Large-eddy Simulations of Microscale Turbulent Winds over Arbitrarily Complex Terrain

A Dissertation

Presented in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

with a

Major in Mechanical Engineering

in the

College of Graduate Studies

University of Idaho

by

Anthony Rey DeLeon

Co-Major Professor: Inanc Senocak, Ph.D.

Co-Major Professor: Ralph Budwig, Ph.D.

Committee Members: John Crepeau, Ph.D.; Tao Xing, Ph.D.

Department Administrator: Steven Beyerlein, Ph.D.

August 2017

Authorization to Submit Dissertation

This dissertation of Anthony Rey DeLeon, submitted for the degree of Doctor of Philosophy with a major in Mechanical Engineering and titled " Large-eddy Simulations of Microscale Turbulent Winds over Arbitrarily Complex Terrain," has been reviewed in final form. Permission, as indicated by the signatures and dates given below, is now granted to submit final copies to the College of Graduate Studies for approval.

Co-Major Professors: _____ Date_____
Inanc Senocak, Ph.D.

_____ Date_____
Ralph Budwig, Ph.D.

Committee Members: _____ Date_____
John Crepeau, Ph.D.

_____ Date_____
Tao Xing, Ph.D.

Department Administrator: _____ Date_____
Steve Beyerlein, Ph.D.

Abstract

As the percentage of wind energy continues to increase in the overall energy portfolio, two essential components of integrating wind energy are designing wind turbines to withstand the forces created by the wind and correctly predicting the power production from wind farms. Accurate wind modeling over complex terrain is paramount to both applications. Existing wind forecasting techniques use computational resolutions on the order of a kilometer, whether the terrain be flat or complex. In this dissertation, a computational framework that applies the large-eddy simulation (LES) technique with resolutions on the order of 10 m is developed to improve predictions of turbulent winds over complex terrain. LES is a computationally expensive technique and a distinguishing aspect of the proposed framework is that the entire simulation is developed for massive-parallelism using clusters of graphics process units (GPUs). To reduce the near-wall resolution requirements, Reynolds-averaged Navier-Stokes (RANS) is used in tandem with LES as a wall-model. The hybrid use of RANS and LES leads to modeling artifacts such as a mismatch in the theoretical logarithmic law-of-the wall. A forcing technique that splits the driving mean pressure gradient based on conservation of mass principle is proposed. The split-forcing approach was found to reduce the modeling artifacts on coarse grids. A new turbulent inflow condition based on buoyancy perturbations is proposed for both smooth-wall-engineering and complex-terrain flows. For complex terrain winds, an improved immersed-boundary method is developed based on the equilibrium assumptions in the atmospheric surface layer, along with a pre-processor to process terrain geometry. The computational framework and individual components are thoroughly tested and validated using well-known benchmark cases. Finally, the computational framework is applied to study the feasibility of dynamic line-rating concept over complex terrain. Potential avenues of further research are also recommended.

Acknowledgements

My thanks go to Dr. Inanc Senocak and Dr. Ralph Budwig for the guidance and continuous mentorship they provided during the course of this research. I would also like to thank Dr. John Crepeau and Dr. Tao Xing for being part of my dissertation committee. I thank the University of Idaho and Idaho EPSCoR for generous support received through the University of Idaho President's Doctoral Scholars Award. Material in this dissertation was also supported by the National Science Foundation under Grant No. 1056110, 1043107, and 1229709. I also extend my thanks to the fellow students who helped obtain some of the results presented in this dissertation and co-author publications: Micah Sandusky, Clancy Umphrey, Tyler Phillips, Benjamine Miller, and Kyle Felzein. Thanks also go to Marty Lukes and Jason Cook for their help maintaining the computational resources used to perform this research.

Dedication

I would personally like to thank my wife and the rest of my family for their continuous support through my endeavors. This wouldn't have been possible without them. I would personally like to extend my appreciation to Dr. Inanc Senocak and Dr. Ralph Budwig for guiding me through this journey.

Table of Contents

List of Tables

List of Figures

Preface

This dissertation contains published work and work that has been submitted to the peer-review process. The role of the author of this dissertation in this published or submitted work is either primary author or a major contributor. Copyright information is provided in Appendix A. The roles are outlined as follows:

- "A Novel Fix to Reduce the Log-layer Mismatch in Wall-modeled Large-eddy Simulations of Turbulent Channel Flow," refereed at the ASME FEDSM 2016 research conference (DeLeon and Senocak, 2016), appearing on pages 16- 35 in this dissertation.

  Copyright permission can be found in §10.4 in Appendix A.

  This article was a conference paper for the ASME Fluids Engineering Division 2016 Summer Meeting. I am primary author and Dr. Inanc Senocak is the second author. As primary author, my contribution was to develop the split-forcing approach that reduced the modeling artifact known as log-layer mismatch.

- "Investigation of a Split-forcing Approach to Simulate Wall-modeled Turbulent Channel Flows with Hybrid RANS-LES Methods," submitted to AIAA Journal, appearing on pages 36- 54in this dissertation.

  The split-forcing approach presented in this section is under consideration for the AIAA Journal. I am the primary author with Dr. Inanc Senocak as a co-author. As primary author, my contribution was the continuing investigation of the split-forcing approach to higher Reynolds numbers and analysis of the method.

- "An Immersed Boundary Geometric Preprocessor for Arbitrarily Complex Terrain and Geometry," published in Journal of Oceanic and Atmospheric Technologies (Senocak *et al.*, 2015), appearing on pages 60- 82 in this dissertation.

  Copyright permission can be found in §10.4 in Appendix A.

This work was a joint effort among the six authors. As a co-author, I worked closely with the other authors in developing the algorithms and in the implementation.

- "Direct Numerical Simulation of Turbulent Katabatic Slope Flows with an Immersed Boundary Method," in-press for Boundary-layer Meteorology, appearing on pages 83-104 in this dissertation.

  Copyright permission can be found in §10.4 in Appendix A.

  Clancy Umphrey is the primary author, I am the second author, and Dr. Inanc Senocak is the third author. As second author, my contribution was to develop the general immersed boundary method framework that began in DeLeon (2012). I also collaborated closely with the first author in implementing the immersed boundary reconstruction schemes. My contributions are contained mostly in Sections 5.2 and 5.3 and the implementation of the schemes in Section 5.4.

- "A Direct-Forcing Immersed Boundary Method for Attached Flow Over Complex Terrain," under review in Boundary-layer Meteorology, appearing on pages 105- 129in this dissertation.

  I am the primary author with co-authors Micah Sandusky and Dr. Inanc Senocak. As primary author, my contribution was the formulation and implementation of the immersed boundary method presented in Chapter 6.

- "Dynamic Rating of Overhead Transmission Lines Over Complex Terrain Using a Large-eddy Simulation Paradigm," published in Renewable Energy (Phillips *et al.*, 2017), appearing on pages 172- 198 in this dissertation.

  Copyright permission can be found in §10.4 in Appendix A.

  My contribution was performing the CFD simulation to provide the wind velocity data for the dynamic line rating analysis as described in Section 9.4. Tyler Phillips is primary author, I am second author, and Dr. Inanc Senocak is the third author for this work appearing in Renewable Energy (Phillips *et al.*, 2017).

CHAPTER 1

Introduction

The wind engineering field encompasses many applications, with wind-structure interaction being perhaps the most commonly-known example. Wind loading on bridges, light-weight flexible structures, and wind turbines are common examples of wind-structure interactions. In the particular application of wind-structure interaction for wind turbines, quantities often sought after with high accuracy include wind shear and turbulence intensity. Wind shear refers to the wind speed profile tending to lower values as the height above ground level, h_{agl}, becomes smaller with the shape of the profile greatly depending on the surface roughness of the ground. Turbulence intensity refers to the root-mean-square amount of turbulent fluctuations in the wind flow. Both contribute to structural stresses within the turbine.

A wind velocity profile typically has increasing velocity with increasing h_{agl}. As a wind turbine rotates, the wind speed that a blade tip experiences at the highest position in a rotation will be considerably higher than when the same blade tip is at its lowest position in a rotation. To illustrate this concept, an example wind shear profile is given in Figure 1.1. A 127 m diameter swept area of a large wind turbine with a hub height of 135 m is represented as the rectangle in Figure 1.1 and will experience a wind speed of 14.8 m/s at the top of the rotation and a speed of 12.2 m/s at the bottom of the rotation, making for a difference of 2.6 m/s between the highest and lowest positions of the turbine blades. Turbulence intensity also has a similar effect, with the wind turbine experiencing variations in wind speed at high levels of turbulent intensity. For example, if the turbulence intensity is 20 % at the hub height of the previously-mentioned turbine wind shear example (the hub height is often used as the measuring height for turbulence intensity (International Electrotechnical Commission, 2017)), then the turbine blades can be expected to experience variations of wind speed 2.8 m/s from the mean value of 14.2 m/s at the hub height over short periods of time and will experience even higher variations as turbulence intensity is a root-mean-square quantity.

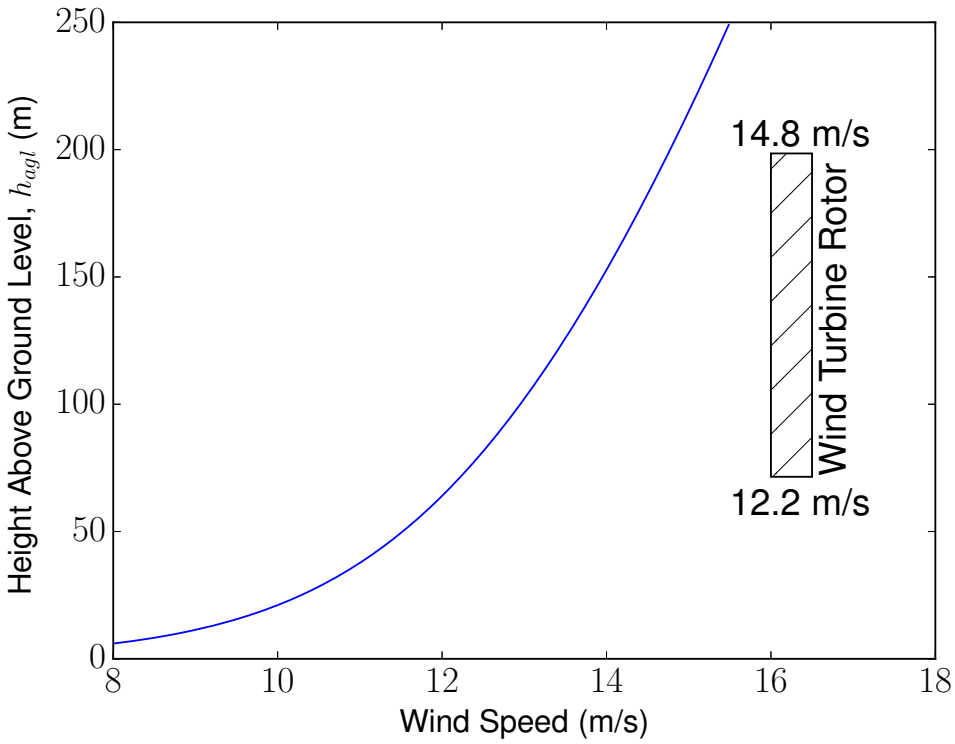


Figure 1.1: Wind shear example. Wind shear profile illustrating the difference in velocity experienced by wind turbine blade tips at the uppermost and lowermost positions in their rotation. The rectangle represents the swept area of a 127 m diameter wind turbine with hub height at 135 m.

Overall, the forces experienced by a wind turbine vary with h_{agl} and time and if this cyclic loading is not designed against, the turbine can fail prematurely, e.g. gearbox failure (Jain, 2011).

Wind resource assessment is the in-depth process of determining the economic feasibility of a potential wind farm (Jain, 2011). A wind-measurement campaign is performed for one to two years to mean wind speed and turbulence intensities. Data assimilation of historical satellite data is performed to adjust the mean wind speed. Using the adjusted value, economic feasibility of the project over a target lifetime is performed using various wind turbine models Jimenez (2013). With multiple turbines, wakes behind the turbine need to be accounted for in the resource assessment to avoid decreased power production from turbines in the wake of other turbines. Wake-turbine interactions need to be minimized when siting wind turbines. The wakes depend on the atmospheric conditions and the turbulence intensities of the incoming wind (Stevens and Meneveau, 2017).

Once a wind farm is operational, balancing authorities, such as the Bonneville Power Administration (Bonneville Power Administration, 2017), require accurate forecasts of wind power production. A balancing authority's responsibility is to efficiently schedule unit commitment of power-producing resources and dispatch electricity where it is in demand. Constant generation sources, such as natural gas burners or hydroelectric dams, have known spin up times and known power production estimates. Wind energy is categorized as variable generation because wind speed that is high enough to spin a wind turbine rotor occurs at various times of day with variable duration. To meet the demand on the electrical grid, i.e. the amount of electricity being used by consumers, a balancing authority needs to use both variable and constant generation. Without predictive capabilities for variable generation, a balancing authority may not be able to effectively commit and balance both constant and variable generation resources. If not enough generation units are committed, then not enough electricity will be produced to meet demand, leaving some consumers without electricity. If too many generation units are committed, then curtailment is a possibility to avoid damage

to the electrical grid. An example of curtailment with respect to wind energy is to engage the brakes of wind turbines to avoid the grid receiving more power from variable wind resources than the grid was designed to transport. The disadvantage is that renewable energy is lost for both the balancing authority and the wind farm owner who expects to sell the electricity produced.

Wind power forecasting models can be classified as either a physical, statistical, or learning (Foley *et al.*, 2012). A physical model, such a numerical weather prediction (NWP) model, is a simulation of atmospheric flow dynamics. A statistical model, the more traditional approach, analyzes a historical data to make a prediction. The learning approach uses artificial intelligence concepts to make predictions based on historical data. Combining classifications is referred to as hybrid methods. There is another prediction approach, the persistence approach (Soman *et al.*, 2010), that is not a model but is the concept that the wind speed and power production will not vary over a given period of time. Obviously, the approach is not robust and loses accuracy with a longer time frame but the persistence approach can be used as a benchmark for other methods. Long-term predictions, i.e. monthly, quarterly, or annual time frames, and very-short-term predictions, i.e. subhourly time frames, are usually well predicted by models requiring statistical input. Short-term predictions, i.e. one-hour-ahead, six-hour-ahead, and daily time frames, rely on the dynamics of the atmosphere and are expected to be better predicted by physical approaches. However, accurately capturing the dynamics of the atmosphere has multiple challenges, depending both on the time horizon and the physical size of the area of interest.

The aforementioned applications of wind turbine design, wind turbine siting, and wind farm power production would all benefit from a tool that is able to model the wind accurately, particularly when the farm or potential farm is located in areas of complex terrain such as hills, valleys, or canyons. The focus of this dissertation is on physical models of turbulent wind within the atmospheric boundary layer (ABL) over complex terrain. The ABL is defined as the lowest part of the atmosphere, typically being approximately 1 km above the

Earth's surface. The end goal that starts with the research presented in this dissertation is to accurately predict wind velocity and turbulence intensities on the physical length scales with the same order of magnitude as a wind turbine or a wind farm, e.g. numerical model resolution of 10-100 m in the horizontal directions and 1-20 m in the vertical direction. Models of the ABL at this resolution are referred to as microscale models in meteorology. The industry standard is to use mesoscale models, i.e. horizontal numerical resolutions of 1-100 km, for predicting wind power production meaning that the wind passing through a wind farm is modeled by a few points which will have to be interpolated. The algorithms are also chosen to perform well on the chosen graphics processing unit (GPU) computational accelerator (Jacobsen and Senocak, 2011, 2013; DeLeon *et al.*, 2013), i.e. the algorithm is well-suited to memory architecture of the GPU. The goal is to forecast the wind for power predictions and a distinguishing aspect of this work is the development of a LES framework exclusively for massively-parallel, multi-GPU supercomputers. The contribution of this dissertation is developing new approaches for components of an LES framework to simulate winds over complex terrain from a computational fluid dynamics perspective. However, before stating the goals of the research more specifically, the ABL is discussed in more detail to present the challenges and the difference from an aerodynamic boundary layer.

1.1 Atmospheric Boundary Layer

The ABL differs drastically from the smooth-wall boundary layer defined for aerodynamic applications. Perhaps the most significant is the diurnal cycle the ABL experiences. The diurnal cycle creates a different yet well-defined flow structure depending on time of day as shown in Figure 1.2. A typical diurnal cycle is described here although other atmospheric conditions can lead to the same atmospheric stability. During the day, the turbulence is convectively driven by heat fluxes from the warm ground or radiative cooling from clouds and driven mechanically by wind shear (Stull, 1988, pg. 12). This is referred to as the convective

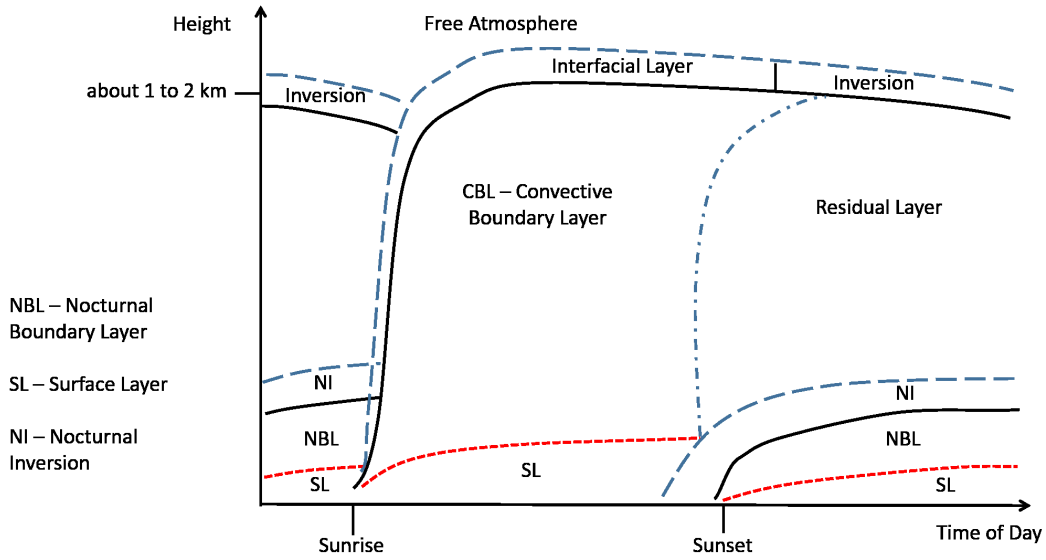


Figure 1.2: Sketch of diurnal cycle in the atmospheric boundary layer (Stull, 1988).

boundary layer. When buoyancy is negligible and driven predominantly by mechanical means, the ABL is said to be neutral. The lowest 10%, approximately the lowest 100 m, is driven mechanically and the mean velocity profile exhibits logarithmic behavior. At night, the cool ground creates a stably stratified flow structure with a bottom nocturnal boundary layer exhibiting intermittent turbulence and a residual layer above that is neutrally stratified (Stull, 1988, pg. 14). A surface layer is still present, although does not extend as high above the surface as during the daytime. These observations are made over flat terrain. As the terrain topography becomes more complex, the flow behavior becomes more difficult to model.

The surface of the Earth is composed of roughness elements of varying dimensions, e.g. buildings, trees, grass, rocks, meaning that the distinct turbulent flow regions, viz. the viscous sublayer or buffer layer, that develop in smooth-wall, wall-bounded turbulent flows, e.g. pipe flows or smooth-wall turbulent boundary layer, are not as distinctly observed in the atmosphere. Thus, the law-of-the-wall for smooth-wall, wall-bounded turbulence does not apply for ABLs. Earth's surface is also thermally active, meaning moisture and heat fluxes are significant surface boundary conditions for the turbulent flow. In the surface layer,

the mean velocity does exhibit logarithmic behavior as in engineering flows. However, the definition is a rough-surface log-law with a term for stability correction, ψ,

$$\frac{U}{u_\tau} = \frac{1}{\kappa}\left[\ln\left(\frac{z+z_0}{z_0}\right) + \psi\left(z, z_0, L\right)\right] \tag{1.1}$$

with an aerodynamic roughness length, z_0, which represents the location where the velocity is zero in the mean sense. For statically neutral or statically near-neutral atmospheric conditions, i.e. buoyancy effects do not change the laminar or turbulent state of the flow, the stability correction term goes to zero. The aerodynamic roughness length varies depending on the heights and shapes of the roughness elements but is always higher than ground level. To further illustrate the difference between an aerodynamic boundary layer mean velocity profile and a rough-surface boundary layer, Figure 1.3 compares the smooth-wall law-of-the-wall with a rough-wall log-law (Equation 1.1) using the same friction velocity value for both laws. Note that the slope of the velocity is similar away from the surface yet the rough-wall law does not start at the wall due to the aerodynamic roughness length.

Similar challenges shared by the ABL and general engineering flows are part of the motivation for investigating application of CFD techniques to microscale meteorological applications. While there are many more challenges than discussed here, three are presented that will be the focus of this dissertation. One shared challenge is to resolve turbulent fluctuations using the large-eddy simulation (LES) technique without requiring impractically-large computational resources. For context, Choi and Moin (2012) showed that wall-resolved LES of a high-Reynolds-number boundary layer takes approximately $Re_{L_x}^{13/7}$ grid points, where L_x in this case is the streamwise flat-plate length. The computational cost is a result of the essential idea of LES; LES is a turbulence modeling technique that filters turbulent eddies depending on their characteristic size. Eddies passing below the filter width are considered statistically universal and are modeled whereas the eddies passing above the filter width are numerically calculated, or resolved. A rule of thumb for a resolved LES is that 80% of the

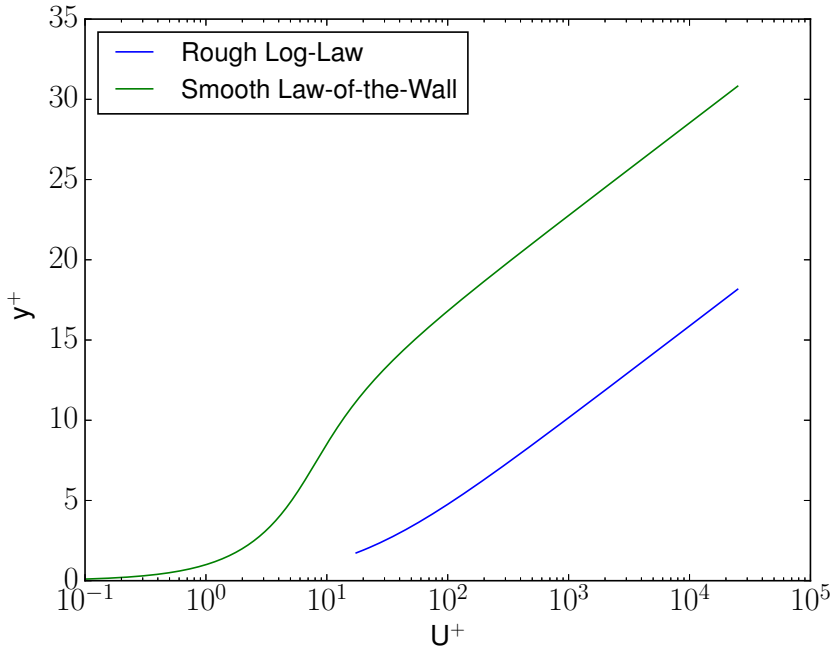


Figure 1.3: A sketch of a rough log-law and a smooth-walled law-of-the-wall using the same friction velocity value in both laws.

turbulent kinetic energy needs to be resolved, which is contained within scales larger than one-sixth the integral length scale of turbulence (Pope, 2000, pg 242). As the length scales of turbulence diminish as a surface is approached leading to smaller and smaller integral length scales, more numerical grid resolution is required to properly resolve 80% of the turbulent kinetic energy at these scales. As the order of magnitude of Reynolds number is $O(10^6)$ in the ABL, these near-surface scales are very, very small compared to the larger scales away from the surface. One approach to alleviate computational costs of LES is to use a wall model (Deardorff, 1970; Schumann, 1975), which has commonly been in ABL simulations since Moeng (1984) proposed a wall-modeled LES approach for the ABL. The idea of a wall model is to use a coarse mesh near the surface and a model for the velocity and Reynolds stresses to provide accurate information to the rest of the LES, which is resolved. The challenge is then to combine the wall-model with the resolved LES without introducing modeling artifacts.

