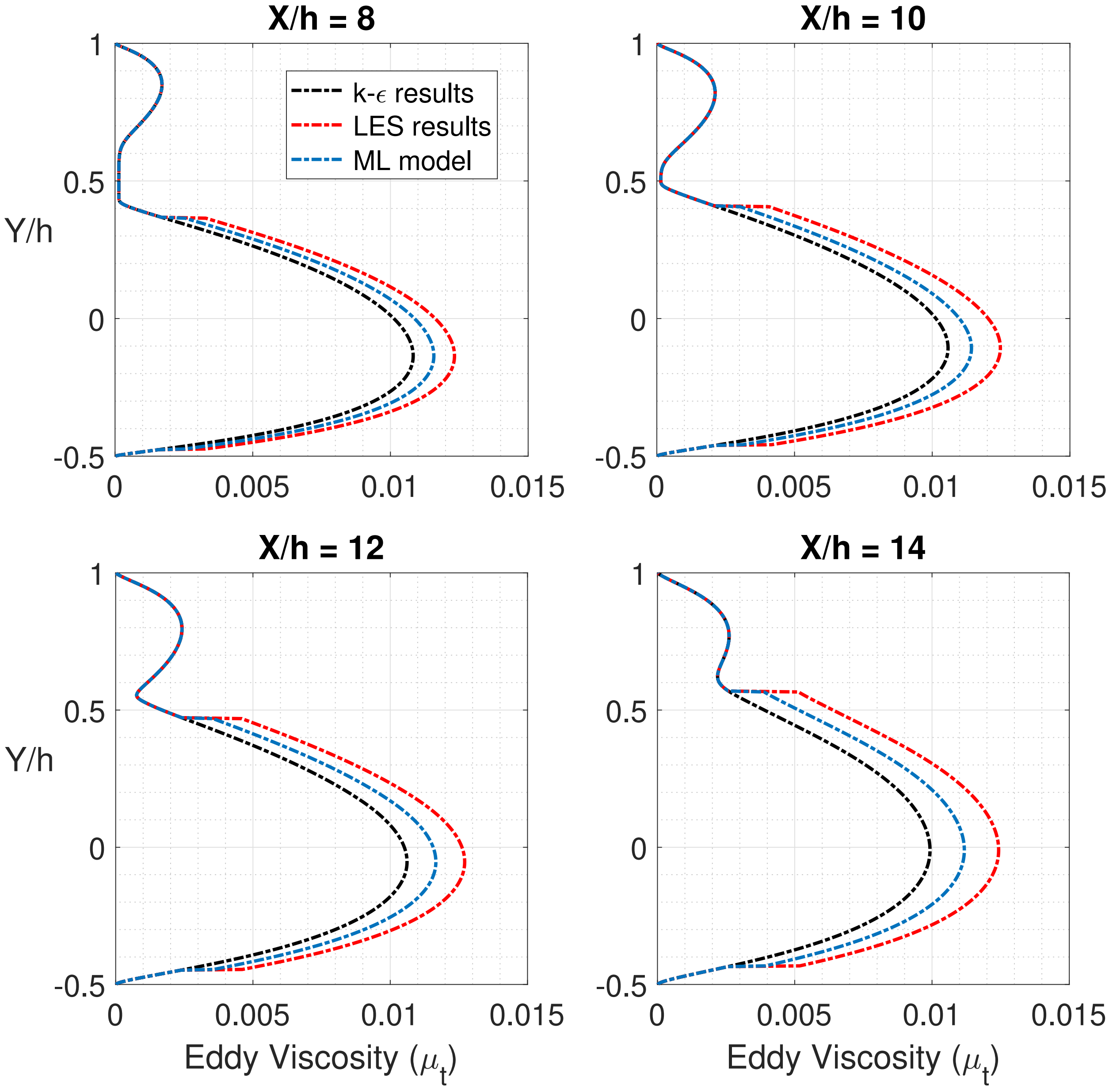


Figure 5.9. Comparison of LES, RANS, and ML model eddy viscosity (μ_t) for $ER = 1.5$ and $Re = 20,000$ at different domain locations. The locations are chosen in and around the downstream part of the flow domain.

5.5 Functional Analysis of the Zonal Data

The functional analysis within the zones attempts to determine if the eddy viscosity is a unique function of the chosen parameters (k and ϵ) within that zone. Figure 5.10 shows the scatter plot of eddy viscosity (represented by the color bar) varying with turbulent kinetic energy (k) and dissipation rate (ϵ). The scatter plots for the whole domain, inlet, and downstream zones are similar, revealing huge point-to-point variation, hence a chaotic scatter plot. It is safe to assume that eddy viscosity can not be expressed as a unique function of k and ϵ. On the other hand, the re-circulation scatter plot shows a smooth transition of color across the domain, implying the derived values of eddy viscosity from LES data can potentially be expressed as a unique function of k and ϵ. This further validates our observations in the previous sections, where the re-circulation zone shows better prediction than the other zones.

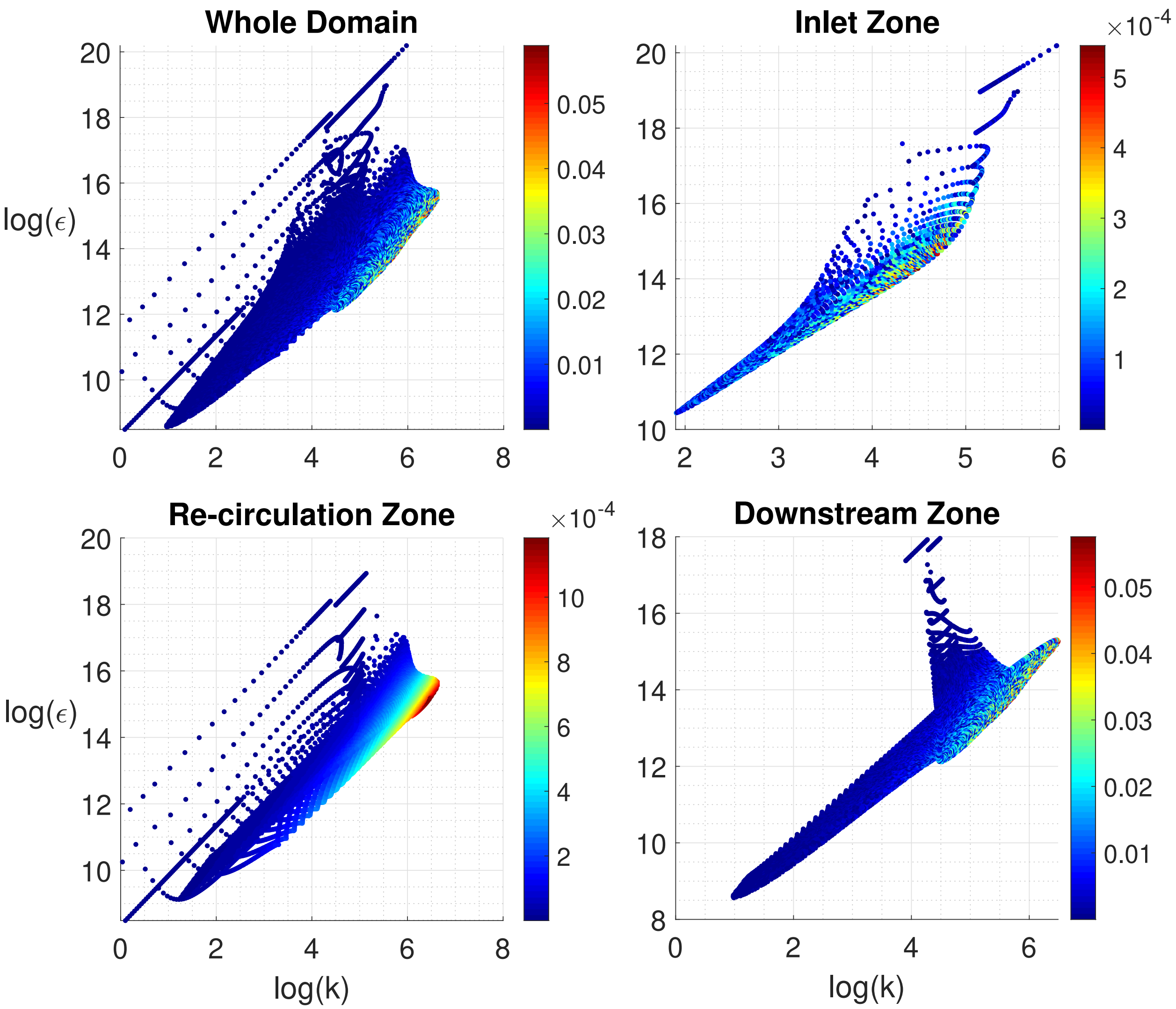


Figure 5.10. Functional analysis presented for the whole domain, inlet, re-circulation, and downstream zones.