A second shared challenge, again concerning computational requirements, is the impo-

sition of boundary conditions from complex geometries, or in the case of ABL, complex terrain. Even with a perfect wall-model, complex geometries present a meshing challenge as concavities and convexities can lead to skewed grid cells if using structured meshing types. Unstructured grids are able to capture complex geometries with high mesh quality but also may require high mesh refinement, defeating the purpose of saving computational resources. The immersed boundary method (Mittal and Iaccarino, 2005; Sotiropoulos and Yang, 2014) uses Cartesian grids to immerse a solid thus avoiding cumbersome meshing and poor mesh quality. The challenge becomes imposing the boundary conditions on the Cartesian grid. IBM has been studied extensively in the CFD community and has been extended to ABL but research is an ongoing area.

A third shared challenge is the inflow boundary conditions. Idealized boundary conditions, such as periodic, are not suitable for obtaining realistic flow predictions. Inflow and outflow conditions are much better suited. However, LES requires inflow information on the structure of the turbulence, i.e. correct spatial and temporal correlations. If not present, the LES will require an impractically long development region, increasing the computational cost tremendously. Turbulent inflow conditions for LES is an on-going research area in the engineering and atmospheric communities (Keating *et al.*, 2004; Tabor and Baba-Ahmadi, 2010; Wu, 2017). Wu (2017) highlights the need for more robust methods that maintain the three-dimensionality of the mean velocity in atmospheric flows. However, few studies have investigated turbulent inflows that have time-varying mean flow directions as experienced in the ABL. An example mean flow direction as a function of time that is a polynomial fit to actual wind directional measurements is shown in Figure 1.4.

1.2 Research Questions

The aforementioned shared challenges between the ABL and general engineering flows are the focus of the dissertation. The research questions that motivated the research presented in this dissertation are as follows:

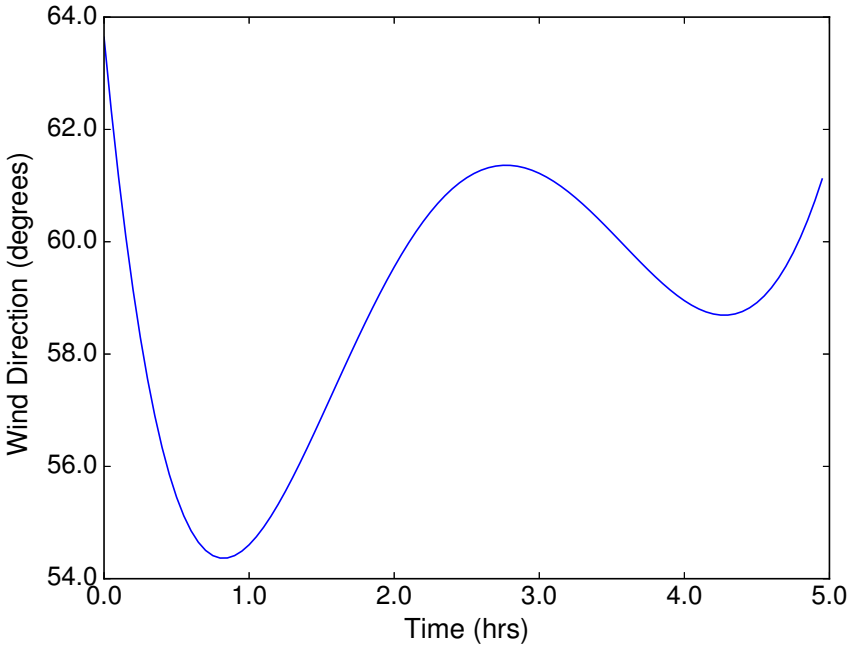


Figure 1.4: Time-varying mean flow direction example. A polynomial fitted to actual wind speed measurement data.

1. How reliable is the wall-modeled LES approach of hybrid RANS-LES for turbulent winds over complex terrain?

2. How accurate is the wind flow predicted when using equilibrium attached-flow assumptions to impose boundary conditions created by an immersed complex terrain?

3. How can buoyancy-driven flow physics be used as a turbulence inflow generation technique for LES of wind flow over complex terrain?

The investigation of these research questions are presented in this dissertation. Chapters 2 and 3 are a proposed method resulting from research into question 1. Chapters 4, 5, and 6 discuss the various aspects of research into question 2. Chapter 7 pertains to research question 3 for applications in engineering-type flows. Chapter 8 also pertains to research question 3 and focuses on wind flow over complex terrain. An application of the methods proposed in this dissertation can be found in Chapter 9. Recommendations for future work are discussed in Chapter 10.

1.2.1 Discussion of Hybrid RANS-LES

Hybrid RANS-LES encompasses a broad range of approaches that use RANS when grid resolution is insufficient for an LES to resolve 80% of the turbulent kinetic energy (TKE) in a flow. A discussion on the eighty-percent-TKE, fully-resolved LES criteria can be found in Pope (2000, pg 240-42, 570). The goal of hybrid RANS-LES is to alleviate computational cost of high-Reynolds-number simulations by taking advantage of the lower mesh resolution requirements of RANS. A RANS model is used near the wall such that the small scales in the viscous sublayer need not be resolved but are rather the contributions to the flow far away from the wall are modeled. The challenge becomes the interface condition between RANS and LES. RANS simulates only the largest of turbulent scales, i.e. the mean values, and models all turbulent fluctuations. LES resolves the larger scales of turbulence and models the smaller isotropic scales, the separation of larger and smaller scales accomplished with a low-pass filter operation. For accurate results, the LES region requires information on the resolved Reynolds stress components, otherwise the model for isotropic small-scales will push the LES result to an inaccurate solution. The reason is that not all TKE is being accounted for by the lack of Reynolds stresses coming from the RANS region and the LES will push another variable towards an unnatural value to maintain a global momentum balance, often times the mean velocity gradients. The unnatural deformation of mean velocity gradients is referred to as the *log-layer mismatch* problem.

As Reynolds numbers in atmospheric flows are on the order of $\mathcal{O}(10^6)$, wall-modeled LES is a necessity. Hybrid RANS-LES was chosen as the model can be formulated without knowledge of the surface stresses. In Chapters 2 and 3, work into reducing the log-layer mismatch problem in a hybrid RANS-LES of turbulent channel flow is presented. The idea is to validate the approach in a known canonical case before moving onto more general flows, such as turbulent wind over complex terrain. A systematic investigation of approaches led to the proposal of a method referred to as *split forcing*, details of which will be described in

the following sections. However, split forcing is 'channel-friendly' and does not extend easily to general flows. The idea is that the concept can lead to other developments to reduce log-layer mismatch in general flows. More discussion to this matter is given in Chapter 10.

Log-layer mismatch arises when hybridizing LES with a Reynolds-averaged Navier-Stokes (RANS) descriptions of turbulence for a wall model. RANS only provides mean quantities and does not provide the necessary resolved Reynolds shear stresses to the LES zones. Most research to eliminate the log-layer mismatch involves adding forcing terms in the momentum equations that are not based on physical principles. The split-forcing method is presented in Chapters 2 and 3 using two high Reynolds number for a turbulent, periodic channel flow. The split-forcing approach is distinct from other approaches in literature as it relies on conservation of mass to calculate the magnitude of the forcing terms to reduce the log-layer mismatch in wall-modeled LES. While not necessarily applicable to flows other than a plane channel flow, the study stemmed from searching for an answer to research question 1 and provided interesting results with insight into hybrid RANS-LES. The work in Chapter 2 is presented as a work-in-progress manner for a prognostic hybrid RANS-LES approach while the work in Chapter 3 is presented as a numerical experiment.

1.2.2 Discussion of Immersed Boundary Methods

An immersed boundary (IB) method is an alternative to body-fitted or unstructured meshes. The general idea is to immerse a geometry in a Cartesian grid to greatly simplify the meshing process. The challenge is then the imposition of the boundary conditions at the surface as the immersed surface does not coincide with the Cartesian grid. With laminar flows or turbulent flows with mesh resolution in the viscous sublayer, the velocity behavior can be represented by linear or quadratic laws and a reconstruction scheme, i.e. interpolation scheme, can be used to impose velocity boundary conditions. However, in turbulent flows without resolution of the viscous sublayer, the IB method can be a bit more challenging. Also, with regards to turbulent wind flow over complex terrain, IB methods are highly desirable to avoid the

possibility of skewed cells in terrains exhibiting a variety of concavities and convexities whose characteristic size are on the order of the mesh resolution. Poor aspect ratio of mesh cells can pollute a solution and complex terrain is susceptible to skewed mesh cells. Chapter 4 addresses the issue of obtaining geometric information for IB methods in regards to complex terrain. Chapter 5 is the validation of an immersed boundary method use in a DNS of katabatic slope flow. Finally, Chapter 6 explores an IB method for turbulent wind flows over complex terrain.

1.2.3 Discussion of Buoyancy-Perturbations Turbulent Inflow

Similar to the discussion in Chapter 2, LES requires accurate turbulent flow structure with the proper correlations, otherwise resolved quantities are pushed to unnatural values due to missing or improper resolved Reynolds stresses. While Chapter 2 looked at the lack of resolved Reynolds stresses LES receives from a RANS wall model in hybrid RANS-LES, Chapter 7 looks at the incoming flow boundary condition in non-periodic, fully-resolved LES. Unlike in Chapter 2, the viscous sublayer is resolved. Chapter 7 focuses on the development of the *perturbation box* turbulent inflow generation method in canonical engineering-type flows. Again, the idea is to validate the method in a canonical case before applying to more general flow conditions. Chapter 8 applies the proposed turbulent inflow method to complex terrain. The purpose is to investigate if the parameters from the engineering-type flow cases apply to turbulent wind flow over complex terrain. The mean flow characteristics of the wind flowing over hills are investigated. Turbulent quantities are not included in this study as the hybrid RANS-LES approach does not allow for accurate representation of the second-order moments very close to the surface where the measurement campaigns gathered data. More discussion on this issue in Chapter 10.

With regard to LES of atmospheric flows, incoming flow may be prescribed from a larger mesoscale weather model that only provides a mean flow direction and magnitude. Thus, a method that can generate realistic and non-periodic turbulence at the LES inflow is needed

as LES requires accurate turbulent flow field information. Also, the mean flow direction may change in time. An inflow condition needs to accommodate this. However, most turbulent inflow techniques are developed for engineering-type flows where time-varying mean flow direction is not present. The work presented in this dissertation accommodates meandering winds.

1.2.4 Discussion of Dynamic Transmission Line Rating

In Chapter 9, the wall-modeled LES simulation with IB methods presented in this dissertation in a proof-of-concept of the dynamic line rating concept. The simulation presented in Section 9.4 was over an area 18 km $\times$ 24 km that required 327 hours (13.6 days) to complete on 32 GPUs. The dynamic line rating concept is to use the convective cooling of wind flowing over transmission lines to dynamically change the maximum limit of electricity that can be safely transmitted along the lines.

CHAPTER 2

A Novel Fix to Reduce the Log-layer Mismatch in Wall-Modeled Large-eddy Simulations of Turbulent Channel Flow

DeLeon, R. and Senocak, I. "A novel fix to reduce the log-layer mismatch in wall-modeled large-eddy simulations of turbulent channel flow" *In ASME 2016 Fluids Engineering Division Summer Meeting collocated with the ASME 2016 Heat Transfer Summer Conference and the ASME 2016 14th International Conference on Nanochannels, Microchannels, and Minichannels.* 2016.

2.1 Abstract

The log-layer mismatch arises when a Reynolds-averaged Navier-Stokes (RANS) model is blended with a large-eddy simulation (LES) model in a hybrid fashion. Numerous researchers have tackled this problem by simulating a turbulent channel flow. We show that the log-layer mismatch in hybrid RANS-LES can be reduced substantially by splitting the mean pressure gradient term in the wall-normal direction in a manner that keeps the mass flow rate constant. Additionally, an analysis of the wall-normal variation of the friction velocity shows a constant value is recovered in the resolved LES region different than the value at the wall. Second-order turbulence statistics agree very well with direct numerical simulation (DNS) benchmarks when scaled with the friction velocity extracted from the resolved LES region. In light of our findings, we suggest that the current convention to drive a turbulent periodic channel flow with a uniform mean pressure gradient be revisited in testing eddy-viscosity-based hybrid RANS-LES models as it appears to be the culprit behind the log-layer mismatch.

2.2 Introduction

A common drawback for broad adoption of wall-bounded large-eddy simulation (LES) in engineering flows is the high computational cost of fully resolving the near-wall region. The stringent resolution requirement arise from the scales of turbulence diminishing drastically as the wall is approached Pope (2000). Wall-modeled LES (WMLES) is an alternative to a fully-resolved LES which alleviates the stringent near-surface grid requirements and allows adoption of LES in general engineering flows.

WMLES has two major elements: modeling the wall shear stress directly and modeling the inner part of the boundary layers with a Reynolds-averaged Navier-Stokes representation of turbulence. Modeling the wall shear stress is often accomplished by replacing the no-slip condition with a wall shear stress derived from flow variables Schumann (1975). In this regard, a coarse grid can be used as the fine resolution needed to resolve the viscous sublayer with the no-slip condition at the wall is no longer required. The disadvantage of modeling the wall shear stress is that a general theory is needed to compute the wall shear stress during computations. A RANS description of turbulence does not require resolution of energy-containing eddies meaning the grid can be much coarser than an LES grid. However, RANS and LES fields produce statistically different quantities. The process of hybridizing RANS and LES, referred to as *hybrid RANS-LES*, is often considered an ad-hoc approach as modeling artifacts resulting from two different quantities are often remedied by introducing empirical mechanisms to energize the flow. The focus of this study is to demonstrate that modeling artifacts can be remedied with the use of conservation principles only.

The commonly known *log-layer mismatch* has plagued hybrid RANS-LES approaches when the velocity profile overshoots the theoretical log-law profile above the interface between the RANS and LES regions, often observed in hybrid RANS-LES of channel flows. The periodic channel is a common testbed for hybrid RANS-LES models. An early study that highlighted the existance of log-layer mismatch in a turbulent channel flow simulation

was by Nikitin et al. Nikitin *et al.* (2000), who used the detached-eddy simulation (DES) Spalart *et al.* (1997) technique as the wall-modeled LES. Turbulence was sustained but velocity gradients were too steep and the issue was referred to as the "DES buffer layer." Hamba Hamba (2003) observed the log-layer mismatch issue in channel flow with simple eddy viscosity models and proposed different filters to alleviate the log-layer mismatch.

Studies by Keating and Piomelli Keating and Piomelli (2006) and Davidson and Billson Davidson and Billson (2006) demonstrated methods of energizing the flow at the RANS-LES interface by using stochastic forcings and imposing turbulent fluctuations, respectively. However, both of these approaches would be difficult to generalize to other flows. Dual-mesh approaches have also been proposed, the first being the two-layer model by Balaras and Benocci Balaras and Benocci (1994). Here, the RANS and LES are solved on separate domains and the solutions are hybridized together. Xiao and Jenny Xiao and Jenny (2012) is a more recent example of a dual-mesh hybrid RANS-LES. For more hybrid RANS-LES methodologies, the reader is referred to reviews by Frölich and von Terzi Fröhlich and von Terzi (2008) and Piomelli Piomelli (2008).

With regards to hybrid RANS-LES approaches that employ eddy-viscosity-type models, the log-layer mismatch in periodic channel flow stems from a combination of missing resolved shear stresses and high eddy viscosity when transitioning from RANS to LES. Because of this fact, many approaches focus on energizing the flow at the RANS-LES interface through empirical techniques, even leading some to argue that hybrid RANS-LES are not a viable approach because of these empirical remedies to fix the log-layer mismatch. We propose an alternative methodology that relies upon conservation principles only to reduce the log-layer mismatch. The proposition is simple, give the conservation of momentum another degree of freedom by allowing the driving mean pressure gradient to vary in the wall-normal direction depending on the magnitude of the eddy viscosity. We test our hypothesis on a periodic channel flow with a driving mean pressure gradient that varies in the wall-normal direction while maintaining a constant mass flow rate.

In addition to fixing the log-layer mismatch, we also place our first grid point in the logarithmic region of the channel flow. As is commonly known and performed, the placement of the first grid point at approximately one wall unit away from the wall gives the best results. Most, if not all, studies of hybrid RANS-LES for a periodic channel flow place the first point in the viscous sublayer and alleviate computational costs by coarsening the grid in the streamwise and spanwise directions only. Hence, a distinguishing feature of our investigation is demonstrating channel flow simulations where the grid is coarsened in *all* three directions. We use the standard wall shear stress model proposed by Schumann and Grötzbach in our simulations where the first off-wall grid point falls into the logarithmic region, i.e. $y^+ \approx 30$.

2.3 Methodology

We solve the filtered form of the governing equations for incompressible flows which are written as

$$\frac{\partial \overline{u}_j}{\partial x_j} = 0, \tag{2.1}$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u}_i \overline{u}_j\right) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\nu \overline{S}_{ij} - \tau_{ij}\right), \tag{2.2}$$

where

$$S_{ij} = \frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right), \tag{2.3}$$

is the deformation tensor, and

$$\tau_{ij} = \overline{u_i u_j} - \overline{u}_i \overline{u}_j, \tag{2.4}$$

is the tensor representing the interaction of the modeled subgrid-scales on the resolved energy-containing scales.

The governing equations are solved using a three-dimensional, incompressible, GPU-accelerated flow solver Thibault and Senocak (2012); Jacobsen and Senocak (2013); DeLeon *et al.* (2013). The code uses the projection algorithm Chorin (1968) on directionally-uniform Cartesian grids with second-order central differences for spatial derivatives and a second-order Adams-Bashforth scheme for time advancement. The pressure Poisson equation is solved by an amalgamated 3D geometric parallel multigrid solver specifically designed for GPU clusters Jacobsen and Senocak (2011).

2.3.1 Mean Streamwise Pressure Gradient in Periodic Channel Flow

As periodic boundary conditions do not introduce an incoming mass flow rate, a mean streamwise pressure gradient forcing term, $\langle \frac{dp}{dx} \rangle$, must be applied to the streamwise momentum equation to drive the flow. There are two ways to formulate $\langle \frac{dp}{dx} \rangle$: impose a constant body force balancing the wall shear stress but causing the mass flow rate to fluctuate, or dynamically adjust $\langle \frac{dp}{dx} \rangle$ to maintain a constant target mass flow rate. The latter is advantageous when the wall shear stress is not known. With both approaches, $\langle \frac{dp}{dx} \rangle$ is applied uniformly to the entire domain at any given time step.

The forcing method chosen for this study to ensure mass conservation is that of Benocci and Pinelli (1990), which can be formulated as follows

$$\left\langle \frac{dp}{dx} \right\rangle^{n+1} = \left\langle \frac{dp}{dx} \right\rangle^{n} - \frac{2}{\Delta t}\left(\frac{\dot{m}^{n+1}}{A_c} - \frac{\dot{m}^{0}}{A_c}\right) + \frac{1}{\Delta t}\left(\frac{\dot{m}^{n}}{A_c} - \frac{\dot{m}^{0}}{A_c}\right), \tag{2.5}$$

where the superscript n is the time step counter, Δt is the physical time step value required to be held constant throughout the simulation as per the original formulation, $\dot{m}^0$ is the prescribed mass flow rate, and A_c is the constant cross-subsectional area. Other examples of forcing terms that maintain constant mass flow rate are those of You *et al.* (2000) that

modifies the psuedo-pressure term in a fractional step method and Brandt (2003) which uniformly corrects the velocity field to match the correct mass flow rate.

2.3.2 Split Forcing of Mean Streamwise Pressure Gradient

Our proposed approach for hybrid RANS-LES using blended eddy viscosity models is to simply allow $\frac{dp}{dx}$ to vary in the wall-normal direction. The hypothesis is that the velocity gradient can then relax toward a more natural value with a spatially-varying mean pressure gradient. The driving mean pressure gradient, referred to as forcing hereinafter, is decomposed into a piece-wise function of distance from the wall. To uphold conservation of mass, the magnitude in each piece, referred to as a forcing region hereinafter, is governed by modifying Eq. 2.5 to

$$\left\langle \frac{dp}{dx} \right\rangle_r^{n+1} = \left\langle \frac{dp}{dx} \right\rangle_r^{n} - \frac{2}{\Delta t}\left(\frac{\dot{m}_r^{n+1}}{A_c} - \frac{\dot{m}_r^{0}}{A_c}\right) + \frac{1}{\Delta t}\left(\frac{\dot{m}_r^{n}}{A_c} - \frac{\dot{m}_r^{0}}{A_c}\right), \tag{2.6}$$

where the subscript r represents the region and $\dot{m}_r^0$ is the target mass flow rate for the corresponding forcing region. Conservation of mass in enforced by

$$\dot{m}_T^0 = \sum_{r=1}^{N} \dot{m}_r^0, \tag{2.7}$$

where $\dot{m}_T$ is total mass flow rate through the channel. For this study, the mass flow rate is calculated at the inflow plane and the target mass flow rate is obtained from a direct numerical simulation (DNS). We refer to this approach as *split forcing*.

The demarcations of each forcing region are chosen based on the magnitude of eddy viscosity for the conventional forcing approach. Figure 2.1 shows an eddy viscosity profile for a conventionally-forced WMLES $\mathrm{Re}_\tau = 2000$ channel flow with 65 wall-normal grid points and the first grid point placed in the logarithmic region. The mean eddy viscosity is normalized by the value in the core of the flow which is nearly constant, i.e. determined

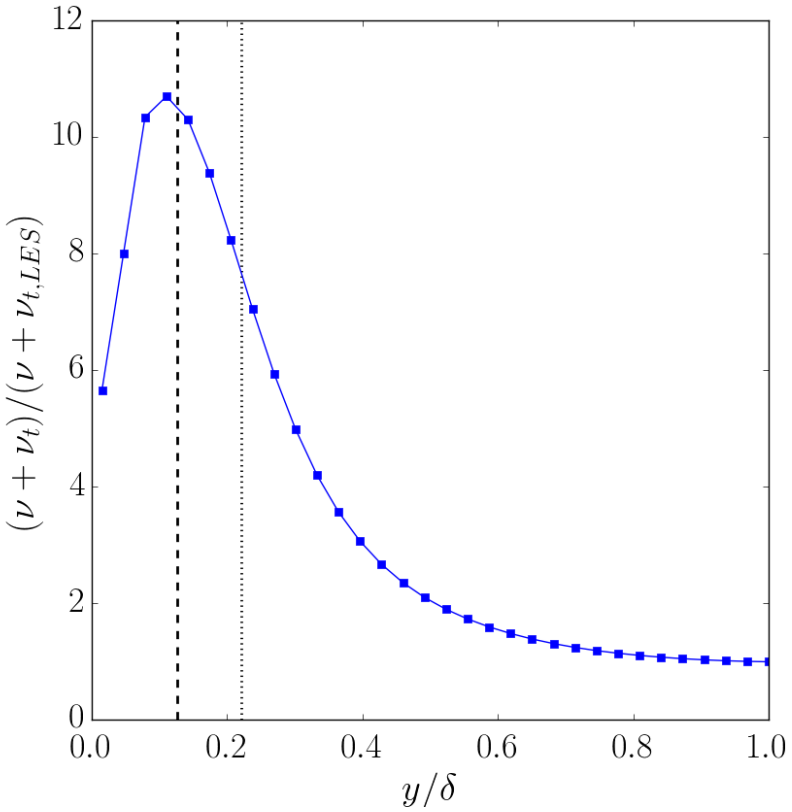


Figure 2.1: Mean eddy viscosity profile. Obtained from WMLES of $\mathrm{Re}_\tau = 2000$ channel flow with 65 wall-normal grid points and the first grid point placed in the logarithmic region. The eddy viscosity is normalized by the eddy viscosity obtained from the core region of the channel flow. The dashed line represents the RANS-LES interface as the first split forcing demarcation and the dotted line represents the chosen height for the second split forcing demarcation.

solely by the sub-grid model. The choice of normalization is based on illuminating the magnitude difference between RANS and LES regions. For this study we investigate three forcing regions. The RANS-LES interface is chosen as the first demarcation and the second demarcation chosen where the eddy viscosity is approximately eight times that of the LES core value as shown in Fig. 2.1. While not shown in this paper, two forcing regions were also investigated but did not perform as well as three regions due to high eddy viscosity present in the LES region.