6. CONCLUSIONS

This chapter summarizes the principal conclusions drawn from the analysis shown in the above sections and the future challenges presented in the current work. Based on the computational analysis done above, the following conclusions are drawn:

- For the turbulence analysis, the back-step was chosen as the flow domain due to its simplicity and application in several high-profile engineering systems. The problem area was identified as the properties under-prediction from RANS-based analysis, and sustained efforts were made to improve upon those areas.

- Grid convergence analysis was conducted on five different grids as shown in Section 4.1.1 including a multiplicative parameter P to determine the coarseness of grids. A medium grid indexed by $P = 2$ was chosen for RANS and LES analysis followed in Chapters 4 and 5.

- Parametric studies concerning flow effects controlled by Reynolds number, geometric effects controlled by expansion ratio, and wall functions were conducted. The normalized re-circulation length (X_r/S), coefficient of pressure (C_p), and skin friction coefficient (C_f) were the chosen parameters of interest.

- The maximum values of re-circulation length and the coefficient of skin friction show a positive correlation to the increase in Reynolds number, whereas the coefficient of pressure broadens towards the step vicinity with increasing Reynolds number. All parameters show an increase in value with an increase in expansion ratio.

- Chapter 5 introduces the working of the Random Forest model as a tool for the machine learning analysis in the present work. The training data for the machine learning model was derived from the LES variables using the mathematical procedures as shown in Section 5.1.

- The ML model was validated using the eddy viscosity data derived from the k-ϵ model to ensure reliable results. The training and testing of training data derived from LES variables do not show a good match between the predicted and true values of eddy viscosity.

- A zonal analysis was done by dividing the flow into 3 zones i.e. *Inlet*, *Re-circulation*, and *Downstream* zones. The ML model was applied to each zone individually to check its viability. The zonal analysis reveals that the ML model gives better results for each zone than the whole flow domain. Further analysis reveals that the re-circulation zone shows the best prediction of eddy viscosity compared to the other two.
- The significance of this zonal analysis lies in the fact that it enables us to make a workable reduced order turbulence model for specific zones of the flow domain. It has enabled us to predict RANS results better using a new machine learning framework aided by high fidelity LEs data.

6.1 Future Work

Due to time limitations, only a preliminary ML model was constructed. The prospective future scope of the current work is discussed below:

- The turbulence analysis scope can be widened to include more complicated flow domains which will provide a chance chance to develop a truly independent machine learning model.
- Some passive control methods can be employed in the RANS-based analysis to improve flow characteristics without the use of machine learning.
- The machine learning model can be extended to include several other input and output parameters to present a more comprehensive analysis. The ML model shown in the current works presents a preliminary version to which extra parameters can be added with ease.

REFERENCES

[1] P. R. Spalart, “Comments on the feasibility of les for wings, and on a hybrid rans/les approach,” in *Proceedings of first AFOSR international conference on DNS/LES*, Greyden Press, 1997.

[2] R. B. Langtry, “A correlation-based transition model using local variables for unstructured parallelized cfd codes,” 2006.

[3] P. A. Durbin, “Near-wall turbulence closure modeling without damping functions,” *Theoretical and computational fluid dynamics*, vol. 3, no. 1, pp. 1–13, 1991.

[4] K. Duraisamy and G. Iaccarino, “Curvature correction and application of the v2-f turbulence model to tip vortex flows,” *Center for Turbulence Research Annual Research Briefs*, pp. 157–168, 2005.

[5] M. L. Shur, M. K. Strelets, A. K. Travin, and P. R. Spalart, “Turbulence modeling in rotating and curved channels: Assessing the spalart-shur correction,” *AIAA journal*, vol. 38, no. 5, pp. 784–792, 2000.

[6] P. E. Smirnov and F. R. Menter, “Sensitization of the sst turbulence model to rotation and curvature by applying the spalart–shur correction term,” *Journal of turbomachinery*, vol. 131, no. 4, 2009.

[7] E. Decenciere, G. Cazuguel, X. Zhang, G. Thibault, J.-C. Klein, F. Meyer, B. Marcotegui, G. Quellec, M. Lamard, R. Danno, *et al.*, “Teleophta: Machine learning and image processing methods for teleophthalmology,” *Irbm*, vol. 34, no. 2, pp. 196–203, 2013.

[8] Z. Lin, “A methodological review of machine learning in applied linguistics.,” *English Language Teaching*, vol. 14, no. 1, pp. 74–85, 2021.

[9] L. Deng and X. Li, “Machine learning paradigms for speech recognition: An overview,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 1060–1089, 2013.

[10] S. Parneix, D. Laurence, and P. Durbin, “A procedure for using dns databases,” 1998.

[11] H. Raiesi, U. Piomelli, and A. Pollard, “Evaluation of turbulence models using direct numerical and large-eddy simulation data,” *Journal of Fluids Engineering*, vol. 133, no. 2, 2011.

[12] M. Milano and P. Koumoutsakos, “Neural network modeling for near wall turbulent flow,” *Journal of Computational Physics*, vol. 182, no. 1, pp. 1–26, 2002.

[13] S. H. Cheung, T. A. Oliver, E. E. Prudencio, S. Prudhomme, and R. D. Moser, "Bayesian uncertainty analysis with applications to turbulence modeling," *Reliability Engineering & System Safety*, vol. 96, no. 9, pp. 1137–1149, 2011.

[14] W. N. Edeling, P. Cinnella, R. P. Dwight, and H. Bijl, "Bayesian estimates of parameter variability in the k–ε turbulence model," *Journal of Computational Physics*, vol. 258, pp. 73–94, 2014.

[15] H. Kato and S. Obayashi, "Data assimilation for turbulent flows," in *16th AIAA Non-Deterministic Approaches Conference*, 2014, p. 1177.

[16] T. A. Oliver and R. D. Moser, "Bayesian uncertainty quantification applied to rans turbulence models," in *Journal of Physics: Conference Series*, IOP Publishing, vol. 318, 2011, p. 042 032.

[17] E. Dow and Q. Wang, "Uncertainty quantification of structural uncertainties in rans simulations of complex flows," in *20th AIAA Computational Fluid Dynamics Conference*, 2011, p. 3865.

[18] E. Dow and Q. Wang, "Quantification of structural uncertainties in the k-w turbulence model," in *52nd AIAA/ASME/ASCE/AHS/ASC Structures, structural dynamics and materials conference 19th AIAA/ASME/AHS adaptive structures conference 13t*, 2011, p. 1762.

[19] H. Xiao, J.-L. Wu, J.-X. Wang, R. Sun, and C. Roy, "Quantifying and reducing model-form uncertainties in reynolds-averaged navier–stokes simulations: A data-driven, physics-informed bayesian approach," *Journal of Computational Physics*, vol. 324, pp. 115–136, 2016.

[20] J.-X. Wang, J.-L. Wu, and H. Xiao, "Physics-informed machine learning approach for reconstructing reynolds stress modeling discrepancies based on dns data," *Physical Review Fluids*, vol. 2, no. 3, p. 034 603, 2017.

[21] S. B. Pope, *Turbulent flows*, 2001.

[22] J. Weatheritt, "The development of data driven approaches to further turbulence closures," PhD thesis, University of Southampton, 2015.

[23] R. Matai and P. Durbin, "Large-eddy simulation of turbulent flow over a parametric set of bumps," *Journal of Fluid Mechanics*, vol. 866, pp. 503–525, 2019.

[24] H. Tennekes and J. L. Lumley, *A first course in turbulence.* MIT press, 2018.

[25] A. Kolmogorov, “The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds’ Numbers,” *Akademiia Nauk SSSR Doklady*, vol. 30, pp. 301–305, Jan. 1941.

[26] A. P. Singh, “A framework to improve turbulence models using full-field inversion and machine learning,” PhD thesis, 2018.

[27] J. Eggels, F. Unger, M. Weiss, J. Westerweel, R. J. Adrian, R. Friedrich, and F. Nieuwstadt, “Fully developed turbulent pipe flow: A comparison between direct numerical simulation and experiment,” *Journal of Fluid Mechanics*, vol. 268, pp. 175–210, 1994.

[28] H. Le, P. Moin, and J. Kim, “Direct numerical simulation of turbulent flow over a backward-facing step,” *Journal of fluid mechanics*, vol. 330, pp. 349–374, 1997.

[29] M. Lee and R. D. Moser, “Direct numerical simulation of turbulent channel flow up to,” *Journal of fluid mechanics*, vol. 774, pp. 395–415, 2015.

[30] R. D. Moser, J. Kim, and N. N. Mansour, “Direct numerical simulation of turbulent channel flow up to re τ= 590,” *Physics of fluids*, vol. 11, no. 4, pp. 943–945, 1999.