2.3.3 Turbulence Modeling

Turbulence closure in the LES region is accomplished by the eddy viscosity sub-grid scale (SGS) model. We use both a constant model coefficient (Smagorinksy, 1963) and a dynamic model coefficient with the methodology of the Lagrangian dynamic model (Meneveau *et al.*, 1996). The eddy viscosity model is given by

$$\overline{\tau}_{ij} = \nu_t \overline{S}_{ij}, \tag{2.8}$$

where the eddy viscosity is calculated by

$$\nu_t = (C_S \Delta)^2 \, |\overline{S}_{ij}|, \tag{2.9}$$

where Δ is the filter width and C_S is the model coefficient. When using a constant coefficient, 0.16 is the prescribed C_S value.

An eddy viscosity model is also used for the RANS region and is the mixing length model of Prandtl (1925)

$$\nu_t = (\kappa y)^2 \, |\overline{S}_{ij}|, \tag{2.10}$$

where κ is the von Kármán constant of 0.41 and y is the distance from the wall.

The hybrid RANS-LES approach used in this study is similar to that of Senocak *et al.* (2007). In their study, they hybridize RANS and LES by blending the square of the length scales with an exponential blending function. We slightly modify their approach by blending the length scale, not the square of the length scale, creating a mixing length to blend RANS and LES in a fashion similar to DES by defining l_{mix} as,

$$l_{mix} = \left[1 - \exp\left(-y/h\right)\right] C_S \Delta + \exp\left(-y/h\right) \kappa y, \tag{2.11}$$

where y is the wall-normal coordinate and h is the RANS-LES transition height. The eddy viscosity then becomes,

$$\nu_t = l_{mix}^2 |\overline{S}|. \tag{2.12}$$

2.3.4 Wall Boundary Conditions

As we aim to place the first off-wall grid point in the logarithmic region ($y^+ > 30$), we obtain the wall shear stress through the wall model of Schumann (1975), later modified to a more general form by Grötzbach (1987). The essence of the Schumann model is to provide an instantaneous wall shear stress boundary condition by providing components of the total wall shear stress and explicitly enforcing the impermeability condition. The model is

$$\overline{\tau}_{xy,w}\left(x, z, t\right) = \frac{\overline{u}\left(x, y_1, z, t\right)}{\langle \overline{u}\left(y_1\right) \rangle} \langle \overline{\tau}_{xy,w} \rangle, \tag{2.13}$$

$$\overline{v} = 0, \tag{2.14}$$

$$\overline{\tau}_{zy,w}\left(x, z, t\right) = \frac{\overline{w}\left(x, y_1, z, t\right)}{\langle \overline{u}\left(y_1\right) \rangle} \langle \overline{\tau}_{xy,w} \rangle, \tag{2.15}$$

where y_1 is the first off-wall grid point. Schumann originally obtained the mean wall shear stress from the momentum balance in a channel flow, $\langle \overline{\tau}_{xy,w} \rangle = -\delta \langle \frac{\partial \overline{p}}{\partial x} \rangle$, where $\langle \frac{\partial \overline{p}}{\partial x} \rangle$ was given

Table 2.1: Physical domain parameters and grid sizes.

Case	Re_τ	N_X	N_Y	N_Z	L_X	L_Y	L_Z
1	2000	65	65	65	$2\pi\delta$	$2\ \delta$	$\pi\delta$
2	2000	257	65	193	$8\pi\delta$	$2\ \delta$	$3\ \pi\delta$

a priori. Grötzbach generalized this wall model by determining the stress from the log-law:

$$\frac{\langle \overline{u}\,(y_1)\rangle}{u_\tau} = \frac{1}{\kappa}\ln\left(\frac{y_1 u_\tau}{\nu}\right) + B, \tag{2.16}$$

$$\langle \overline{\tau}_{xy,w}\rangle = \rho u_\tau^2, \tag{2.17}$$

where κ is the von Kármán constant, B is a intercept constant, and ρ is density (set to unity in this study). While eliminating the need for an *a priori* $\langle\frac{\partial \overline{p}}{\partial x}\rangle$, Grötzbach's modification requires knowledge of κ and B. For this study, we adopt the commonly used values of $\kappa = 0.41$ and $B = 5.2$, but various other possible values exist (Nagib and Chauhan, 2008; Lee and Moser, 2015).

To avoid contamination from discretization and SGS modeling errors near the wall (Cabot and Moin, 2000; Kawai and Larsson, 2012), Eq. 2.16 is solved at the third off-wall grid point such that y_1 becomes y_3 (Lee *et al.*, 2013). The strain rate tensor in Eq. 2.12 requires the computation of wall-normal derivatives and will be incorrect due to the coarseness of the grid if calculated explicitly. Therefore, the eddy viscosity in the first off-wall grid cell is computed via linear extrapolation from the second and third off-wall grid cells (Cabot and Moin, 2000).

2.4 Results and Discussion

The periodic boundary conditions in all channel flows were applied in the streamwise and spanwise directions corresponding to the x-direction and z-direction, respectively. The wall shear stress boundary conditions were applied in the wall-normal direction. The flow was ini-

Table 2.2: Non-dimensional domain parameters. The superscript, $^+$, indicates normalization by friction velocity, u_τ, and molecular viscosity, ν. y_1^+ represents the wall units of the first off-wall grid point.

Case	Re_τ	Δx^+	Δy^+	Δz^+	y_1^+
1	2000	200	64	1004	32
2	2000	198	64	99	32

Table 2.3: Split forcing parameters. The SGS model (SM for the original Smagorinsky model and LD for the Lagrangian dynamic model), the RANS-LES transition height, h, and the split forcing height, h_{SP}, for each set of domain parameters, normalized by friction velocity, u_τ, and molecular viscosity, ν, indicated by the superscript, $^+$, and by channel half-height, δ.

Case	Re_τ	Model	h^+	h_{SP}^+	h / δ	h_{split}/δ
1	2000	SM	254	444	0.127	0.222
2	2000	LD	254	444	0.127	0.222

tialized with high amplitude, low-frequency, divergence-free sinusoidal velocity gradients imposed on a smooth-wall logarithmic mean profile. Development time was 200 eddy turnover times (δ/u_τ where δ is the channel half-height) and turbulent statistic samples were collected over an additional 200 eddy turnover times. For quantities normalized by u_τ, the friction velocity is the time average of u_τ calculated during the simulation by the Schumann-Grötzbach model in Eq. 2.16. The initial results shown here reflect this normalization. We later discuss how this normalization does not hold for our split forcing approach and propose a alternate method to obtain a normalization factor.

We present results for friction Reynolds number, Re_τ, of 2000. The domain and grid parameters for each of the three cases are given in Tab. 2.1. The non-dimensional grid spacing normalized by the theoretical u_τ and the kinematic viscosity are given in Tab. 2.2. We show benefits of the split forcing approach with three forcing regions, which we refer to as *tri-split forcing*. The parameters for the forcing regions demarcations of the RANS-LES interface, h and the splitting height, h_{SP}, discussed in the methodology are given in Tab. 2.3. In all plots, h is represented by a dashed black line and h_{SP} is represented by a dotted black

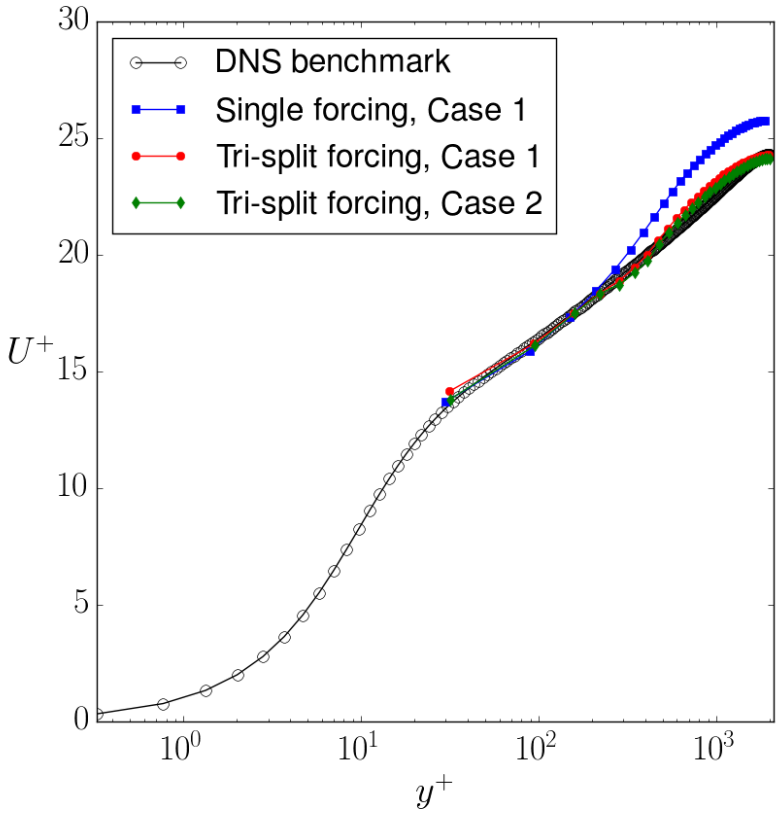


Figure 2.2: Non-dimensional mean velocity profiles. $\mathrm{Re}_\tau = 2000$ using case 1 and case 2 demonstrating the traditional single forcing region in hybrid RANS-LES and three forcing regions (tri-split forcing) in the split forcing approach. Friction velocity derived from wall shear stress.

line. We originally attempted two forcing regions (one for RANS, the other for LES) but the results were not as beneficial as the results for three regions and are not shown here. We also use two different SGS models (also given in Tab. 2.3) to demonstrate the applicability of the split forcing approach. We use the DNS data of Hoyas and Jiménez (2006) for $\mathrm{Re}_\tau = 2000$ comparisons.

2.4.1 The Effect of Split Forcing

The primary motivation behind pursuing the split forcing approach was to reduce the log-layer mismatch problem. Figure 2.2 shows how using a single forcing region (the traditional uniform body force) on the smaller Case 1 domain at $\mathrm{Re}_\tau = 2000$ creates an overshoot in

the mean velocity profile. For the small domain, we used the original Smagorinsky model. Figure 2.2 also shows the effect of the split forcing approach with three forcing regions has on the mean velocity profile on the same small domain, also using the Smagorinsky model. By creating the different forcing regions, implicit shear stresses are introduced at the interfaces of these forcing regions allowing the velocity gradient to conform to a more natural form while Reynolds stresses have yet to develop near the RANS-LES interface. As the original Smagorinsky model tends to be overly dissipative, we perform the tri-split forcing approach on the larger Case 2 domain with the Lagrangian dynamic Smagorinsky model which also gives the same result of improving the mean velocity profile.

2.4.2 Variation of Friction Velocity

We discovered that our proposed split forcing approach results in a different normalizing factor for turbulence statistics in the LES region that is not the wall shear stress. In theory, a channel flow with no mean acceleration has a total shear stress profile (Pope, 2000) of

$$\tau = \tau_w \left(1 - \frac{y}{\delta}\right), \tag{2.18}$$

where τ_w is a constant and gives the normalization factor of $u_\tau = \sqrt{\tau_w}$ (density is set to unity). However, we are introducing shear stresses into the total shear stress with the split forcing approach. It follows that Eq. 2.18 no longer holds with split forcing, more specifically τ_w being the normalizing scale for the entire flow no longer holds. Instead of having a single friction velocity, we now have a friction velocity dependent on the normal distance from the wall, or a friction velocity profile. The concept of a "local" friction velocity is not unprecedented as it has been used to normalize Reynolds stresses in the study of prescribing mean velocity profiles to DNS performed by Tuerke and Jiménez (2013).

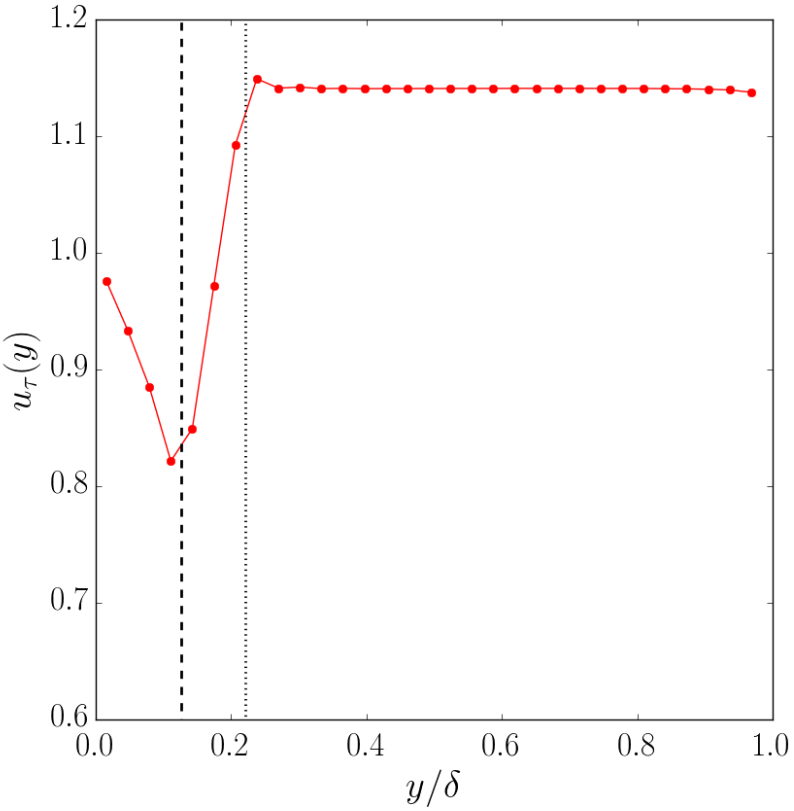


Figure 2.3: Friction velocity profile. Derived from total shear stress profile obtained from three forcing regions in case 2 at $\mathrm{Re}_\tau = 2000$. Dashed line: h, RANS-LES interface; dotted line: h_{SP}, split height

To obtain a friction velocity profile, we solve Eq. 2.18 for u_τ,

$$u_\tau(y) = \sqrt{\frac{\langle \tau \rangle}{1 - y/\delta}}. \tag{2.19}$$

Figure 2.3 shows the friction velocity profile obtained from Eq. 2.19 using the total shear stress profile corresponding the the tri-split forcing one the Case 2 domain. This approach is essentially "pinning" the simulated total shear stress profile to the theoretical one, i.e. calculating a u_τ value at each wall-normal distance such that the two non-dimensional profiles match. The profile shows how friction velocity varies linearly in the first two off-wall forcing regions then approaches a constant value in the LES outer layer region. The friction velocity is no longer constant near the wall where the shear stresses created by different forcing

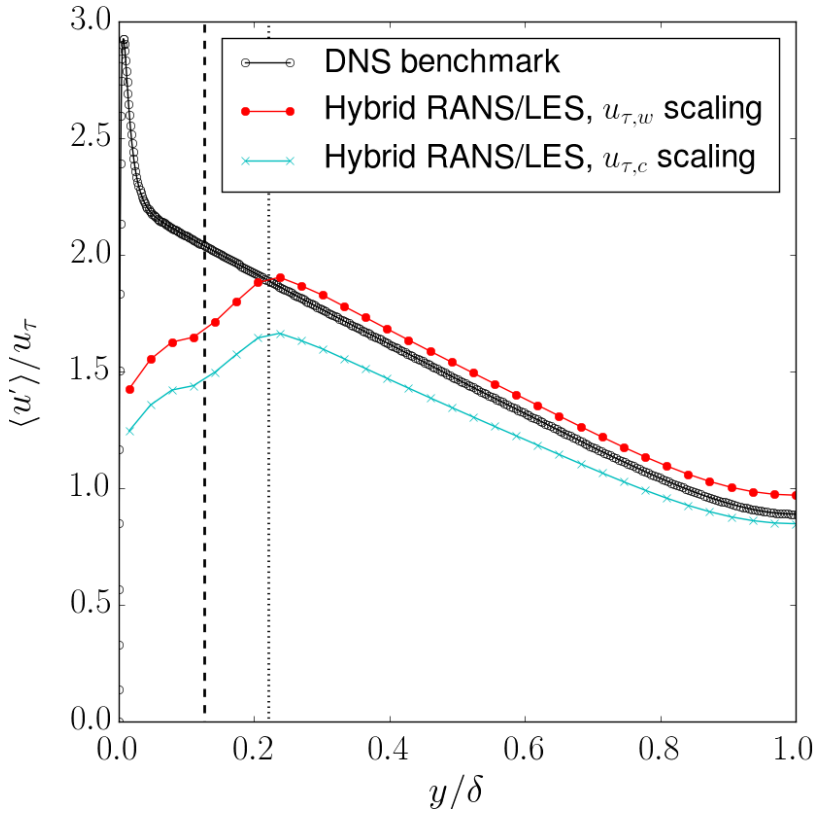


Figure 2.4: Effect of friction velocity scaling on streamwise turbulent intensity for $\mathrm{Re}_\tau = 2000$. Case 2 is simulated. $u_{\tau,w}$ is computed from wall shear stress. $u_{\tau,c}$ is the friction velocity extracted as a constant from the core les region as plotted in Fig. 2.3. Dashed line: h, RANS-LES interface; dotted line: h_{SP}, split height

regions alter the friction velocity value. A constant value is recovered in the core of the flow, i.e. in the LES outer region above $y/\delta = 0.22$ where the results are desired away from the wall. This shows the LES recovers a constant scaling even with wall modeling, however, the scaling is not equal to the one obtained at the wall. As the constant value resides in the core region of the flow, we label this as the core friction velocity, $u_{\tau,c}$. The friction velocity obtained at the wall is $u_{\tau,w}$.

To see the effect of using $u_{\tau,c}$ as the normalization factor, we look at the fluctuation intensities in Figs. 2.4, 2.5, and 2.6. Two profiles for each quantity are shown; one for the normalization with the friction velocity value obtained from the wall shear stress ($u_{\tau,w}$) and one for normalization with the constant scaling factor obtained from the friction velocity

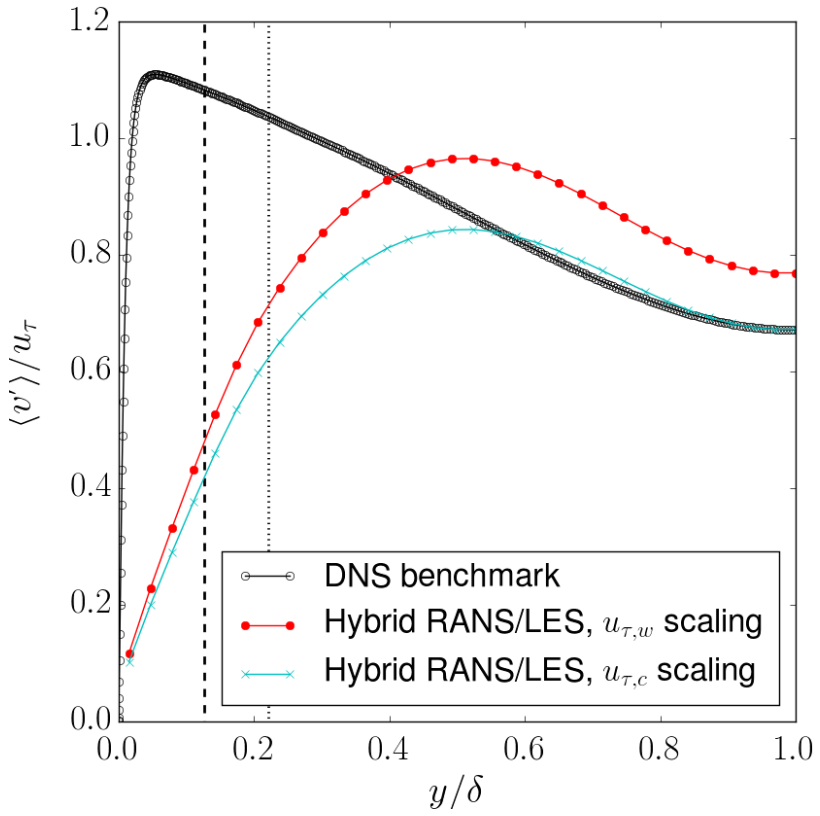


Figure 2.5: Effect of core friction velocity scaling on wall-normal turbulent intensity. Same legend used as in Fig. 2.4.

profile ($u_{\tau,c}$). When using $u_{\tau,w}$, all three profiles overshoot the DNS. But when $u_{\tau,c}$ is used as the normalization factor, the results do not drastically overshoot and show better agreement in the LES outer layer above the split height. The fact that the streamwise intensity undershoots in Fig. 2.4 may be due to the filtering of the LES region. Figure 2.7 shows the modeled and resolved components of shear stress normalized by the friction velocity profile value. Looking at the region where the resolved shear stress dominates the modeled shear stress, normalizing by the friction velocity profile value gives good agreement with the DNS. While not shown, normalizing the resolved shear stress with the wall shear stress would not show good agreement with the DNS. These results indicate that the recovered LES region constant friction velocity is the proper normalizing factor for the turbulence statistics when using the split forcing approach.

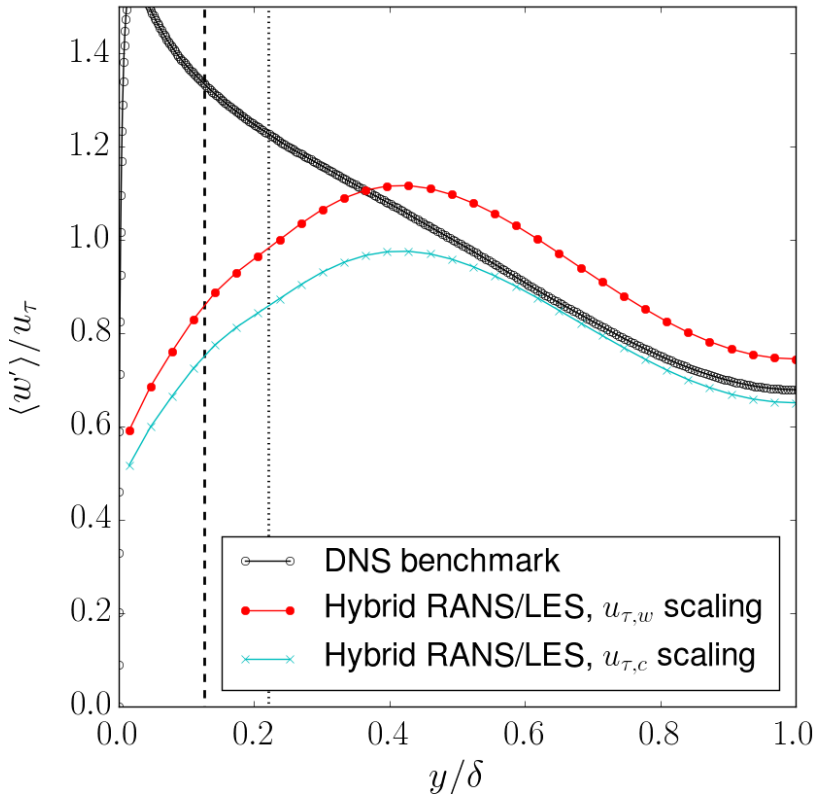


Figure 2.6: Effect of core friction velocity scaling on spanwise turbulent intensity. Same legend used as in Fig. 2.4.