[31] B. Data, “Bigger digital shadows, and biggest growth in the far east,” *IDC Digital Universe Study, EMC*, 2012. [Online]. Available: http://www.whizpr.be/upload/medialab/21/company/Media_Presentation_2012_DigiUniverseFINAL1.pdf.

[32] J. Ling, R. Jones, and J. Templeton, “Machine learning strategies for systems with invariance properties,” *Journal of Computational Physics*, vol. 318, pp. 22–35, 2016.

[33] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[34] R. Kohavi *et al.*, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *Ijcai*, Montreal, Canada, vol. 14, 1995, pp. 1137–1145.

[35] T. G. Dietterich, “Ensemble methods in machine learning,” in *International workshop on multiple classifier systems*, Springer, 2000, pp. 1–15.

[36] T. K. Ho, “The random subspace method for constructing decision forests,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[37] Y. Amit and D. Geman, “Shape quantization and recognition with randomized trees,” *Neural computation*, vol. 9, no. 7, pp. 1545–1588, 1997.

[38] R. Daz-Uriarte and S. A. De Andres, “Gene selection and classification of microarray data using random forest,” *BMC bioinformatics*, vol. 7, no. 1, pp. 1–13, 2006.

[39] A. M. Prasad, L. R. Iverson, and A. Liaw, “Newer classification and regression tree techniques: Bagging and random forests for ecological prediction,” *Ecosystems*, vol. 9, no. 2, pp. 181–199, 2006.

[40] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, “Random forest: A classification and regression tool for compound classification and qsar modeling,” *Journal of chemical information and computer sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

[41] G. Biau and E. Scornet, “A random forest guided tour,” *Test*, vol. 25, no. 2, pp. 197–227, 2016.

[42] D. Nguyen, C. Nguyen, T. Duong-Ba, H. Nguyen, A. Nguyen, and T. Tran, “Joint network coding and machine learning for error-prone wireless broadcast,” in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, IEEE, 2017, pp. 1–7.

[43] W. Jones and B. E. Launder, “The prediction of laminarization with a two-equation model of turbulence,” *International journal of heat and mass transfer*, vol. 15, no. 2, pp. 301–314, 1972.

[44] A. Kolmogorov, “Equations of turbulent incompressible fluid motion,” *Izv. Akad. Nauk SSSR, Ser. Fiz.*, vol. 6, no. 1, p. 2, 1942.

[45] P. G. Saffman, “A model for inhomogeneous turbulent flow,” *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, vol. 317, no. 1530, pp. 417–433, 1970.

[46] D. C. Wilcox *et al.*, *Turbulence modeling for CFD*. DCW industries La Canada, CA, 1998, vol. 2.

[47] D. C. Wilcox, “Reassessment of the scale-determining equation for advanced turbulence models,” *AIAA journal*, vol. 26, no. 11, pp. 1299–1310, 1988.

[48] D. C. Wilcox, “Formulation of the kw turbulence model revisited,” *AIAA journal*, vol. 46, no. 11, pp. 2823–2838, 2008.

[49] H. Choi and P. Moin, “Grid-point requirements for large eddy simulation: Chapmans estimates revisited,” *Physics of fluids*, vol. 24, no. 1, p. 011 702, 2012.

[50] B. F. Armaly, F. Durst, J. Pereira, and B. Schönung, “Experimental and theoretical investigation of backward-facing step flow,” *Journal of fluid Mechanics*, vol. 127, pp. 473–496, 1983.

[51] G. Biswas, M. Breuer, and F. Durst, “Backward-facing step flows for various expansion ratios at low and moderate reynolds numbers,” *J. Fluids Eng.*, vol. 126, no. 3, pp. 362–374, 2004.

[52] D. M. Driver and H. L. Seegmiller, “Features of a reattaching turbulent shear layer in divergent channelflow,” *AIAA journal*, vol. 23, no. 2, pp. 163–171, 1985.

[53] E. W. Adams, “Experiments on the structure of turbulent reattaching flow,” PhD thesis, Stanford University, 1984.

[54] R. Westphal, J. Eaton, and J. Johnston, “A new probe for measurement of velocity and wall shear stress in unsteady, reversing flow,” 1981.

[55] J. Kim, S. Kline, and J. Johnston, “Investigation of a reattaching turbulent shear layer: Flow over a backward-facing step,” 1980.

[56] J. Eaton and J. Johnston, “A review of research on subsonic turbulent flow reattachment,” *AIAA journal*, vol. 19, no. 9, pp. 1093–1100, 1981.