The constant value of the friction velocity in the LES outer layer should not be used to nondimensionalize the mean velocity profile. This makes sense as the mean velocity profile emerges from a balance of the mean pressure gradient and the stresses at the wall. The friction velocity obtained from the core LES region is an appropriate scale for the LES flow field developing over a RANS field. Therefore, turbulence statistics need to be normalized by $u_{\tau,c}$. To emphasize the fact that mean velocity cannot be normalized by the friction velocity profile, we show the dimensionalized mean velocity for the large domain in Case 2 in Fig. 2.8, along with the commonly used single forcing approach. If normalized by anything other than $u_{\tau,w}$ obtained from the wall shear stress, the tri-split forcing profile would miss the target non-dimensional DNS profile in Fig. 2.2. Also note, the mismatch for the single forcing does not look as significant as when plotted on a semi-logarithmic plot, but the mismatch does

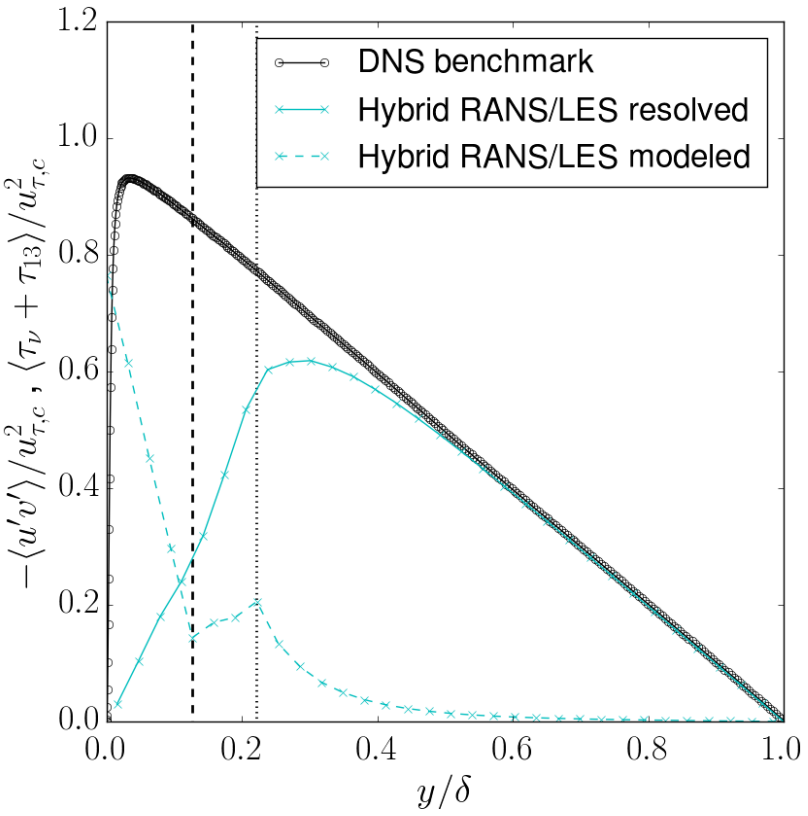


Figure 2.7: Modeled and resolved shear stresses. Obtained from case 2 at $\mathrm{Re}_\tau = 2000$. The friction velocity is obtained from the constant core value in Fig. 2.3. Dashed line: h, RANS-LES interface; dotted line: h_{SP}, split height

indeed exist.

2.5 Conclusions

We showed that the log-layer mismatch that commonly arises from hybrid RANS-LES can be reduced substantially via our proposed split forcing approach. We split the forcing term to drive the flow field in a periodic channel into different regions in a piece-wise fashion based on the eddy viscosity profile. The magnitude of the forcing terms in different regions is determined such that mass flow rate is held constant meaning our proposed split forcing approach only relies on conversation principles and avoids any tunable model parameters. The split forcing also enables the emergence of a unique velocity scale (i.e., friction velocity)

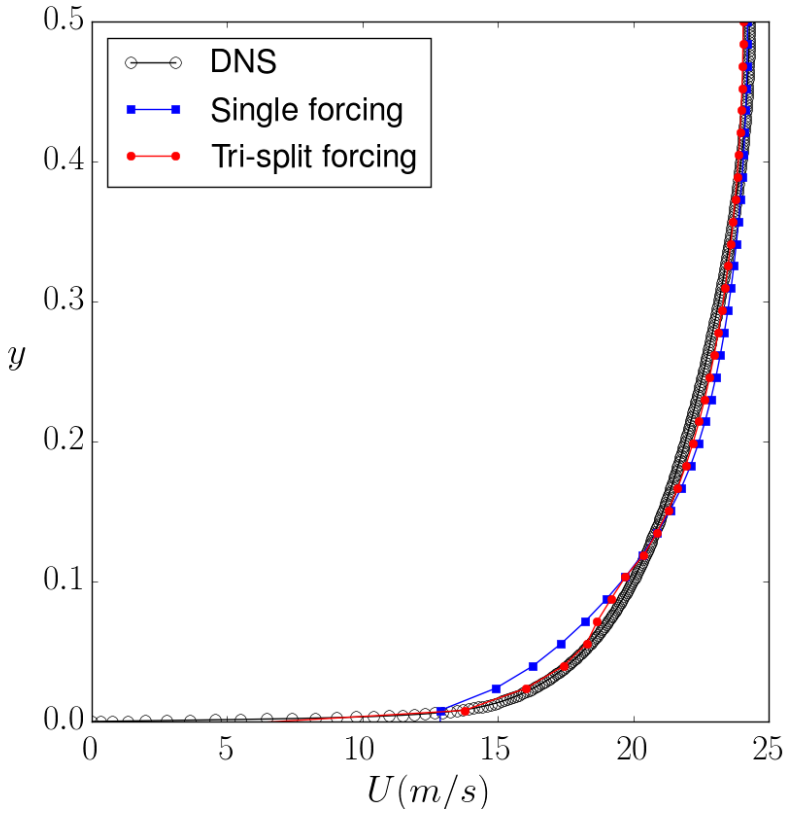


Figure 2.8: Dimensionalized mean velocity. Obtained from case 2 at $\mathrm{Re}_\tau = 2000$ using three forcing regions. DNS profile dimensionalized by using prescribed kinematic viscosity and theoretical friction velocity satisfying $\mathrm{Re}_\tau = 2000$.

in the LES region above the RANS region. When the friction velocity derived from the core LES region is used as the normalizing factor for turbulence statistics, very good agreement with DNS in the LES region is achieved.

We also demonstrated that a basic hybrid RANS-LES model along with a wall-stress boundary conditions can produce results in good agreement with DNS on a mesh that is coarse in the wall-normal direction. This is very encouraging as it would enable the study of very high Reynolds number turbulent flows. In light of these new findings, we suggest the current convention to drive a turbulent periodic channel flow with a uniform mean-pressure gradient term be revisited to test eddy-viscosity-based hybrid RANS-LES models, as the single forcing approach appears to be the culprit behind the log-layer mismatch. Simulating

higher Reynolds number channel flows to further test and validate our split forcing approach will be part of our future work.

CHAPTER 3

Investigation of a Split-forcing Approach to Simulate Wall-modeled Turbulent Channel Flows with Hybrid RANS-LES Methods

3.1 Abstract

We investigate the effect of splitting the constant mean pressure gradient when forcing a periodic turbulent channel flow, simulated by a hybrid RANS-LES approach. We find that splitting the driving pressure gradient into wall-parallel layers while maintaining the same flow rate can reduce the log-layer mismatch problem and recover scaled second-order statistics of turbulence. We simulate turbulent channel flows at friction Reynolds numbers of 2000 and 5200 on coarse meshes that necessitate wall-modeling and demonstrate good agreement with benchmark data. When the mean pressure gradient is split, the LES region develops over the RANS region, attaining its own velocity scale that is different than the friction velocity calculated from the wall shear stress. Second-order turbulence statistics and turbulence spectra agree well with the benchmark data when normalized by the core friction velocity, whereas the friction velocity at the wall remains the appropriate scale for the mean velocity profile. The reason for this duality in velocity scale becomes clear when the mean split-force is integrated across the channel height, confirming a global force balance.

3.2 Introduction

The computational cost of large-eddy simulation (LES) of wall-bounded flows at high Reynolds numbers is expected to remain a challenge in the foreseeable future. The stringent resolution requirements arise from the scales of turbulence diminishing drastically as the wall is approached (Pope, 2000). The computational burden of resolving the near-wall region can be relaxed through *wall-modeled LES* (WMLES) (Deardorff, 1970; Schumann, 1975), in which

the inner part of the boundary layer is modeled with a Reynolds-averaged Navier-Stokes (RANS) representation of turbulence. The resulting method is known as hybrid RANS-LES. In cases where the viscous layer is not resolved, a shear stress boundary condition can replace the no-slip condition at the wall. The shear stress at the wall is derived from flow variables with the help of the law-of-the-wall (Schumann, 1975). There have been numerous studies and proposals to merge RANS and LES calculations in a single simulation.

One of the popular approaches to alleviate computational costs of LES is *detached eddy simulation* (DES), first proposed by Spalart *et al.* (1997); Spalart (2009) for high-Reynolds-number, massively-separated external aerodynamics where governing equations in the thin boundary layer are solved in the RANS sense and LES is used away from the boundary layer where the grid size is adequate for resolving turbulent motions. Nikitin *et al.* (2000) applied DES as a sub-grid scale model in an LES of turbulent channel flow. Turbulence was sustained but there was the appearance of a "DES buffer layer" where too steep velocity gradients created a drastic mean profile mismatch relative to the expected logarithmic velocity profile. The large velocity gradients were deemed to be the result of diminishing eddy viscosity when transitioning from the RANS near-wall flow to LES outer flow. Nikitin et al. observed that turbulent eddies did not form during this transition. Large "super-streaks," or eddies that are unrealistically large, have been observed in the transition region (Piomelli *et al.*, 2003). This issue of an artificial overestimation of the mean velocity profile in the LES region is later coined as the *log-layer mismatch* problem, which is not specific to the DES model and has been reported in other hybrid RANS-LES studies as well.

Hamba (2003) observed the log-layer mismatch when hybridizing RANS and LES with a simple eddy viscosity model, and attributed the mismatch to the discrepancy in filter widths between the RANS and LES regions in the transition zone. Hamba introduced additional filtering to create two separate velocity components at the interface each having a different filter width. The log-layer mismatch was reduced but the problem still persisted.

Both Keating and Piomelli (2006) and Davidson and Billson (2006) introduced a stochastic-

type forcing in the LES-RANS interface region to generate fluctuations that mimick Reynolds stresses. Keating and Piomelli (2006) used a proportional controller to force the flow based on the difference between resolved and modeled Reynolds stress in the transition zone. This approach generates enough Reynolds stress to correct the log layer mismatch and break up the so-called "super-streaks" that occur in transition regions (Piomelli *et al.*, 2003). Radhakrishnan *et al.* (2006) later observed accurate results without stochastic forcing in unstable flows where the mean values tend to have perturbations. Davidson and Billson (2006) introduced turbulent fluctuations that were obtained from synthesized homogeneous turbulence.

Shur *et al.* (2008) proposed an idea to combine the delayed DES (DDES) approach of Spalart *et al.* (2006) for areas where grid resolution prevents resolving dominant eddies and WMLES for grid resolutions that can resolve dominant eddies. The method, referred to as improved DDES (IDDES), blends together these two approaches. IDDES performed just as well in massively separated flows as DES or DDES (the latter improved upon DES by allowing for more grid refinement and thicker boundary layers), and improved upon the log-layer mismatch in plane channel flow. IDDES also performed very well in flows with both attached and separated flow regions.

Instead of merging RANS and LES solutions in a single domain, Balaras and Benocci (1994) proposed a dual-mesh approach that solves the RANS and LES governing equations on two different meshes allowing for each mesh to be refined in the directions important to the respective turbulence simulation method. More recently, Xiao and Jenny (2012) developed a dual-mesh framework for hybrid RANS-LES where relaxation terms are introduced into the LES momentum equations to relax under-resolved LES towards a RANS solution and into the RANS momentum equations to ensure consistency in turbulent kinetic energy between RANS and LES in the well-resolved LES region. Results showed an improvement over using LES on the same mesh.

Abe (2013) proposed a sub-grid-scale model (SGS) combining an isotropic linear eddy viscosity model with an extra anisotropic term obtained from subtracting an eddy viscosity

SGS model from the Bardina scale-similarity model (Bardina *et al.*, 1983). The model agreed well with DNS results at a friction Reynolds number of 395 but started showing mean velocity and Reynolds stress discrepancies at higher friction Reynolds numbers of 1020 and 1200. Davidson (2014) applied the partially-averaged Navier-Stokes (PANS) framework proposed by Girimaji (2006) as a zonal hybrid RANS-LES model, setting the PANS coefficients to unity in the RANS region and a lesser value in the LES region. Results from channel flow and flow over a hump gave good agreement as long as the RANS-LES transition height was sufficiently high, i.e. ≈ 500 wall units. Also found in Piomelli *et al.* (2003) and Davidson and Billson (2006), bringing the RANS-LES interface too close to the wall, say less than 200 wall units, resulted in poor results.

There have been several other proposed hybrid RANS-LES methodologies. For comprehensive reviews, the reader is referred to the works of Fröhlich and von Terzi (2008) and Piomelli (2008). As one can glean from previous work in hybrid RANS-LES, the vast majority of approaches are tested in a turbulent channel flow with periodic boundary conditions with eddy-viscosity-type models and the first off-wall grid point residing in the viscous sublayer. We note that blended eddy viscosity models create a large variation in effective viscosity in the wall-normal direction that permeates well into the LES region. In the simulations the LES core develops over a very viscous RANS region. Formation of an artificial boundary layer between RANS and LES region has been suggested (Baggett, 1998) which acts as a displacement height to the core LES region. We hypothesize that the LES core develops differently than the near-wall region due to this displacement created by large effective viscosity variation. To validate our hypothesis, we drive LES and RANS regions separately with a split-forcing approach and investigate resulting first and second-order statistics of turbulence.

The aforementioned studies reduced the log-layer mismatch by introducing an additional degree of freedom into the global momentum balance, often by introducing an empirical forcing term into the governing equations that mimics resolved shear stresses or by altering

the mixing length to accommodate lack of resolved shear stress. Based on our hypothesis, we introduce an additional degree of freedom to the momentum balance by splitting the mean pressure gradient. The splitting is accomplished by introducing zonal layers across the channel. The driving mean pressure gradient takes on a different value in each zonal layer in a way to keep the mass flow rate constant. Forcing each zone separately allows shear stresses to emerge naturally between RANS and LES regions. Hereinafter, we refer to the driving mean pressure gradient that drives a channel flow with periodic boundary conditions as *forcing* and our approach as *split-forcing*.

In most hybrid RANS-LES methods, as those reviewed above, the greatest computational cost savings often come from coarsening the resolution in the streamwise and spanwise direction. The first off-wall grid point is commonly placed in the viscous sublayer to avoid the prescription of wall-stress as boundary conditions. A distinguishing aspect of our present work is the coarse resolution used in the wall-normal direction, in addition to the coarse resolution in the spanwise and streamwise directions, which results in significant savings in computational cost of performing high Reynolds turbulent flows.

3.3 Numerical Formulation

We solve the following filtered form of the governing equations for incompressible flows.

$$\frac{\partial \overline{u}_j}{\partial x_j} = 0, \tag{3.1}$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u}_i \overline{u}_j\right) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\nu \overline{S}_{ij} - \tau_{ij}\right) + f_i, \tag{3.2}$$

where

$$S_{ij} = \frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right) \tag{3.3}$$

is the strain rate tensor, and

$$\tau_{ij} = \overline{u_i u_j} - \overline{u}_i \overline{u}_j \tag{3.4}$$

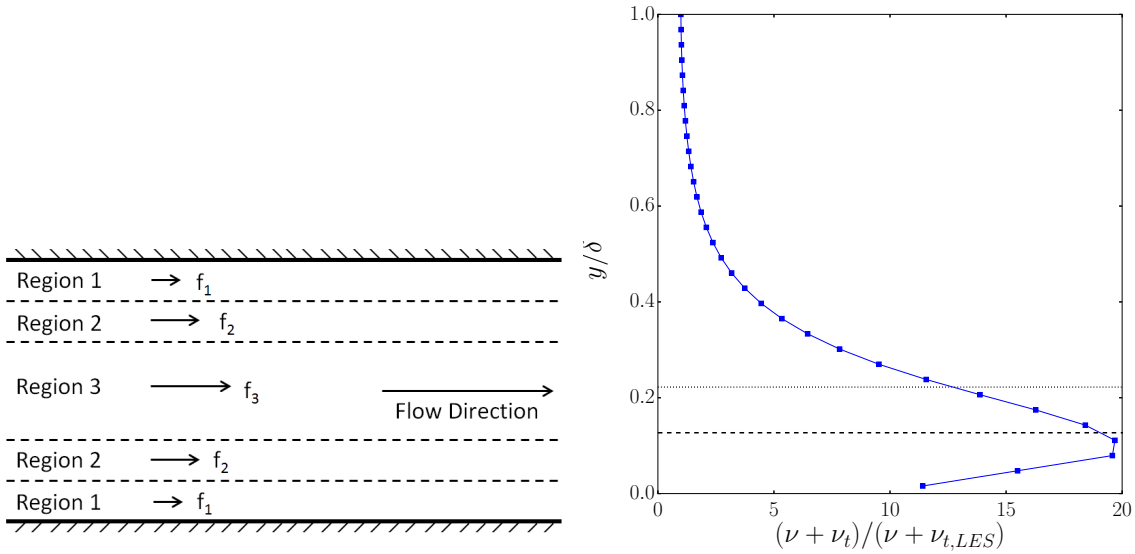
is the tensor for the SGS Reynold stresses, and f_i is a body forcing term. For a fully-developed channel flow, f_1 represents the mean streamwise pressure gradient to drive the flow in the x direction. For the spanwise and wall-normal directions, respectively, f_2 and f_3 are zero.

The governing equations are solved using a GPU-accelerated, three dimensional incompressible flow solver (Thibault and Senocak, 2012; Jacobsen and Senocak, 2013; DeLeon *et al.*, 2013). The code uses the projection algorithm (Chorin, 1968) on directionally-uniform Cartesian grids with second-order central differences for spatial derivatives and a second-order Adams-Bashforth scheme for time advancement. The pressure Poisson equation is solved by an amalgamated parallel 3D geometric multigrid solver designed for GPU clusters, specifically (Jacobsen and Senocak, 2011).

3.3.1 The Mean Pressure Gradient in a Periodic Channel Flow

As periodic boundary conditions alone cannot sustain a constant mass flow rate in the channel, f_1, referred to as f_x hereinafter, has to be introduced as a mean streamwise pressure gradient in the streamwise momentum equation to drive the flow field. There are two approaches to impose a mean pressure gradient: prescribe a constant body force that would balance the target shear stress at the wall, or dynamically adjust a uniform body force to maintain a target mass flow rate through the channel. We adopt the latter approach, which is preferable when the wall shear stress is not known *a priori*.

In our study we force the flow field through the channel by keeping a constant mass flow rate. The particular method chosen for this study is that of Benocci and Pinelli (1990),



(a) (b)

Figure 3.1: Split-forcing demarcations. (a) Sketch of three forcing regions in a channel flow. (b) The normalized mean eddy viscosity profile obtained from a hybrid RANS-LES simulation of $\mathrm{Re}_\tau = 2000$ turbulent channel flow. The dashed line represents the RANS-LES blending height, h_{RL}, and the first split-forcing demarcation. The dotted line represents the second split-forcing demarcation, h_{sp}.

which can be formulated as follows

$$f_x^{t+1} = f_x^t - \frac{2}{\Delta t}\left(\frac{\dot{m}^{t+1}}{A_c} - \frac{\dot{m}^0}{A_c}\right) + \frac{1}{\Delta t}\left(\frac{\dot{m}^t}{A_c} - \frac{\dot{m}^0}{A_c}\right), \tag{3.5}$$

where the superscript t is the time step level, Δt is the physical time step value (which needs to be held constant throughout the simulation as per the original formulation), $\dot{m}^0$ is the prescribed mass flow rate, and A_c is the constant cross-sectional area. Other formulations to maintain constant mass flow rate can be found in the literature (You *et al.*, 2000; Brandt, 2003).

3.3.2 Split-forcing Technique

The log-layer mismatch observed in turbulent channel flow simulations with eddy-viscosity-based hybrid RANS-LES methods has been attributed to a lack of resolved Reynolds shear stress provided to the LES zone near the RANS-LES interface (Davidson and Billson, 2006; Keating and Piomelli, 2006; Larsson *et al.*, 2007). Here, we revisit how the log-layer mismatch arises in eddy-viscosity-based hybrid RANS-LES methods of channel flows and propose our split-forcing technique to reduce it. Our explanation pertains to blended eddy viscosity models, but several other explanations and solutions for the log-layer mismatch can be found in literature Davidson and Billson (2006); Larsson *et al.* (2007); Shur *et al.* (2008); Hu (2013).

For a fully-developed, statistically-stationary channel flow, the mean streamwise momentum equation reduces to the following

$$\frac{d\tau}{dy} = -\frac{dp}{dx}. \tag{3.6}$$

The driving mean pressure gradient is constant on the right hand side of Eq. 3.6 (Pope, 2000). Integrating Eq. 3.6 from the wall ($y = 0$) to some wall-normal distance y and solving for τ_w, we obtain,

$$\tau_w = \underbrace{\frac{dp}{dx}y}_{\text{I}} + \underbrace{\nu\frac{dU}{dy}\bigg|_y}_{\text{II}} + \underbrace{\nu_t(y)\frac{dU}{dy}\bigg|_y}_{\text{III}} - \underbrace{\langle uv\rangle|_y}_{\text{IV}}, \tag{3.7}$$

where term I represents the driving pressure gradient, term II represents the viscous shear stress, term III is the shear stress modeled with an eddy-viscosity approach, and term IV represents the resolved Reynolds shear stress. In a wall-resolved LES at a wall-normal distance sufficient to ignore viscous effects (i.e. ignore term II), we can argue that Eq. 3.7

has two degrees of freedom to achieve force balance through terms III and IV as $\frac{dp}{dx}$ is constant ($\frac{dp}{dx}$ is f_x in Eq. 3.5).

We now evaluate Eq. 3.7 when a hybrid RANS-LES model is adopted in which wall-normal variation of the eddy viscosity is significant, taking on a large value in the RANS region and a small value in the LES region. We place the RANS-LES transition height sufficiently away from the wall to ignore term II. In theory, the RANS turbulence model simulates the ensemble averaged field, thus approaching the interface from the RANS side means negligible resolved stresses, therefore term IV can be ignored within the RANS region. Now the previous two degrees of freedom in a wall-resolved LES (terms III and IV) are reduced to one in a hybrid RANS-LES (term III). Consequently, the mean velocity profile will conform to unexpected profiles to maintain the force balance in Eq. 3.7.

As numerous studies have shown, introducing a second degree of freedom through an empirical forcing term around the RANS-LES interface reduces the log-layer mismatch markedly. However, we note that when using eddy viscosity models, the molecular viscosity and eddy viscosity are combined to create an *effective viscosity* in the shear stress term in the governing momentum equations. This effective viscosity is significantly larger when blending RANS and LES eddy viscosity models as opposed to sub-grid-scale viscosity in a wall-resolved LES. Loosely speaking, we can imagine a less viscous fluid flowing over a highly viscous fluid with variable viscosity. Based on this analogy, we sought to alter the spatially-uniform mean streamwise pressure gradient to drive a channel flow and split the mean streamwise pressure gradient, hereinafter referred to as *forcing*, into wall-parallel layers in an approach dubbed *split-forcing*. The hypothesis is that split-forcing will reduce the log-layer mismatch in hybrid RANS-LES of channel flow by introducing the missing second degree of freedom (term IV in Eq. 3.7) and propose to force flow regions affected by different effective viscosities with different pressure gradients, while keeping mass flow rate unchanged.

We sketched the split-forcing approach in Fig. 3.1a applied to a channel flow. Each region

of the flow, referred to as a forcing region, is driven by a constant forcing value different than in the other regions in one channel half-height and reflected about the centerline to enforce symmetry. Figure 3.1a shows three streamwise forcing regions, f_1, f_2 and f_3 (note the $_x$ script was dropped). Each forcing region has its own target mass flow rate such that the mass flow rate for the entire channel remains unchanged. We adjust Eq. 3.5 by introducing a subscript, n, as an index for the forcing region as

$$f_n^{t+1} = f_n^t - \frac{2}{\Delta t}\left(\frac{\dot{m}_n^{t+1}}{A_c} - \frac{\dot{m}_n^0}{A_c}\right) + \frac{1}{\Delta t}\left(\frac{\dot{m}_n^t}{A_c} - \frac{\dot{m}_n^0}{A_c}\right), \tag{3.8}$$

where $\dot{m}_n^t$ now represents the mass flow rate through a particular forcing region. We prescribe the target mass flow rate for each forcing region following a target mean profile, usually obtained from benchmark data. Target mass flow rate for the channel is ensured by the following constraint,

$$\dot{m}_T^0 = \sum_{n=1}^{N} \dot{m}_n^0, \tag{3.9}$$

where $\dot{m}_T^0$ is the total mass flow rate through the channel and N is the total number of forcing regions.

To determine the boundaries of the forcing regions, we inspect the mean eddy viscosity profile from a hybrid RANS-LES simulation using a spatially-uniform forcing that we refer to as *single forcing*. Note that we do not deploy any remedies to reduce the log-layer mismatch in single forcing. Fig. 3.1b shows the mean eddy viscosity profile from a single-forcing hybrid RANS-LES on $\mathrm{Re}_\tau = 2000$ channel flow with simulation parameters given in Table 3.1. The dashed line in Fig. 3.1b is the RANS-LES interface. The mean eddy viscosity is normalized by the value in the core of the flow, which is nearly constant, i.e. determined solely by the LES sub-grid model.

Maximum value of eddy viscosity occurs near the RANS-LES interface, h_{RL}, and is a logical place for a split-forcing demarcation. We initially attempted using only two forcing

regions but the results did not show satisfactory improvement because the eddy viscosity is still large above the RANS-LES interface relative to the LES value. A second splitting height, h_{sp}, is placed in the large eddy viscosity region approximately where the eddy viscosity is 10 to 12 times that of the core value. The hypothesis was that one more additional forcing region will introduce enough shear stress in the blending region to provide satisfactory results. The use of three forcing regions is referred to as *tri-split*. We use approximately the same splitting heights in nondimensional wall units for the $\mathrm{Re}_\tau = 5200$ case (see Tab. 3.1). Note that using more forcing regions than three may not be necessary as the goal of this experiment is to force regions of high eddy viscosity independent of the LES core where the eddy viscosity profile is nearly constant. Also, when eddy viscosity is nearly uniform, splitting the mean pressure gradient would run against the theory described by Eq.3.7.

3.3.3 Turbulence Modeling

We adopt the localized Lagrangian dynamic Smagorinsky eddy viscosity model (Meneveau *et al.*, 1996) for LES. In the RANS region, we use the mixing length model of Prandtl (1925) as it is an adequate model for attached flows. Hybridization of RANS and LES is often accomplished by blending either SGS stress terms, eddy viscosities, or turbulent length scales (Sagaut, 2002; Fröhlich and von Terzi, 2008). In this study, we follow the work of Senocak *et al.* (2007) and blend only the length scales in the eddy viscosity formula.

$$l_{mix} = [1 - \exp(-y/h_{RL})]C_S\Delta + \exp(-y/h_{RL})\,\kappa y, \tag{3.10}$$

$$\nu_t = l_{mix}^2|\overline{S}|, \tag{3.11}$$

$$\tau_{ij} = \nu_t S_{ij}, \tag{3.12}$$

where h_{RL} is the RANS-LES transition height. Note that we blend the length scales instead of blending the squared values as was originally proposed in Senocak et al. By blending only the length scale instead of stresses or eddy viscosity, our approach essentially becomes

similar to that of DES.

3.3.4 Wall Boundary Conditions

As we wish to place the first off-wall grid point in the logarithmic region ($y^+ > 30$) to mimic a resolution situation in LES of ABL in practice, we obtain the wall shear stress through the wall model of Schumann (1975), later modified to a more general form by Grötzbach (1987). Moeng (1984) extended these wall shear stress boundary conditions to simulate ABL and are still commonly used to this day. The essence of the Schumann model is to provide an instantaneous wall shear stress boundary condition by providing components of the total wall shear stress and explicitly enforcing the impermeability condition via

$$\overline{\tau}_{xy,w}\left(x,z,t\right) = \frac{\overline{u}\left(x,y_1,z,t\right)}{\langle\overline{u}\left(y_1\right)\rangle}\langle\overline{\tau}_{xy,w}\rangle, \tag{3.13}$$

$$\overline{v} = 0, \tag{3.14}$$

$$\overline{\tau}_{zy,w}\left(x,z,t\right) = \frac{\overline{w}\left(x,y_1,z,t\right)}{\langle\overline{u}\left(y_1\right)\rangle}\langle\overline{\tau}_{xy,w}\rangle, \tag{3.15}$$

where y_1 is the first off-wall grid point. Schumann originally obtained the mean wall shear stress from the momentum balance in a channel flow, $\langle\overline{\tau}_{xy,w}\rangle = -\delta\langle\frac{\partial\overline{p}}{\partial x}\rangle$, where $\langle\frac{\partial\overline{p}}{\partial x}\rangle$ was given *a priori*. Grötzbach generalized this wall model by determining the stress from the log-law:

$$\frac{\langle\overline{u}\left(y_1\right)\rangle}{u_\tau} = \frac{1}{\kappa}\ln\left(\frac{y_1 u_\tau}{\nu}\right) + B, \tag{3.16}$$

$$\langle\overline{\tau}_{xy,w}\rangle = \rho u_\tau^2, \tag{3.17}$$

where κ is the von Kármán constant, B is a constant, and ρ is the density, set to unity in this study. While eliminating the need for an *a priori* $\langle\frac{\partial\overline{p}}{\partial x}\rangle$, Grötzbach's modification requires knowledge of κ and B. For this study, we adopt the commonly used values of $\kappa = 0.41$ and $B = 5.2$, but various other possible values exist (Pope, 2000; Nagib and Chauhan, 2008; Lee

Table 3.1: Simulation parameters. The superscript, $^+$, indicates nondimensional quantities in wall units. y_1^+ is the distance of the first off-wall streamwise velocity component.

Re_τ	N_x	N_y	N_z	L_x	L_y	L_z	Δx^+	Δy^+	Δz^+	y_1^+
2000	257	65	193	$8\pi\delta$	$2\ \delta$	$3\ \pi\delta$	200	64	100	32
5200	513	129	513	$8\pi\delta$	$2\ \delta$	$3\ \pi\delta$	250	82	100	41

Table 3.2: Split-forcing parameters. The superscript, $^+$, indicates nondimensional quantities in wall units. h_{RL} is the RANS-LES transition height and h_{sp} is the splitting height applicable to tri-split forcing.

Re_τ	h_{RL}^+	h_{sp}^+	$h_{RL}\ /\ \delta$	h_{sp}/δ
2000	250	440	0.127	0.222
5200	250	490	0.047	0.094

and Moser, 2015).

To avoid contamination from discretization and SGS modeling errors near the wall (Cabot and Moin, 2000; Nicoud *et al.*, 2001; Kawai and Larsson, 2012), Eq. 3.16 is solved at the third off-wall grid point at every time step such that y_1 becomes y_3 (Lee *et al.*, 2013). The strain rate tensor in Eq. 3.11 requires the computation of wall-normal derivatives and will be incorrect due to the coarseness of the grid if calculated explicitly. Therefore, the eddy viscosity in the first off-wall grid cell is computed via linear extrapolation from the second and third off-wall grid cells (Cabot and Moin, 2000). Also, as friction velocity cannot be specified for a general case, we believe the true performance of wall-modeled LES comes about when surface stresses (i.e. friction velocity) are computed in-situ from the simulated velocity field. We note that not every study in literature respects this subtle issue.

3.4 Results and Discussion

Tables 3.1 and 3.2 present the simulation parameters and the split-forcing parameters for each turbulent channel flow case, respectively. The friction Reynolds number is based on friction velocity, u_τ, and channel half-height, δ. We compare $\mathrm{Re}_\tau = 2000$ channel flow to the readily-available DNS data of Hoyas and Jiménez (2006), and $\mathrm{Re}_\tau = 5200$, to the recent DNS

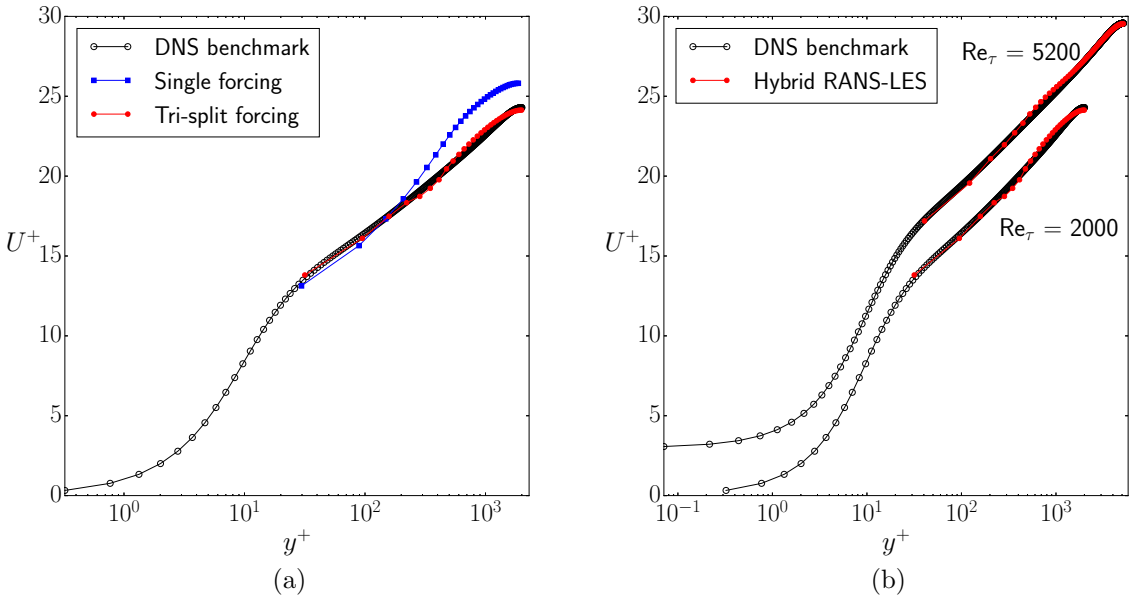


Figure 3.2: Nondimensional mean velocity profiles. (a) Log-layer mismatch reproduced using the single forcing approach. $\mathrm{Re}_\tau = 2000$. (b) $\mathrm{Re}_\tau = 5200$ (shifted up by 3.0) and $\mathrm{Re}_\tau = 2000$ simulated with the tri-split approach. All results are normalized by $u_{\tau,w}$ obtained from the wall shear stress.

data of Lee and Moser (2015). In all of our simulations, periodic boundary conditions were applied in the streamwise and spanwise directions (x-direction and z-direction, respectively). Spin-up time for turbulence was 200 δ/u_τ and sampling for turbulence statistics took place over an additional 200 δ/u_τ.

The effect of split-forcing on the mean velocity profile is shown in Fig. 3.2a for $\mathrm{Re}_\tau = 2000$. The single forcing of the channel flow reproduces the well-known log-layer mismatch problem. The tri-split forcing significantly reduces the mismatch from the DNS mean velocity, demonstrating a clear improvement over the single forcing approach. The effect of split-forcing is to introduce shear stresses into the flow to compensate for the lack of resolved shear stress in the RANS-LES interface region and enable the mean velocity field to match the expected profile. In Fig. 3.2b we demonstrate the effectiveness of our tri-split forcing approach for an even higher Reynolds number flow, $\mathrm{Re}_\tau = 5200$. The corresponding bulk Reynolds number (based on channel height) for $\mathrm{Re}_\tau = 5200$ is approximately 250,000.

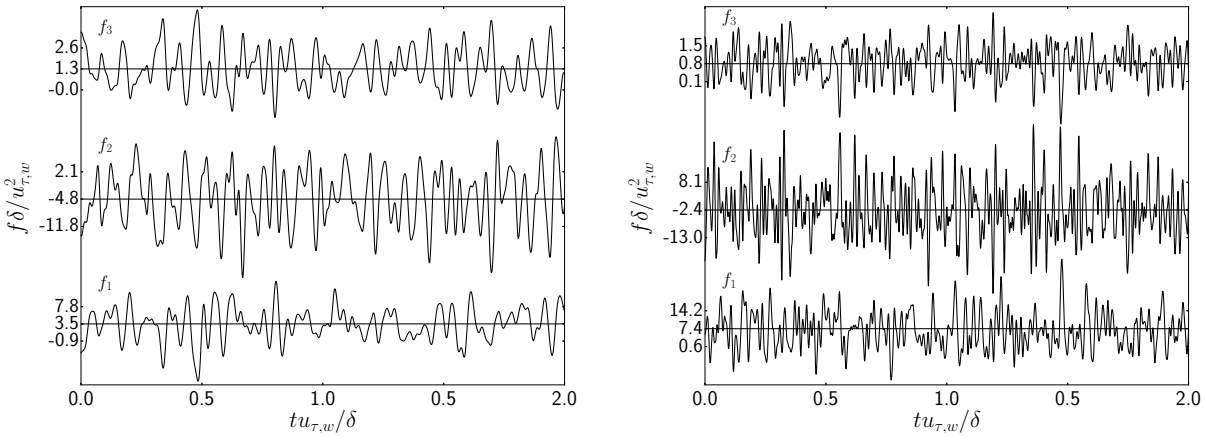


Figure 3.3: Tri-split forcing time series for (a) $\mathrm{Re}_\tau = 2000$ and (b) $\mathrm{Re}_\tau = 5200$ normalized by δ and $u_{\tau,w}$. Results are shown over two eddy turnover times, $\delta/u_{\tau,w}$. The bottom time series in the figure is the near-wall region and the topmost time series is for the core forcing region. The solid horizontal line is the mean of the time series found over the entire statistical sampling period. The other two tickmarks are one standard deviation above and below the mean found over entire statistical sampling period.

Both profiles agree very well with the DNS benchmark data despite a mesh resolution that is coarse in the wall normal direction ($y_1^+ \approx 30$). For context, Lee and Moser (2015) used roughly 121 billion mesh points for DNS of $\mathrm{Re}_\tau = 5200$ whereas we used 33.9 million mesh points without any stretching in the mesh.

Figures 3.3a and 3.3b show a time series of the forcing values for the tri-split for for both $\mathrm{Re}_\tau = 2000$ and $\mathrm{Re}_\tau = 5200$, respectively, over a time lapse of two eddy turnover times, $\delta/u_{\tau,w}$. We plot the evolution of the forcing in each zone to better understand the impact of split-forcing on the flow field. As will be discussed later, the wall shear stress can be recovered by converting the mean forcing values to a shear stress by integrating along channel height. The time series shows that the near-wall region and the middle forcing region have much greater variations than the core forcing region, with the middle forcing region being predominantly negative in value while the other two regions are positive in value. The magnitude of the eddy viscosity explains this behavior. In the RANS region, the eddy viscosity is dominantly defined by the mixing length model, exhibiting a positive gradient.

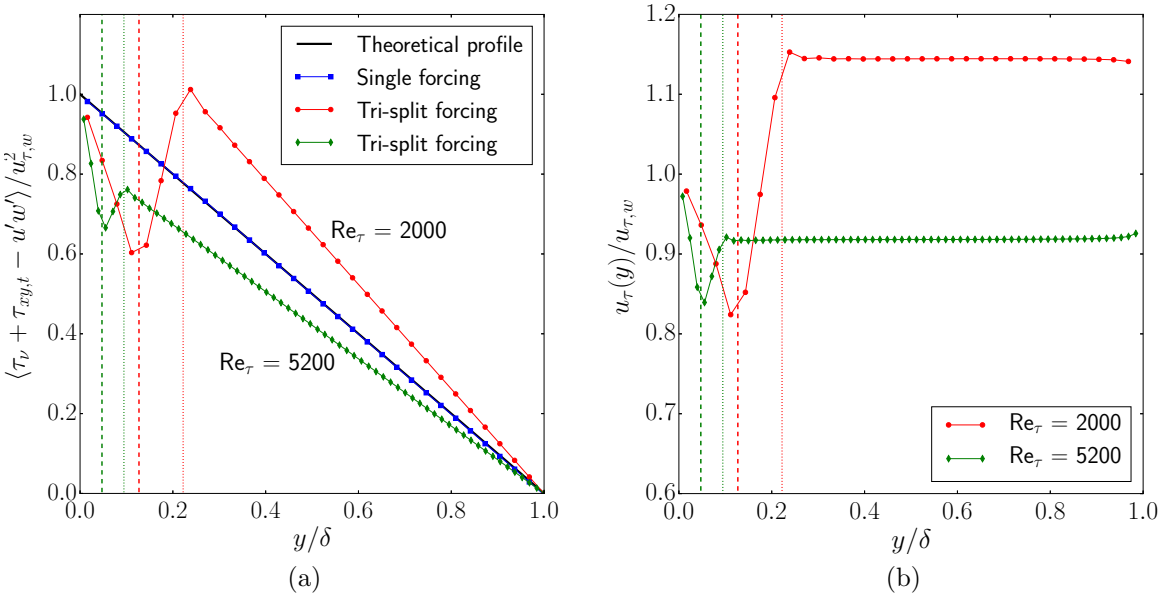


Figure 3.4: (a) Nondimensional total shear stress distribution for $\mathrm{Re}_\tau = 2000$ and 5200 using single and tri-split forcing, normalized by the wall friction velocity $u_{\tau,w}$. Note that the mismatch with theoretical profile disappears when normalized by the core friction velocity. (b) Profiles of normalized friction velocity calculated from Eq. 3.18 using tri-split forcing at both $\mathrm{Re}_\tau = 2000$ and $\mathrm{Re}_\tau = 5200$. Dashed vertical lines, h_{RL}; dotted vertical lines, h_{sp}.

The core forcing region, above h_{sp}, the eddy viscosity is dominantly defined by the sub-grid scale model and does not vary significantly. However, the second forcing region between h_{RL} and h_{sp} has a negative gradient in eddy viscosity, as the model is decreasing from a RANS value to an LES value, leading to the middle forcing region being predominantly negative in value. The opposite signs of the forcing help energize the interface between each zone stochastically.

Analyzing the total shear stress profiles generated by the split-forcing produced an anomaly, which has led us to discover a duality of the friction velocity when using split-forcing. Note that we are not suggesting a duality in velocity scale as part of the physics of turbulent channel flows. The observed duality is a consequence of applying split-forcing to hybrid RANS-LES. When the total shear stress profile is normalized by the wall friction velocity, $u_{\tau,w}$, a significant mismatch between tri-split forcing and the theoretical linear pro-

file exists as shown in Fig. 3.4a. However, simulations do indicate a statistically stationary state. Therefore, something must be amiss. Also of note here is that the profiles in the LES region exhibit a linear variation albeit a wrong slope.

Because a statistically stationary state has been reached, the total shear stress profile should follow the expected theory, which led us to think that a different velocity scale must be in play when the LES region is forced above a RANS region affected by a high effective viscosity. Therefore, we probe the wall-normal variation of the friction velocity as each region may develop a different scale. Tuerke and Jiménez (2013) used a friction velocity profile, which they termed a "local" friction velocity, to collapse Reynolds stress profiles in their inverse RANS (IRANS) study where mean velocity profiles were prescribed as a forcing term to a DNS. We follow the same approach and calculate a local friction velocity profile as follows

$$u_\tau(y) = \sqrt{\frac{\langle \tau(y) \rangle}{1 - y/\delta}}. \tag{3.18}$$

Equation 3.18 is derived by solving the theoretical total shear stress profile in a channel flow Pope (2000) (the theoretical profile is shown in Fig. 3.4a),

$$\langle \tau(y) \rangle = u_\tau^2 \, (1 - y/\delta) \tag{3.19}$$

for u_τ except that u_τ becomes a function of wall-normal distance, y, to satisfy the linear profile shown in Fig. 3.4a. The variation of the friction velocity along the wall normal distance is shown in Fig. 3.4b for Re_τ 2000 and 5200. A consistent feature of both friction velocity profiles is that a constant value is recovered above the split height, h_{sp}, for each Reynolds number case where the LES SGS model is active. We refer to the constant value in the core as $u_{\tau,c}$.

We study the use of $u_{\tau,c}$ by assessing the second-order statistics of turbulence. In Figs. 3.5a and 3.5b, we compare normalized RMS velocity fluctuations with the scaling factors

of $u_{\tau,c}$ and $u_{\tau,w}$. For $\mathrm{Re}_\tau = 2000$ in Fig. 3.5a, the normalization by $u_{\tau,c}$ shows a clear improvement over scaling by $u_{\tau,w}$ for the spanwise and wall-normal components, matching the DNS benchmark data very well. The agreement with the streamwise component u'_{rms} is slightly worse. However, the u'_{rms} profile normalized by $u_{\tau,c}$ does not overestimate the DNS data, which makes sense as this is a filtered field obtained from a coarse mesh. Also for Re_τ =5200, u'_{rms} results scaled by $u_{\tau,c}$ are closer to the DNS normalized values than they are when scaled by $u_{\tau,w}$. Turbulent motions are damped by the high RANS eddy viscosity as expected, therefore we do not seek any agreement with DNS data near the wall and use only $u_{\tau,c}$ as the scaling, and not the entire local friction velocity profile. Relative to the streamwise component, the spanwise and wall-normal components of the RMS velocity fluctuations take some distance past h_{sp} before agreeing with the DNS benchmark data in the core region of the channel, suggesting that the three-dimensionality of turbulence emerges at some distance away from the RANS-LES interface. The lack of agreement below y/δ <0.4 for $\mathrm{Re}_\tau = 2000$ and y/δ <0.2 for $\mathrm{Re}_\tau = 5200$ is mainly an outcome of the RANS-LES model at hand. A sophisticated RANS model might perform better closer to the wall. We note that our focus in this work is not to develop a new SGS model, but instead investigate the effect of splitting the mean pressure gradient on hybrid RANS-LES simulations.

From the profiles of nondimensional mean velocity, total shear stress, and turbulent fluctuation intensities, we observe two distinct velocity scales to nondimensionalize these quantities: $u_{\tau,w}$ and $u_{\tau,c}$. We reiterate that duality in velocity scale is an artifact of the split-forcing approach and does not come from the actual flow physics. The only aspect that is consistent with expected flow physics is that $u_{\tau,c}$ attains a constant profile. It also makes sense to use $u_{\tau,c}$ to scale the fluctuations in the LES region, as this region develops above a RANS region with its own forcing. But a worthy question comes up: Why is $u_{\tau,w}$ a better velocity scale for the mean velocity profile? To answer this question we further analyze the force balance in the mean sense.

As the split regions evolve under the constraint of maintaining the target mass flow rate,

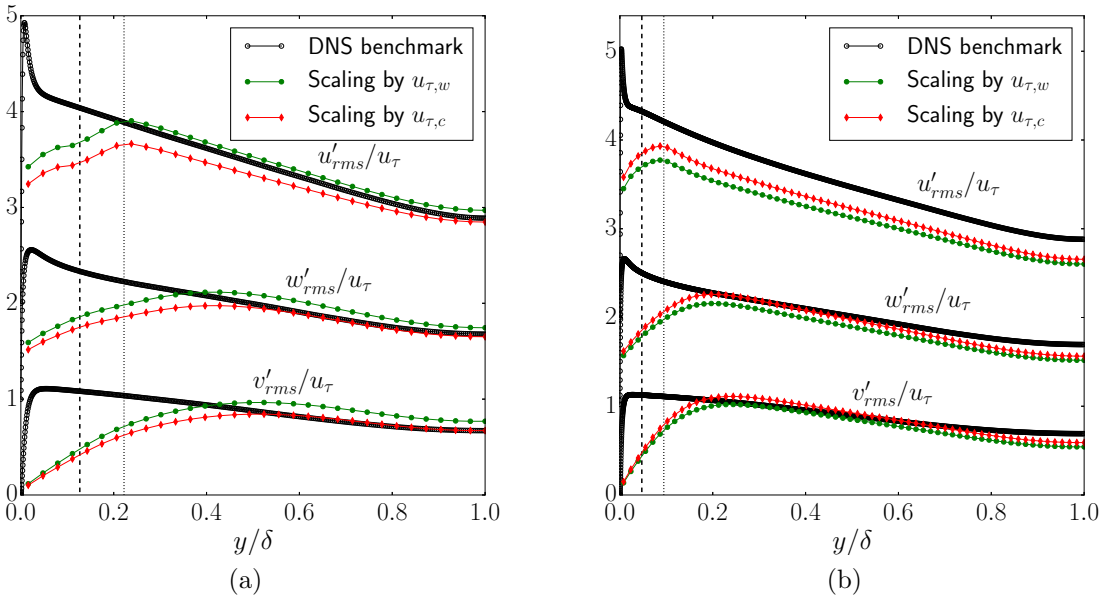


Figure 3.5: Profiles of rms velocity fluctuations and the effect of velocity scale choice on the results. (a) $\mathrm{Re}_\tau = 2000$ and (b) $\mathrm{Re}_\tau = 5200$. Streamwise and spanwise components shifted up by 2.0 and 1.0, respectively. Dashed vertical lines, h_{RL}; dotted vertical lines, h_{sp}.

Table 3.3: Normalized values for verification of Eq. 3.20. Each subscript represents a forcing region. All forcing values are normalized by $u_{\tau,w}$ and δ. All heights are normalized by δ. Mean forcing values can be found in Figs. 3.3a and 3.3b.

Re_τ	h_1	h_2	h_3	$< f_1 >$	$< f_2 >$	$< f_3 >$	$\sum_{i=1}^{3} < f_i > h_i$
2000	0.127	0.095	0.778	3.502	-4.873	1.312	1.001
5200	0.047	0.047	0.906	7.496	-2.445	0.841	1.000

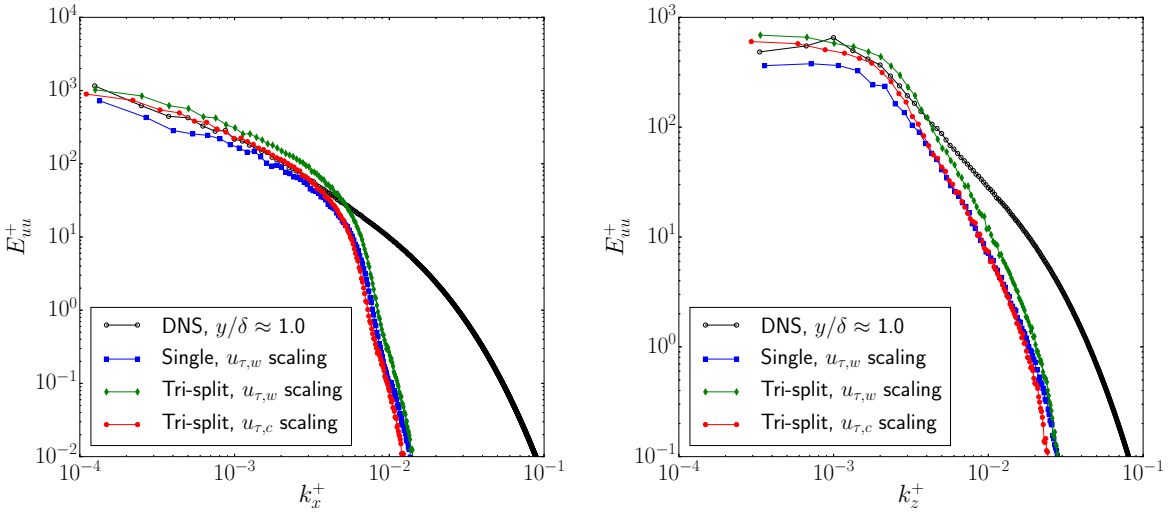


Figure 3.6: Streamwise velocity spectra. Streamwise velocity spectra for $\mathrm{Re}_\tau = 2000$ (a) in streamwise direction (b) in spanwise direction.

a global force balance is satisfied as follows

$$\langle f_1 \rangle h_1 + \langle f_2 \rangle h_2 + \langle f_3 \rangle h_3) = \tau_w = \frac{u^2_{\tau,w}}{\delta}\delta = f_x \delta, \tag{3.20}$$

where $h_1 = h_{RL}$, $h_2 = h_{sp} - h_{RL}$, and $h_3 = \delta - h_{sp}$. From the above relationship, we clearly see that $u_{\tau,w}$ is an average velocity scale for the entire channel flow. Table 3.3 presents the normalized values that can be used to verify the above equation. Therefore, it is not surprising that $u_{\tau,w}$ is still the adequate velocity scale to normalize the mean velocity profile across the channel. It is also evident why we cannot use $u_{\tau,c}$ to scale the entire velocity profile, because it does not satisfy the global force balance.

Split-forcing allows different regions of the flow to develop semi-independently, and it is not unreasonable to expect a different velocity scale to emerge in the core LES region. One-dimensional normalized streamwise velocity spectra as shown in Fig.3.6 further supports our arguments about duality in velocity scale. Fig.3.6 presents E^+_{uu}, for $\mathrm{Re}_\tau = 2000$ at the channel centerline for both streamwise and spanwise directions. The simulations capture

Table 3.4: Comparison of computed skin friction coefficient with Dean's correlation of Dean (1978). Re_b is bulk Reynolds number. Computed skin friction coefficient: $C_{f,comp} = \tau_w / \frac{1}{2}\rho U_b^2$. Dean's correlation for skin friction coefficient $C_{f,Dean} = 0.073 Re_b^{-0.25}$.

Re_τ	Forcing	U_b / $u_{\tau,w}$	Re_b	$C_{f,comp}$	$C_{f,Dean}$	% error
2000	Single	23.44	87793	3.64×10^{-3}	4.24×10^{-3}	14.1
2000	Tri-split	21.99	87698	4.14×10^{-3}	4.24×10^{-3}	2.5
5200	Tri-split	24.26	250543	3.40×10^{-3}	3.26×10^{-3}	4.1

well the large eddies as well as a portion of the inertial subrange at the lower wavenumbers. Note that it is not realistic to expect capturing the spectra at higher wave numbers because of the coarse mesh at hand. The simulation with tri-split forcing when scaled with $u_{\tau,c}$ agrees better with the normalized DNS spectra than both the single forcing and the tri-split forcing when scaled with $u_{\tau,w}$.

Early studies of hybrid RANS-LES pointed out the formation of so-called "super-streaks" near the RANS-LES interface (Piomelli *et al.*, 2003). This modeling artifact has been alleviated in several studies (Keating and Piomelli, 2006; Davidson and Billson, 2006). We observe that our split-forcing approach is able to break up these "super-streaks" as shown in Fig. 3.7 where y / $\delta \approx 0.13$ - 0.18. The coarseness of the current mesh does not accommodate formation of physical streaks, but the ability for the split-forcing to reduce the size of the unphysical streaks relative to single forcing simulation is promising. Away from the RANS-LES interface, the tri-split forcing and the single forcing produced similar flow structures.

Tri-split forcing also improves skin friction estimates relative to the single forcing simulation. Table 3.4 compares the skin friction coefficient, C_f, computed from the simulations and the Dean's correlation based on the bulk Reynolds number (Dean, 1978). The bulk velocity, U_b, is the averaged velocity through the channel cross section, and the bulk Reynolds number, Re_b, is based on the full channel height. The percent error presumes the Dean correlation as the correct value. There is a notable reduction in percent error of tri-split forcing over single forcing with skin friction coefficient calculations in the $\mathrm{Re}_\tau = 2000$ case. The $\mathrm{Re}_\tau = 5200$ also gives a small percent error. These values are below the reported error values

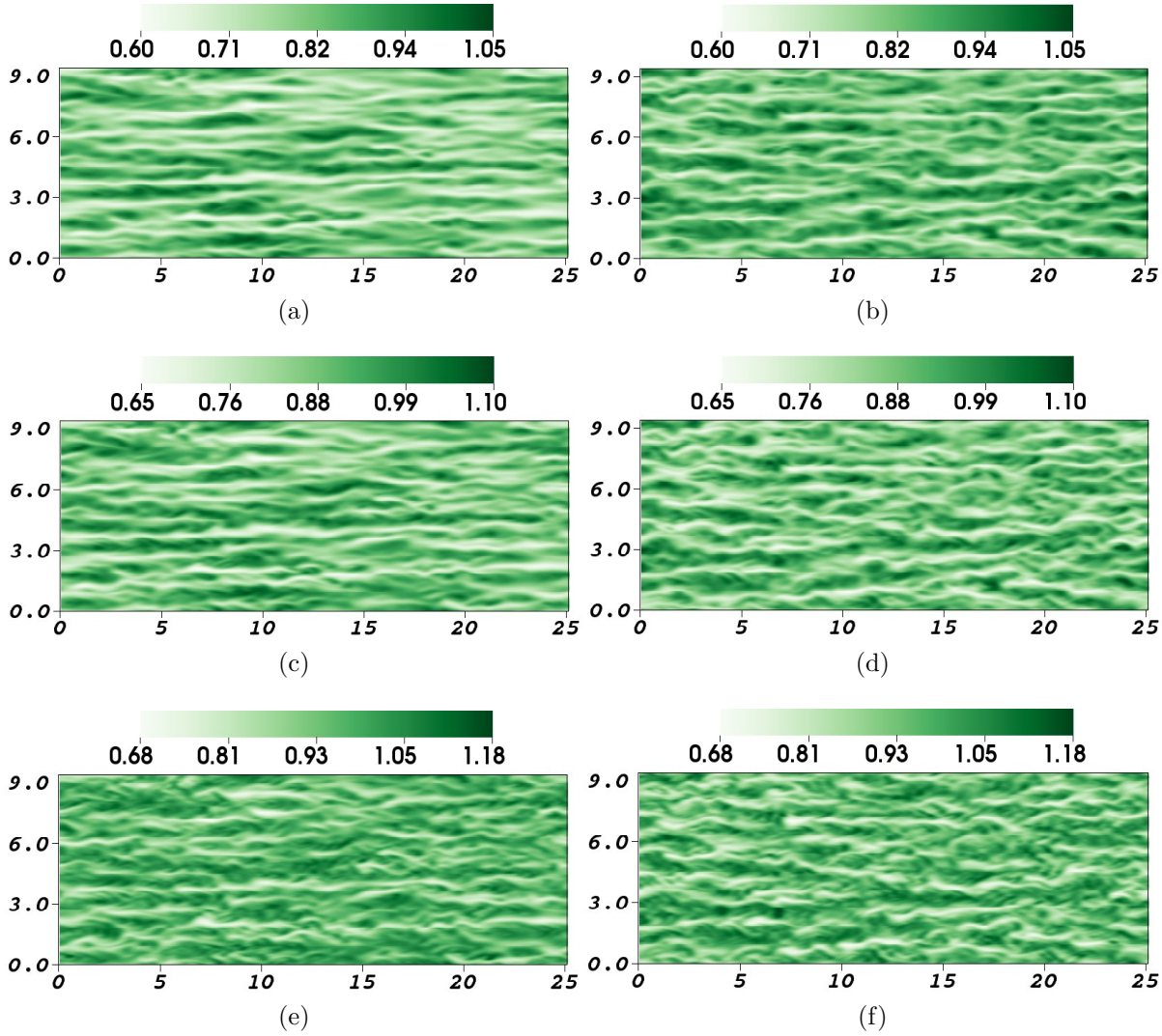


Figure 3.7: Visualization of instantaneous streamwise velocity for $\mathrm{Re}_\tau = 2000$ at wall-normal heights of: (a) and (b) y / $\delta \approx 0.13$; (c) and (d) y / $\delta \approx 0.18$; (e) and (f) y / $\delta \approx 0.27$. $\mathrm{Re}_\tau = 2000$. Single forcing results are shown in the left column and tri-split forcing results are shown in the right column. All velocities and lengths are normalized by the bulk velocity and channel half-height, respectively.

found in early studies of hybrid RANS-LES (Nikitin *et al.*, 2000), which were approximately 10% and above.

3.5 Summary and Conclusions

We investigated the impact of splitting the mean pressure gradient when eddy-viscosity-type hybrid RANS-LES models are used to simulate a turbulent channel flow with periodic boundary conditions. The core idea is to split the driving streamwise mean pressure gradient into three wall-parallel layers such that we can force the RANS, the RANS-LES blended, and the LES core regions separately but under the constraint of a target mass flow rate. Because the effective viscosity in RANS-LES can vary substantially across the channel height, splitting the mean pressure gradient enables each region to develop semi-independently, while keeping the target mass flow rate unchanged.

The split-forcing approach improved upon single-forcing predictions of mean velocity and skin friction coefficients along with reducing the size of unphysical streaks at the RANS-LES interface. We achieved good agreement with normalized DNS benchmarks despite our use of coarse mesh resolutions in all three directions. We uncovered a duality in velocity scale as an artifact of the split forcing approach. We used the concept of local friction velocity (Tuerke and Jiménez, 2013) to find that a constant friction velocity is attained in the LES region, different than the value at the wall. We observed that mean velocity profile scales well with the wall friction velocity, as enforced by a global force balance, whereas the friction velocity value in the LES core is a better scale to normalize the second-order moments. We emphasize that the observed duality is a consequence of the split-forcing approach applied to hybrid RANS-LES and does not apply to the actual physics of turbulent channel flows. We further explained the origins of duality in split-forcing by studying the global force balance in the mean sense.

Split-forcing, as formulated in this work, is currently limited to channel flows. However, it can serve as a useful setup to study turbulent motions in the LES core for much higher

Reynolds number flows without overwhelming computational resources. The current channel flow setup with split forcing could also help formulate a versatile parameterization to energize RANS-LES interface region that can augment existing hybrid RANS-LES models.

CHAPTER 4

An Immersed Boundary Geometric Preprocessor for Arbitrarily Complex Terrain and Geometry

Senocak, I., Sandusky, M., DeLeon, R., Wade, D., Felzien, K., and Budnikova, M. "An immersed boundary geometric preprocessor for arbitrarily complex terrain and geometry. *J. Atmos. Ocean. Tech.*,32 (11): (2015) pp. 2075-2087. (© American Meteorological Society. Used with permission.)

4.1 Abstract

There is a growing interest to apply the immersed boundary method to compute wind fields over arbitrarily complex terrain. The computer implementation of an immersed boundary module into an existing flow solver can be accomplished with minor modifications to the rest of the computer program. However, a versatile preprocessor is needed at the first place to extract the essential geometric information pertinent to the immersion of an arbitrarily complex terrain inside a $3D$ Cartesian mesh. Errors in the geometric information can negatively impact the correct implementation of the immersed boundary method as part of the solution algorithm. Additionally, the distance field from the terrain is needed to implement various subgrid-scale turbulence models, and initialize wind fields over complex terrain. Despite the popularity of the immersed boundary method, procedures used in the geometric preprocessing stage have received less attention. We find that concave and convex regions of complex terrain are particularly challenging to process with existing procedures discussed in the literature. To address this issue, we present a geometric preprocessor with a distance field solver, and demonstrate its versatility for arbitrarily complex geometry, terrain and urban environments. The distance field solver uses the initial distance field at the immersed boundaries and propagates it to the rest of the domain by solving the Eikonal equation with the fast sweeping method.

4.2 Introduction

The immersed boundary (IB) method was first proposed by Peskin (1972) for use in biomedical engineering. The intended purpose was to simulate flow through a human heart valve as a fluid-structure interaction problem on a fixed structured mesh. In the original IB method, the effect of the moving or stationary boundaries were explicitly represented as a body-force term in the momentum equations. However, researchers have reported numerical stability issues associated with having an explicit body-force in the equations. The direct forcing approach introduced by Mohd-Yusof (1997) and Fadlun *et al.* (2000) provided an alternative formulation for the body-force, and eliminated the stability issues arising from the explicit body-force formulation. The popularity of the immersed boundary approach has increased markedly after the introduction of the direct-forcing approach (Mittal and Iaccarino, 2005). In the direct forcing approach, the body force to represent an immersed boundary is implicitly taken into account by a reconstruction step on the primitive variables at the cells cut by the solid boundaries.

There is a growing interest to adopt Cartesian immersed boundary methods to simulate environmental flows under realistic conditions (Senocak *et al.*, 2004; Lundquist *et al.*, 2010, 2012; Kang *et al.*, 2011; DeLeon *et al.*, 2012). Simulations of wind over arbitrarily complex terrain is an active area of research because of the rapidly expanding wind energy field. However IB methods have been primarily proposed for engineering fluid flow applications at modest Reynolds numbers, and therefore do not readily extend to atmospheric flows over arbitrarily complex terrain. Successful simulations of atmospheric flows over arbitrarily complex terrain necessitate reconstruction methods and subgrid-scale models that are suitable for the underlying rough terrain resolved by a relatively coarse spatial resolution used in the computations. In order to make progress on this front for arbitrarily complex terrain, we need a versatile IB preprocessor that can calculate the geometric information needed for a variety of reconstruction methods. In addition, the preprocessor needs to calculate the

distance field to an arbitrarily complex terrain, because distance information is needed in several turbulence models, in initializing three dimensional wind fields over complex terrain, and extracting flow variables over a surface contour at a specified height above ground level for data analysis.

Computer implementation of the IB method inside an existing flow solver can be done with minor modifications to the rest of the solver. The non-intrusive nature of the implementation is one of the reasons why IB methods have become so popular for various applications. Researchers have adopted different interpolation methods (e.g., linear, quadratic, logarithmic) as part of the reconstruction of the primitive variables at the cells cut by an immersed boundary (Fadlun *et al.*, 2000; Gilmanov and Sotiropoulos, 2005; Choi *et al.*, 2007; Mittal *et al.*, 2008). In most validation studies, researchers have relied on analytical shapes (e.g. sphere, Witch of Agnesi Curve, sinusoidal ridges etc.) to validate the underlying numerical method for fluid flow simulation. However, for arbitrarily complex geometries, such as a mountainous terrain or a three dimensional complex geometry, a robust preprocessing program is needed to correctly tag the cells as fluid, solid and immersed boundary nodes, identify neighboring cells for interpolation, bind an IB node to a surface element, and calculate the distance to the surface for use in the interpolation part of the reconstruction step. The preprocessing stage for arbitrarily complex geometry is composed of algorithms from the computational geometry field. Majority of published work on IB method has focused on the numerical accuracy of the flow solution over analytical geometries. Description and computer implementation of the preprocessor for arbitrarily complex geometry has received scant coverage. An exception is the work of Gilmanov and Sotiropoulos (2005), who applied the immersed boundary method to complex geometries such as undulating fish-like body and planktonic copepod, and described the algorithms used in treating the complex flexible immersed boundaries. Although, Gilmanov and Sotiropoulos claimed their preprocesing procedure to be general, we find that their procedure does not readily apply to arbitrarily complex mountainous terrain. Choi *et al.* (2007) and Edwards *et al.* (2010) also point to

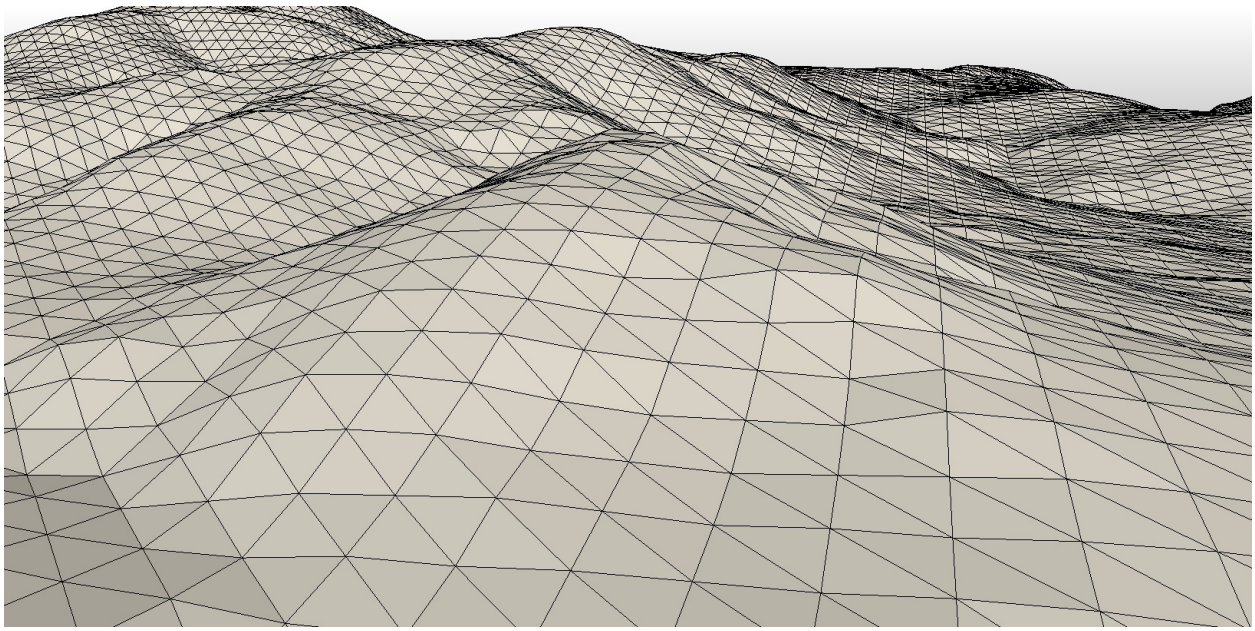


Figure 4.1: A triangulated surface mesh fitted to the terrain of Buckman Springs, CA.

problems with existing procedures and describes an alternative approach.

In what follows, we present a new procedure to handle arbitrarily complex geometry, terrain and urban environments. Additionally, we apply the fast sweeping method (Zhao, 2004) to solve the Eikonal equation to efficiently calculate the distance field from any complex geometry for the full flow domain. We test these two components of the preprocessor for challenging test cases to demonstrate the versatility of the geometric preprocessor. We do not perform any flow simulations in this study, because the preprocessing stage for arbitrarily complex terrain is imperative for correct implementation of any IB method and subgrid-scale models. Therefore we focus entirely on detailing the steps in the geometric preprocessor, and conduct a thorough investigation over different kinds of geometry.

4.3 Review of Existing Procedures

Consider a small portion of the terrain geometry intersecting the background Cartesian mesh as illustrated in Figure 4.1. Note that the Cartesian mesh spacing does not have to be uniform (i.e. $\Delta x \neq \Delta y \neq \Delta z$) First, we must identify all the Cartesian points that are in close

proximity of the terrain. These are called *near boundary nodes*. This can be accomplished by sweeping through all triangular elements of the surface mesh, and identifying all the Cartesian points that are within a search radius, r_s, from the centroid of the triangular element of interest. An optional bounding box can be created to eliminate Cartesian points irrelevant to the search problem and the search can be distributed to multiple parallel processes. Gilmanov and Sotiropoulos (2005) suggest r_s to be approximately equal to the smallest Cartesian mesh dimension, which we find to be insufficient in practice.

Next, we sweep through all the near boundary nodes to distinguish them as being *interior* or *exterior* relative to the terrain. We find the centroid of all triangular elements within a search radius of r_s from each node. We create a position vector $\mathbf{d}$ from the centroid of the triangular element to the near boundary node at hand. We then perform a dot product between the position vector and the surface normal vector of the triangular element. The dot product is evaluated for every triangular element that is found to be within the search radius of a near boundary node. Gilmanov and Sotiropoulos (2005) suggest that if the dot product is greater than zero for *at least one* boundary node within the search radius, then the node is exterior to the geometry. If the dot product is less than zero for *all* triangular elements, then the node is interior to the geometry.

Gilmanov and Sotiropoulos (2005) state that the above procedure is general and applicable to arbitrarily complex geometry. However, we find that the above procedure is not entirely general for all concave configurations of a terrain. Depending on the resolutions of the surface terrain, and the Cartesian mesh at hand, we are not able to separate a node as exterior or interior to a terrain for some challenging concave configurations following their procedure. Choi *et al.* (2007) also report that the above procedure does not work for some complex geometries.

A problematic configuration, which can be common on a complex terrain surface mesh, is illustrated in 2D in Figure 4.2a. In this schematic we tag the nodes manually to illustrate how some of the existing procedures can fail. The filled circular markers show the near

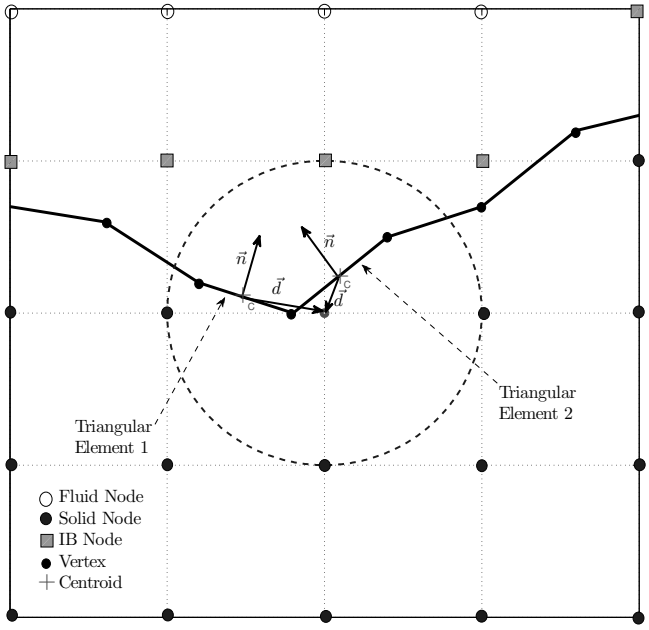
boundary solid points, and the filled square markers show the immersed boundary nodes upon which a reconstruction scheme is to be imposed.

Application of the interior/exterior identification stage of the above procedure to the grid point of interest identified in Figure 4.2a will result in two iterations of the dot product check as two triangular elements fall within the search radius. Triangular element 1 gives $\mathbf{d} \cdot \mathbf{n} > 0$, terminating the identification state, incorrectly labeling the grid point as exterior.

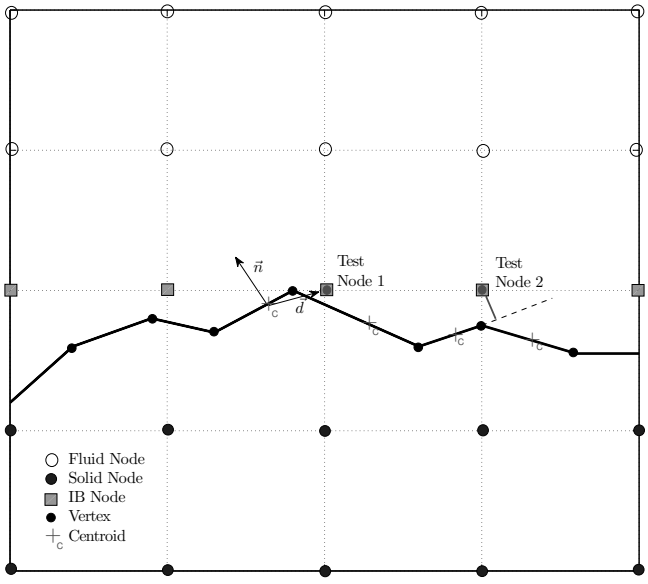
To remedy the above logic failure for some complex geometries, Choi *et al.* (2007) and Edwards *et al.* (2010) proposed an angle-weighted pseudo-normal vector, which essentially averages the variations in the local surface normal vector to ensure smoothness of the underlying geometry. However, as also mentioned by Choi et al., there is no guarantee that the angle-weighted normal vector fixes the issue for all cases. Mittal *et al.* (2008) use a slightly different method for distinguishing between solid, fluid, and IB nodes, which involves finding the closest triangular element to a node, and then taking the dot product of the position vector from centroid of the triangular element to the node with the surface normal vector of the same triangular element. We find that this procedure works as long as the surface mesh has nearly uniform triangular elements. However, when the surface mesh is composed of large and small triangular elements, the procedure also fails to correctly tag all the points.

Figure 4.2b displays an area in which a procedure described in Mittal *et al.* (2008) could fail. In this convex scenario, *test node #1* would wrongly pick the triangular element to its left because its centroid is the closest, despite being positioned above a different triangular element. Consequently, the dot-product with the surface normal of the triangular element would cause test node #1 to be wrongly tagged as solid. Therefore identifying triangular elements based on the closest centroid is not a fail proof logic.

Figure 4.2b illustrates another potential problem that arises when binding a node to a surface element. In the reconstruction stage of the IB method, boundary conditions need to be assumed on a surface element. In analyzing *test node #2* we see that distance to the surface should be the Euclidean distance calculated between the test node and the vertex.



(a)



(b)

Figure 4.2: 2D schematics illustrating inherent discrepancies in published work. $\vec{d}$ is the distance vector from the centroid of the triangular element to the near boundary node, and $\vec{n}$ is the surface normal of the triangular element. (a) Concave region, (b) Convex region.

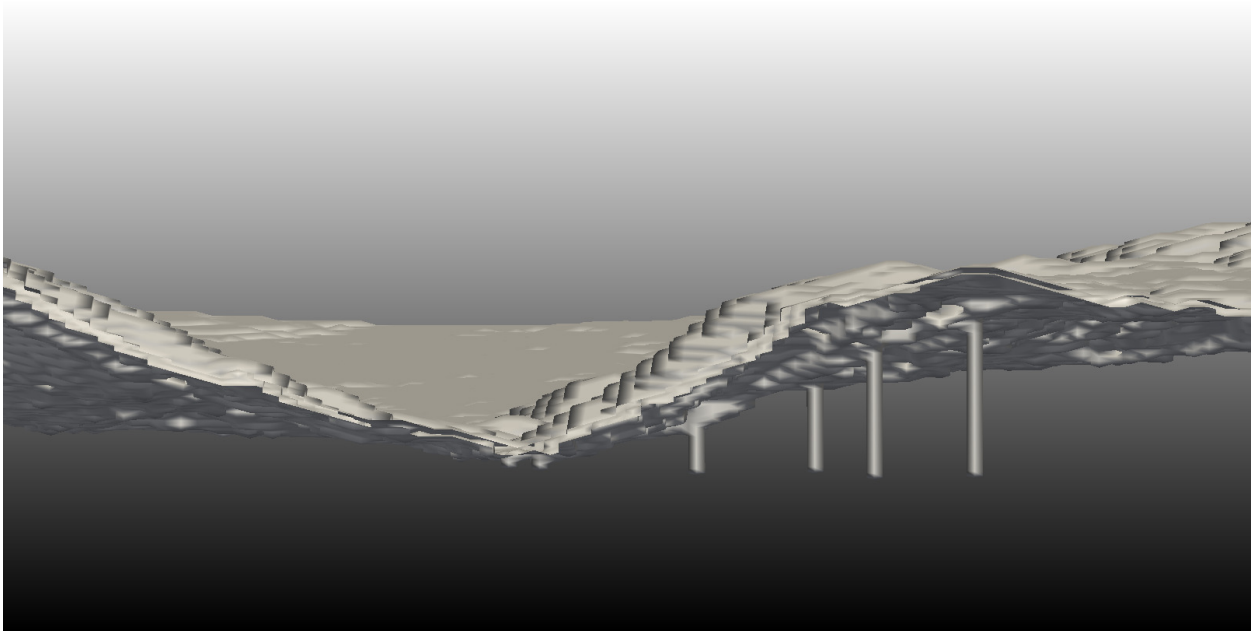


Figure 4.3: Isosurface visualization of the solid node flags for a section of complex terrain. The cylindrical dips from the surface indicate unflagged nodes where the interior/exterior logic fails.

However, because of binding the test node with a surface triangular element, the distance can be wrongly calculated as the shortest distance between the test node and the plane that includes the triangular element as illustrated in Figure 4.2b. This issue does not cause the IB method to fail, but it introduces geometry errors by altering the underlying geometry, which would affect both the reconstruction step and calculation of the distance field using the Eikonal equation. In situations like this, an additional logic is needed to decide whether the appropriate distance is to the triangular element, the edge or the vertex.

We emphasize that these logic failures only occurs for some portion of the complex terrain. Figure 4.3 demonstrates the result of these logic failures as four dips in the isosurface visualization of solid/interior node flags for only a portion of Buckman Springs terrain. For a reasonably large complex terrain area (e.g., 10km $\times$ 10km) these dips result from errors in the solid/fluid node tagging process and can be numerous. These errors may go unnoticed, which will result in missing or incorrect geometric information for the IB method, and will also pollute the calculation of the distance field through the solution of the Eikonal equation.

We should emphasize that the deficiencies we have identified in the above procedures may not manifest in all geometries. The resolutions of the Cartesian and the triangular surface meshes relative to each other can cause these inherent discrepancies to go undetected. We have experienced this issue in our own practice. In the following section, we introduce a new procedure that addresses the shortcomings described above.

4.4 Description of the Geometric Preprocessor

Our new procedure to obtain the essential geometric information (i.e., node tagging, distance to surface from nodes used in the reconstruction, binding an IB node with a surface element) to implement the immersed boundary method benefits from various components discussed in different studies (Gilmanov and Sotiropoulos, 2005; Mittal *et al.*, 2008; Choi *et al.*, 2007; Sunday, 2001a; D. Eberly, 2008; Holcombe, 2012).

In our practice, we obtain terrain maps from the U.S. Geological Survey website and convert it to a Digital Elevation Model (.dem) file. The publicly available MicroDEM software (Guth, 2013) can be used to crop a smaller area of interest from a DEM map. We use DEM maps with 10m resolution in the horizontal. We then fit a triangulated surface mesh to the terrain using a function from MATLAB Central File Exchange (McDonald, 2004), and save the surface mesh in stereolithography (STL) file format. An STL file contains the vertices, centroid, and surface normal of every triangular element in the mesh. Figure 4.1 shows the triangulated surface mesh for an area in Buckman Springs, CA.

The first component of our procedure is to classify Cartesian mesh points as *interior* or *exterior* relative to the immersed geometry. Concave and convex zones resulting from the surface mesh of the complex geometry require a robust method. We developed our own procedure by using some of methods from Holcombe (2012). We follow a 2D approach for the interior/exterior classification step. We first define a 2D search radius that is a multiple of the norm of the maximum x and y span of an STL triangle. We use a conservative search radius which is four times the norm. Note that the size of the search radius only affects the

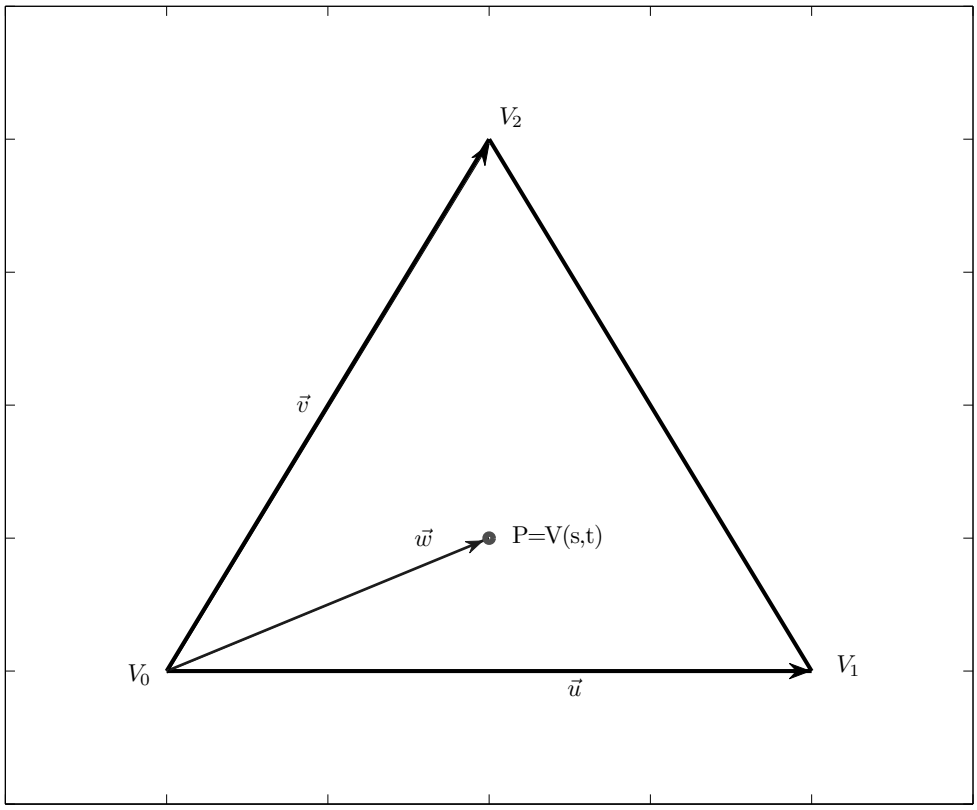


Figure 4.4: Illustration of the methods used to determine if a point is within a triangle.

execution time. Next, a cylindrical slab of this radius is projected down in the z direction from each Cartesian point. Triangular elements of the surface mesh that fall within this slab are tagged, and then those elements are examined to identify the triangular elements that intersect a ray projection from the Cartesian point in the z direction.

There are a few methods for determining whether or not a ray projection intersects a triangle. Figure 4.4 helps illustrate the method presented by (Sunday, 2001a). This method involves, first, finding the intersection of the ray, projected in the z direction with the plane of the triangular element. After the intersection point is identified, an inclusion check is applied to find out if this point is within the triangular element or not. Following Sunday (2001a), the intersection point can be parameterized in terms of s, and t coordinates:

$$s = \frac{(\vec{u}\cdot\vec{v})(\vec{w}\cdot\vec{v}) - (\vec{v}\cdot\vec{v})(\vec{w}\cdot\vec{u})}{(\vec{u}\cdot\vec{v})^2 - (\vec{v}\cdot\vec{v})(\vec{u}\cdot\vec{u})} \tag{4.1}$$

$$t = \frac{(\vec{u}\cdot\vec{v})(\vec{w}\cdot\vec{u}) - (\vec{u}\cdot\vec{u})(\vec{w}\cdot\vec{v})}{(\vec{u}\cdot\vec{v})^2 - (\vec{v}\cdot\vec{v})(\vec{u}\cdot\vec{u})} \tag{4.2}$$

As long as these parameters follow the inequalities $s \geq 0$, $t \geq 0$ and $(s + t) \leq 1$, the point is within the bounds of the triangle. For the interior/exterior classification step of our procedure, we rely on the winding algorithm presented in O'Rourke (1994) with an implementation similar to the one presented by Holcombe (2012). In this 2D approach, the triangle in question and the Cartesian point are projected onto the xy plane by simply neglecting their Z coordinates. The same $\vec{u}$, $\vec{v}$, and $\vec{w}$ are constructed, as in Figure 4.4, for each of three vertices. The angles between $\vec{w}$ and $\vec{u}$, $\vec{w}$ and $\vec{v}$, and $\vec{u}$ and $\vec{v}$ are found from the dot product of the unit vectors as follows:

$$\begin{aligned} \theta_1 &= \arccos(\hat{w}\cdot\hat{u}), \\ \theta_2 &= \arccos(\hat{w}\cdot\hat{v}), \\ \theta_3 &= \arccos(\hat{v}\cdot\hat{u}). \end{aligned} \tag{4.3}$$

Note that quantities with a hat symbol$\hat{}$ are unit vectors. If the condition $\theta_1 \leq \theta_3$ and $\theta_2 \leq \theta_3$ holds true when applied to each of the vertices (i.e., V_0, V_1, V_2), then the point lies within the triangle in the xy plane. If no triangle intersects the 2D projection of the point then the point is exterior, and marked as fluid.

Note that a ray projected in the z direction may intersect multiple triangular elements for a watertight geometry. Therefore, we identify the triangular element closest to the Cartesian point in the z direction by comparing the z coordinate of the Cartesian node against the z coordinate of the centroid of the triangular elements that have been intersected. Once the closest triangle is identified, we perform a three dimensional dot product between the surface normal $\vec{n}$ of the closest triangular element, and the position vector $\vec{r}$ drawn from the centroid to the Cartesian node. If the dot product is greater than zero, then the Cartesian

node is exterior to the geometry and tagged as a fluid node. Else, the node is interior to the geometry, and it is tagged as a solid node. Once all the Cartesian nodes are tagged as exterior (fluid) or interior (solid), we then sweep through the fluid nodes and tag any fluid node that has, at least in one direction, a neighboring solid node as an *immersed boundary* (IB) node.

This procedure does not produce any false outputs from the dot product as the Cartesian point is always positioned directly above or below the triangular element, avoiding the problematic situations described in Figure 4.2.

4.5 Preprocessing for the IB Reconstruction Step

In our approach the immersed boundary reconstruction method relies on interpolation along the vector pointing from the closest point on the STL surface to the IB node. In this step, a separate logic is needed to bind the IB node to the closest surface element, which can be a triangle, an edge or a vertex. Flow reconstruction can then be done on a line along this unit vector $\hat{n}_{BV}$ that binds the IB node to the closest surface element. Depending on the flow regime the reconstruction could assume a linear or a logarithmic profile.

After all of the IB nodes are determined, the shortest distance to the solid must be found for each IB node. This initial distance is then propagated to the rest of the fluid domain through the solution of the Eikonal equation with the fast sweeping algorithm of Zhao (2004). First, the closest triangle vertex is established for each IB Node. Next, the triangles that share this vertex are identified and their indices are stored. A line that passes through the IB node and is parallel to the surface normal of the triangular element at hand is intersected with the plane that includes the triangular element. Equations 4.1 and 4.2 are used to determine if the intersection point falls inside the triangular element or not. If the intersection point is within the triangular element, the IB node is bound to that triangular element. The distance from the intersection point to the IB Node is stored as the closest distance.

If none of the triangles are intersected by the line that passes through the IB node, we move on to check the edges sharing the common vertex using an algorithm described in D. Eberly (2008) and Sunday (2001b). We calculate a vector $\vec{w}$ pointing from the vertex to the Cartesian IB node, and separate unit vectors $\hat{u}_i$ along each triangle edge with a common origin at the vertex. The shortest distance d to each edge is calculated as follows:

$$d = \min(\|\hat{u}_i \times \vec{w}\|), \tag{4.4}$$

where the subscript i is an identifier index that refers to the set of edges that share the closest the vertex to the IB node. Note that if the angle between $\hat{u}$ and $\hat{w}$ is greater than 90° for an edge, then the intersection point is not contained within the edge element, and that edge is ignored.

If we do not find an intersection point lying within any of the edges that share the closest vertex to the IB node, we then simply calculate the Euclidean distance from the IB Node to the closest vertex. Whether it is a triangle, edge or a vertex, we refer to the unit vector along the line that connects the intersection point on the closest element to the IB node as the *binding vector* $\hat{n}_{BV}$.

When we applied these binding methods to the Stanford Bunny geometry, 68.4% of the IB nodes were bound to triangular elements, 31.5% were bound to edges, and 0.1% were bound to vertices. The respective proportions for the Buckman Springs terrain were 74.4 %, 25%, and 0.6%. These percentages suggest that the IB node binding step is an important part of the preprocessor.

Once the IB node is bound to its closest element on the surface mesh, a line that passes through the IB node in the direction of binding unit vector $n\hat{_{BV}}$ is extended toward the fluid domain. Figure 4.5 illustrates the situation for an IB node bound to a triangular element. The geometric information needed to interpolate values onto the projected line is extracted as follows.

1. Identify the closest Cartesian cell face intersected by the projected line. To accomplish this task, we solve the plane-line intersection problem for each face of the Cartesian cell and compare the Euclidean distances. The parametric equation of a 3D line is substituted into the standard vector equation of a plane. Cartesian cell faces in x, y, z directions will have the following unit normal vectors: [1,0,0], [0,1,0], [0,0,1]. Therefore, the parameter, t, of the parametric equation of the line can be found by

$$t = \frac{r_{P,i} - r_{BV,i}}{\hat{n}_{BV,i}} = \frac{\delta_i}{\hat{n}_{BV,i}} \tag{4.5}$$

   where the subscript i indicates the direction, and no summation over repeated indices is implied. $\hat{n}_{BV,i}$ is the i component of the unit vector in the direction of the line. $r_{P,i}$ is a coordinate of a point on the plane, and $r_{BV,i}$ is the coordinate of reference point for the 3D line. Note that we calculate the parameter, t, for each face of the Cartesian cell. Regardless of their values, the quantity $r_{P,i} - r_{BV,i}$ is equal to a Cartesian mesh spacing δ_i in the direction of a unit normal vector belonging to a cell face. We then make use of the parameter t of the parametric equation of the line. At this stage, we do not know which of three faces have been intersected first by the line. The closest Cartesian cell face is determined by the smallest Euclidean distance.

2. Find the coordinates of the intersection point on the cell face using the parametric equation of the line that passes through the IB node as follows:

$$\begin{aligned} x_{int} &= x_{IB} + \hat{n}_{BV,x} t, \\ y_{int} &= y_{IB} + \hat{n}_{BV,y} t, \\ z_{int} &= z_{IB} + \hat{n}_{BV,z} t, \end{aligned} \tag{4.6}$$

   where subscripts int and IB represent the intersection point and IB node, respectively, for each of the coordinate directions. The subscript BV indicates the unit binding vector as discussed earlier.

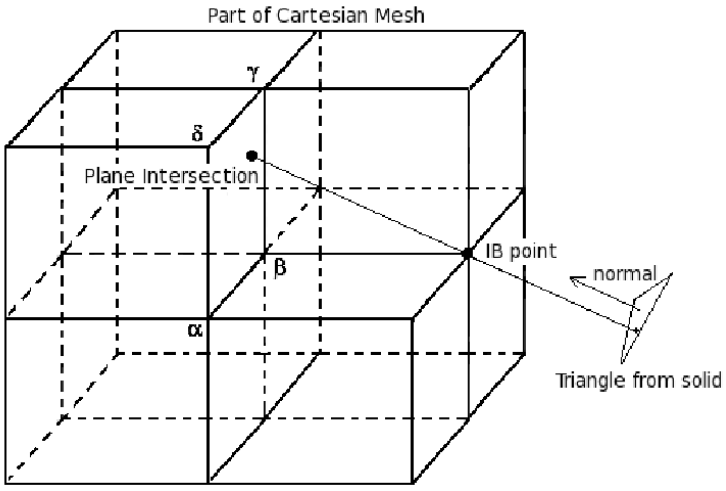


Figure 4.5: 3D example of the surface normal of a triangle intersecting the nearest Cartesian cell face after passing through the immersed boundary node.

3. Calculate the weights to apply a bilinear interpolation on the flow variables on the cell face.

Knowing the flow variables at the intersection point and the immersed surface enables the reconstruction of flow variables (e.g. velocity components and scalar quantities) at the IB node using linear, logarithmic, quadratic, or any other interpolation methods (Fadlun *et al.*, 2000; Gilmanov and Sotiropoulos, 2005; Choi *et al.*, 2007; Mittal *et al.*, 2008).

4.6 Parallel Implementation

The above procedure is implemented in parallel to accelerate the preprocessing stage. In our computer implementation, we partition the surface mesh in STL format in the same manner as the computational domain (i.e., in the x direction), but with a buffer zone on both sides to make sure all triangular element information pertinent to the process is available to each Message Passing Interface (MPI) process. We then execute each section independently because communication between processors is not necessary. We successfully tested our parallel implementation up to 32 processors. We use bounding boxes to find where the terrain sits in the Cartesian domain, meaning the processors do not have to waste computer

memory on empty space.

4.7 Distance Field Calculation

Signed distance fields and level set methods have been used in a number of applications such as computer generated water surfaces for visualization (Osher and Fedkiw, 2003) and improved imaging for medical applications (Pu *et al.*, 2008). The distance field is especially important for our application, because we wish to implement a variety of turbulence models to simulate winds over complex terrain. The distance from a surface is an essential piece of information to apply near-surface parameterizations in subgrid-scale (SGS) turbulence models. Additionally, the distance field information helps to impose specific wind profiles (e.g. power law) at lateral boundary conditions and also for solution initialization.

The level-set method is a fast algorithm to calculate the signed distance from a surface. Two popular approaches to implement the level-set method are the fast-marching method (FMM) (Sethian, 1999), and the fast sweeping method (FSM) (Zhao, 2004). FMM loops over the grid points closest to the surface and moves outward. FMM has the advantage of allowing for calculations of narrow bands of data near the surface when only that data is desired. This approach can be advantageous for dynamic surfaces. The runtime complexity of FMM is $O(N \log(N))$, where N is the number of mesh points. On the other hand, computer implementation of FSM is relatively easier as compared to FMM, and has runtime complexity of $O(N)$. Therefore we use FSM as part of our geometric preprocessor.

4.8 Signed Distance Calculation and Discretization

After the application of the immersed boundary preprocessor, we are left with a computational domain that is split into three parts: the solid, the fluid, and a layer separating the solid from the fluid. This separation layer contains the immersed boundary points that holds the shortest distance to the underlying geometry. However, we also need the distance field

in the fluid points away from the surface for use in certain turbulence models and wind field initialization. Therefore, we propagate the distance field at the immersed boundary points to other mesh points using the FSM.

The signed distance function implicitly defines the terrain on which $\phi(\vec{x}) = 0$. The distance field is positive in the fluid domain, and negative in the solid domain.

Following the derivation in Osher and Fedkiw (2003), the Eikonal equation is written as follows.

$$\|\nabla\phi\| = 1. \tag{4.7}$$

We follow the work of Zhao (2004) to numerically solve the Eikonal equation, but allow a directionally non-uniform mesh (i.e. $dx \neq dy \neq dz$) in our formulation. The discretized form of the Eikonal equation in three dimensions can be written as follows.

$$\begin{gathered}\left(\frac{\phi_{i,j,k} - \phi_{xmin}}{dx}\right)^2 + \left(\frac{\phi_{i,j,k} - \phi_{ymin}}{dy}\right)^2 + \left(\frac{\phi_{i,j,k} - \phi_{zmin}}{dz}\right)^2 = 1 \\ i = 2, ..., NX-1, j = 2, ..., NY-1, k = 2, ..., NZ-1,\end{gathered} \tag{4.8}$$

where $\phi_{xmin} = \min(\phi_{i-1,j,k}, \phi_{i+1,j,k})$,$\phi_{ymin} = \min(\phi_{i,j-1,k}, \phi_{i,j+1,k})$, and $\phi_{zmin} = \min(\phi_{i,j,k-1}, \phi_{i,j,k+1})$. After solving the above equation for $\phi_{i,j,k}$, we are able to calculate the signed distance at any mesh point. The distance field propagation requires initialization which is provided by the immersed boundary geometric processor, as stated earlier.

4.9 Fast Sweeping Algorithm

Now that we have a method to solve Equation 4.7 at any point on our computational domain, we must select a method of iteration that will ensure convergence. Since the calculation at any given point depends on the distance values surrounding that point, it is not sufficient to arbitrarily cycle through the grid points in the domain. Initially, there are only distance values on one layer near the implicit surface which were calculated during the immersed boundary process. As in Zhao (2004), all other points outside the surface (not including our

initialization layer) are set to a large positive number which will be updated later during the iteration process. We then sweep the domain and update the distance values, $\phi_{i,j,k}$, in the following way: $\phi^{new}_{i,j,k} = \min(\phi^{old}_{i,j,k}, \phi^{calc}_{i,j,k})$, where $\phi^{calc}_{i,j,k}$ is the value calculated in Equation 4.8 for the current sweep iteration. This step ensures that the distance value at each grid point remains non-increasing and the value will only update when a smaller distance value has been calculated. The order for a single complete sweep of the domain is critical, and it is given as follows:

$$(1)i = 1 : NX, j = 1 : NY, k = 1 : NZ; \quad (2)i = NX : 1, j = 1 : NY, k = NZ : 1;$$
$$(3)i = NX : 1, j = 1 : NY, k = 1 : NZ; \quad (4)i = 1 : NX, j = 1 : NY, k = NZ : 1;$$
$$(5)i = NX : 1, j = NY : 1, k = NZ : 1; \quad (6)i = 1 : NX, j = NY : 1, k = 1 : NZ;$$
$$(7)i = 1 : NX, j = NY : 1, k = NZ : 1; \quad (8)i = NX : 1, j = NY : 1, k = 1 : NZ$$

The above loop order covers all possible sweep directions for a three-dimensional domain. As suggested in Bridson (2008), the sweep process may be repeated several times for increased accuracy.

4.10 Results

Our geometric preprocessor can handle mountainous terrain, urban environments and complex geometry. We use 3D test models that are popular within the computer graphics community to demonstrate versatility of our preprocessor. To illustrate the success of the interior/exterior node tagging algorithm, a contour plot of the interior (solid) nodes for the so-called Stanford Dragon geometry can be viewed in Figure 4.6. This image shows that the highly complex geometric detail of the dragon can be handled with our procedure.

Figure 4.7a and b presents CAD geometries of the Stanford Dragon and Bunny stored as an STL file. These geometries are immersed inside a 3D Cartesian domain to calculate the geometric information in the vicinity of the surface. The distance field at the immersed

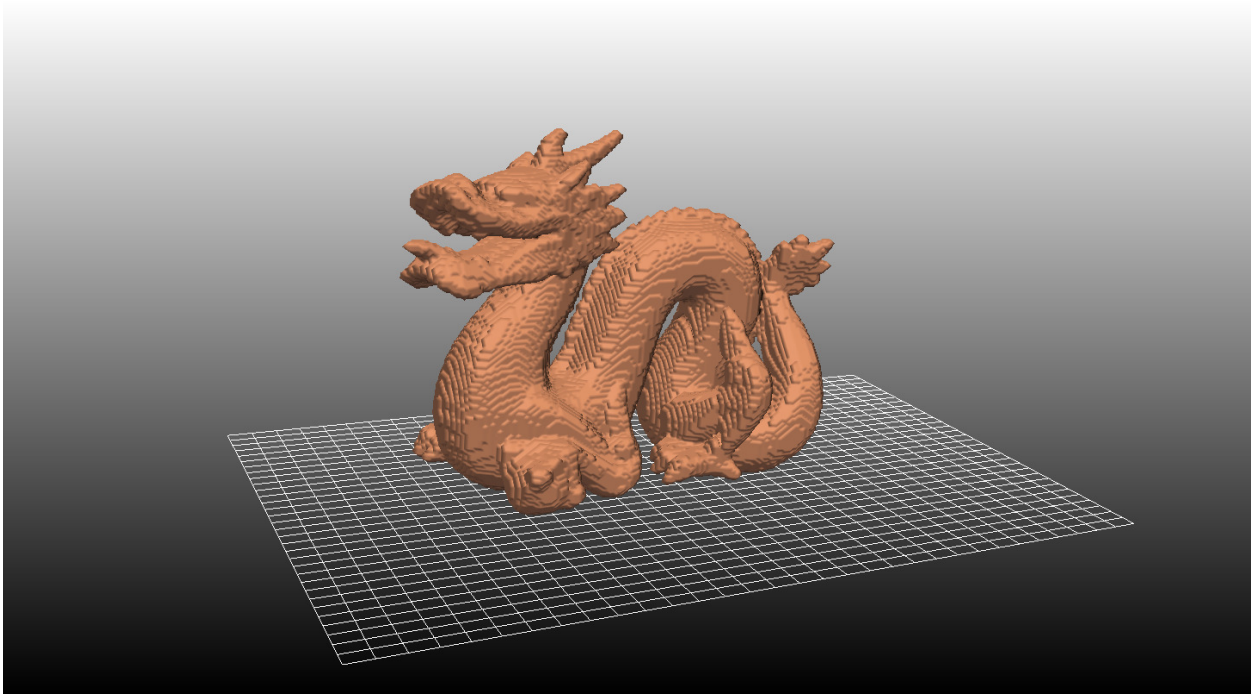
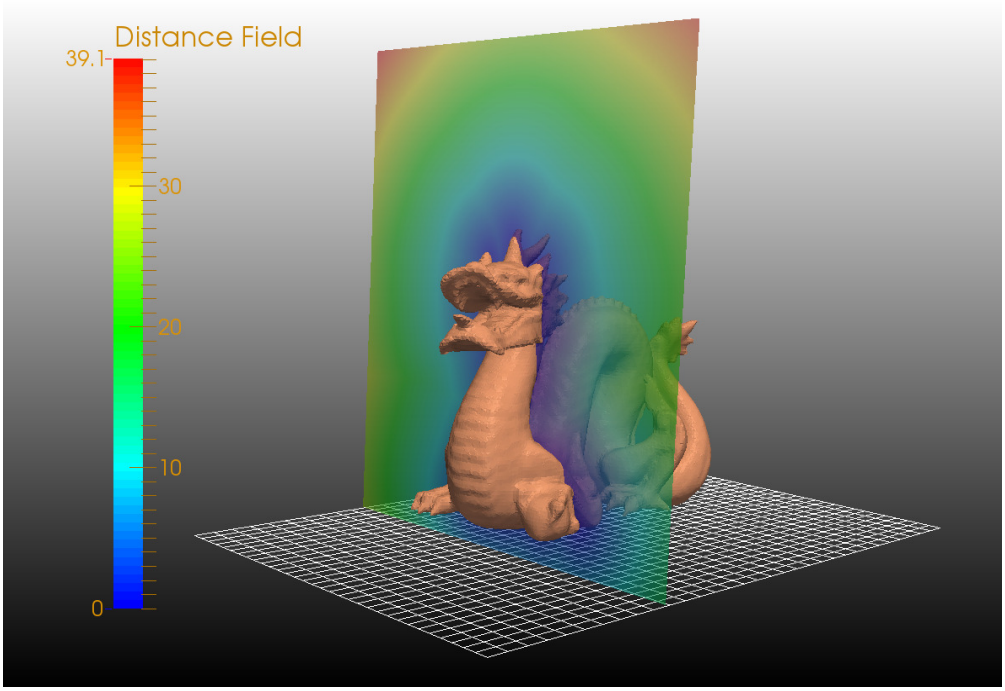
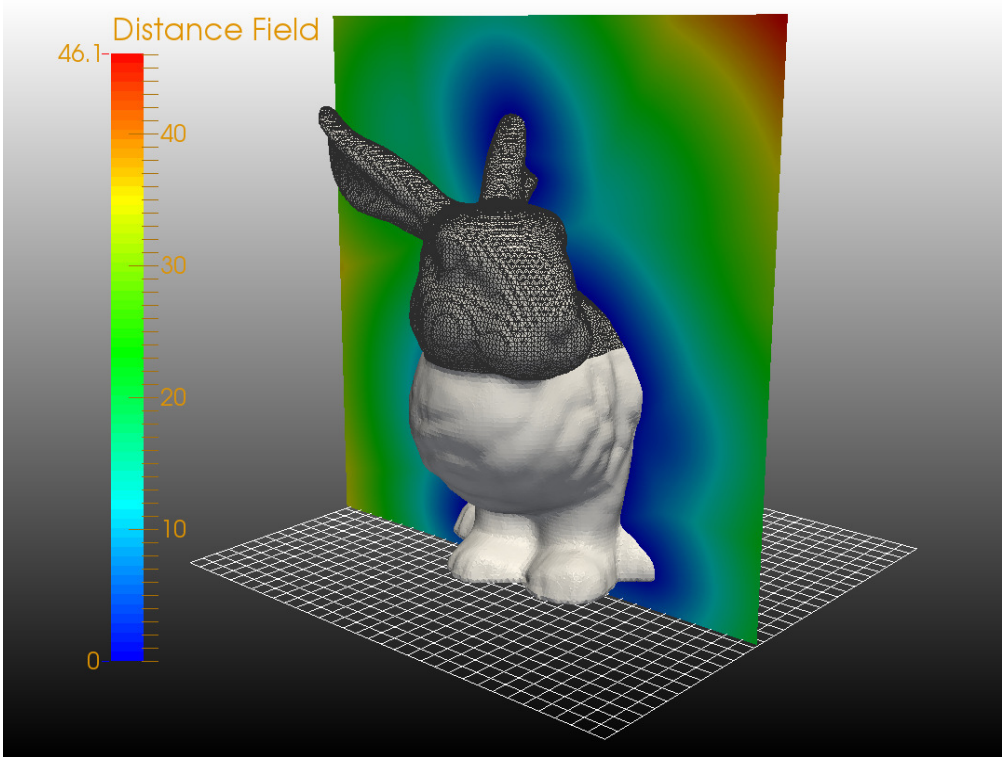


Figure 4.6: Contour plot of the solid nodes for the Stanford Dragon showing good detail. A 2D plane of the Cartesian mesh with 1 of every 8 lines is also displayed.

boundary cells are then propagated using the fast sweeping method. From these images one can see that the distance field emanates smoothly away from the surface without any kinks or disturbances. Kinks or disturbances in the distance field typically hint problems in the initial distance field that the geometric preprocessor calculates. The underlying Cartesian meshes used in the calculations are also displayed in these images. Once the interior/exterior classification is done, we place an IB node between a solid and fluid node. A check that we perform on the preprocessor is to make sure there is always an immersed boundary node between a fluid and a solid node. To perform this check, we color code each of the Cartesian mesh points in the vicinity of the geometry. Figure 4.8 shows that this condition is successfully satisfied for the complex terrain of the Buckman Springs, CA. Figure 4.9a shows the terrain for a portion of the Buckman Spring area in California. The terrain was immersed into a Cartesian domain of approximately 8.8 km $\times$ 6.8 km with horizontal and vertical mesh spacings of 35m and 10m, respectively. The initial distance field is then propagated to the rest of the domain using the FSM approach. We can observe from contour plots that the



(a)



(b)

Figure 4.7: Composite view of the distance field around the Stanford Dragon and Bunny geometries. (a) A 2D plane of the Cartesian mesh with 1 of every 8 lines shown can be seen below the dragon (b) A 2D plane of the Cartesian mesh with 1 of every 4 lines shown is visible below the bunny. A portion of the surface mesh is also displayed over the bunny head.

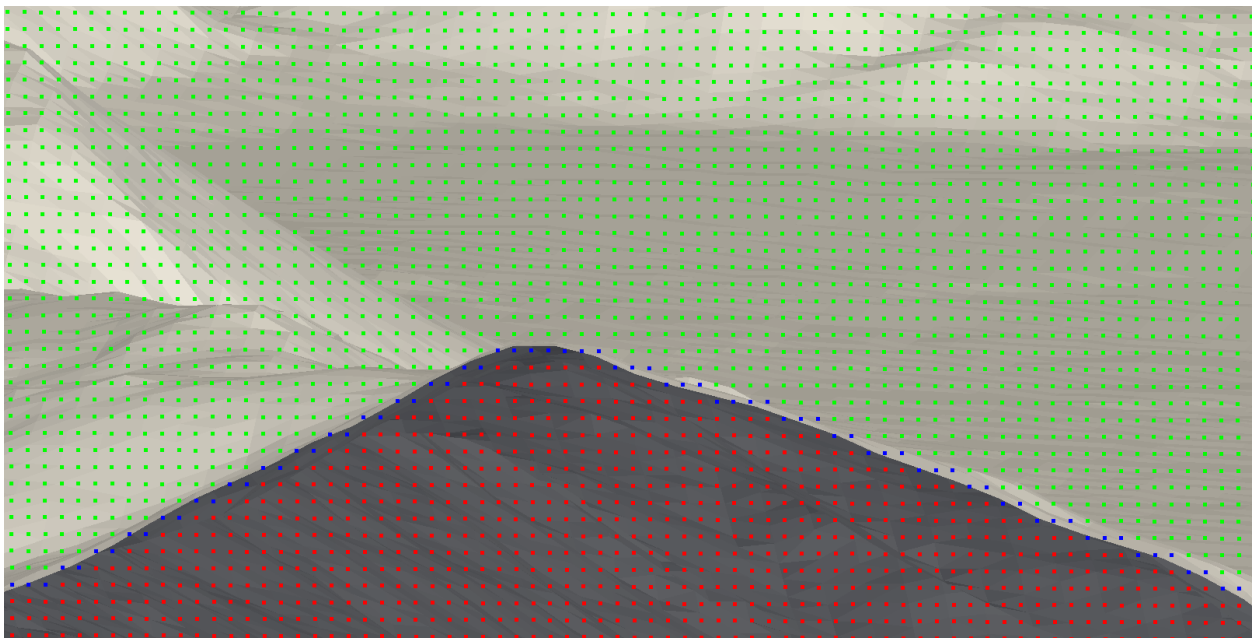
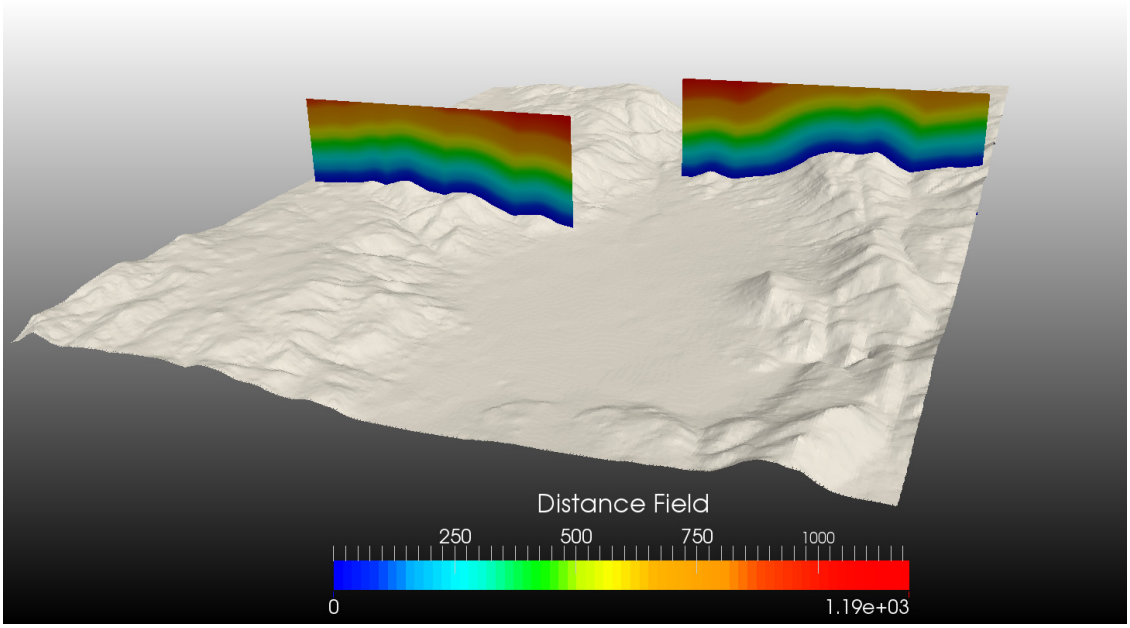


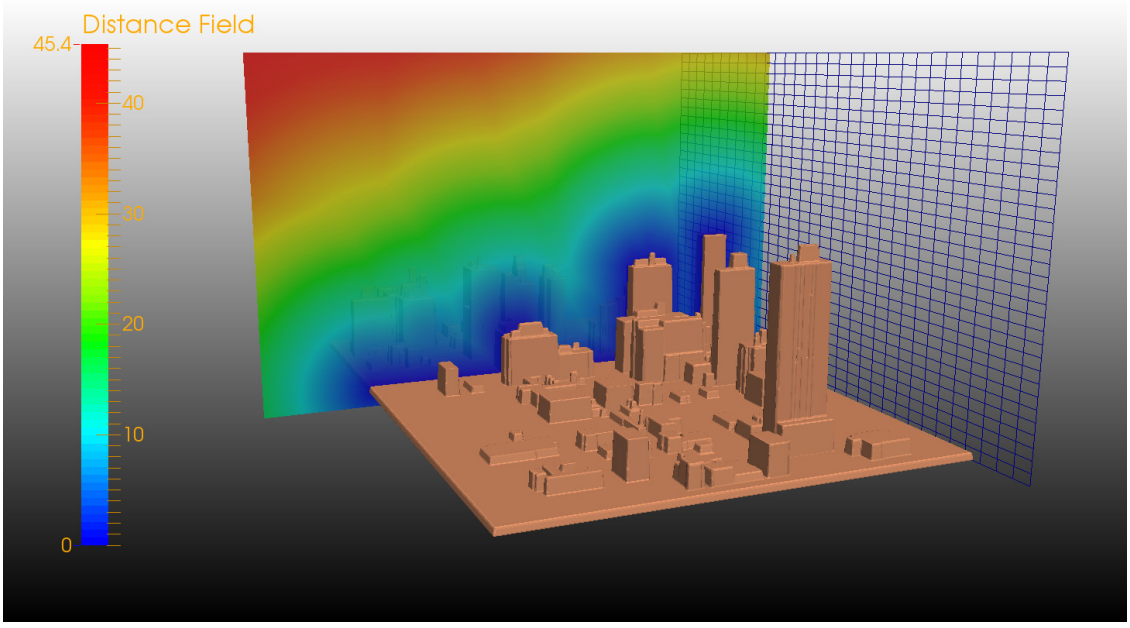
Figure 4.8: Placement of IB Nodes (blue squares) on Buckman Springs, CA. The IB nodes conform well to the STL surface. The fluid (green) points are all above the surface, and the solid (red) points are all below the surface.

shortest distance to the terrain is computed as expected.

Finally, we tested our procedure with different Cartesian mesh resolutions successfully. The bunny and the dragon geometries were both tested on meshes with 128^3 and 256^3 points. The size of the Cartesian domain was $100^3 m^3$ for the bunny, and $100 \times 60 \times 70 m^3$ for the dragon. The surface meshes had $88,753$ triangular elements with an average edge length of $0.53m$, and $66,991$ triangular elements with an average edge length of 0.27 for the bunny and dragon geometries, respectively.The Cartesian domain for the Buckman terrain covers a volume of $8800 \times 6800 \times 1270 m^3$ The Cartesian domain was discretized on meshes with $256 \times 192 \times 128$, 192^3, and $512 \times 256 \times 192$ points. The surface mesh for the terrain had 9962 triangular elements with an average edge length of 27.04m. For the urban environment shown in Figure 4.9b , we used a single 256^3 mesh for a Cartesian domain $80 \times 90 \times 50 m^3$ with a surface mesh of $987,523$ triangular elements with an average edge length of $0.12m$.



(a)



(b)

Figure 4.9: (a) Distance field propagation for the complex terrain of Buckman Springs (b) Distance field propagation for a portion of Oklahoma city urban environment. Cartesian mesh shows 1 of every 8 lines.

4.11 Conclusions

We presented a geometric preprocessor for arbitrarily complex geometry, terrain and urban environments that calculates the necessary geometric information needed to implement the Cartesian immersed boundary method within a flow solver. We identified potential discrepancies in published work, and addressed those issues with a new procedure that combines elements from various studies. The preprocessor classifies Cartesian mesh points as fluid, solid, and immersed boundary nodes. It then proceeds to calculate necessary information for reconstruction along a surface normal direction. Additionally, the preprocessor calculates the shortest distance to the surface from the Cartesian mesh nodes through the solution of the Eikonal equation with the fast sweeping method of Zhao (2004). We successfully demonstrated the versatility of the preprocessor for complex geometries, terrain, and urban environment. We expect our geometric procedure to help broaden the adoption of the immersed boundary method for atmospheric flows over arbitrarily complex terrain.

CHAPTER 5

Direct Numerical Simulation of Turbulent Katabatic Slope Flows with an Immersed Boundary Method

Umphrey, C., DeLeon, R., and Senocak, I. "Direct numerical simulation of turbulent katabatic slope flows with an immersed-boundary method." in-press for *Bound.-Lay. Meteorol.* (2017).

5.1 Abstract

We investigate a Cartesian mesh immersed boundary (IB) formulation within an incompressible flow solver to simulate laminar and turbulent katabatic slope flows. As a proof-of-concept study, we consider four different IB reconstruction schemes for imposing a Neumann type boundary condition on the buoyancy field. We use Prandtl's laminar solution to demonstrate the second-order accuracy of the numerical solutions globally. Direct numerical simulation (DNS) of turbulent katabatic flow is then performed to investigate the applicability of the proposed schemes in the turbulent regime by analyzing both first- and second-order statistics of turbulence. First-order statistics show that turbulent katabatic flow simulations are noticeably sensitive to the specifics of the IB formulation. We find that reconstruction schemes that work well in the laminar regime may not perform as well when applied to a turbulent regime. We propose an IB reconstruction scheme that produced results in close agreement with the terrain-fitted reference solutions in both flow regimes.

5.2 Introduction

Turbulent katabatic flows with stable stratification commonly occur in the atmospheric boundary layer with important implications for weather forecasting, air pollution and wind energy. The subject has been the focus of several numerical and experimental investigations aimed at establishing a better understanding of the underlying flow physics (Mahrt, 2014).

Skyllingstad (2003) conducted large-eddy simulations (LES) of katabatic flows to demonstrate the effect of terrain slope changes on drainage winds that can hinder formation of very stable regimes. Cuxart *et al.* (2007) carried out mesoscale simulations of an island in the Mediterranean to study the interaction of different scales of motion during the night time. Papadopoulos and Helmis (1999) investigated evening and morning transition of katabatic slope flows experimentally and elucidated the processes involved in their formation and desctruction. Mahrt (2016) studied stably stratified flow in a shallow valley experimentally, and identified oscillations within the valley with a period of 10 min. Shapiro and Fedorovich (2007) considered inhomogeneity of sloping surfaces by developing an analytical model in which buoyancy varied linearly down the slope. Shapiro and Fedorovich observed physical processes that are not accounted for in the Prandtl's laminar solution (Prandtl, 1942).

There are many physical processes involved in a katabatic flow over complex terrain with heterogeneous surface conditions (Mahrt, 2014). It can be difficult to unravel the interplay of these processes without introducing certain simplifications. To this end, an inclined, homogeneous flat terrain assumption can serve as an idealized test bed to develop a fundamental understanding of katabatic flows. An inclined flat surface is also a realistic simplification to study katabatic flows in the Great Plains of the United States and Antarctica (Parish and Waight III, 1987). Equally important, slope flows over homogeneous surfaces serve as a canonical test case to validate new numerical methods and physical models, which is our primary interest in the present study.

In the present study our motivation is to evaluate and refine an immersed boundary (IB) approach for katabatic slope flows. The IB method is an attractive approach to represent complex geometry on a Cartesian mesh without the need for a manual mesh generation process. The simplicity of the Cartesian formulation also eases the computer parallelization of the flow solver. The method, pioneered by Peskin (1972), traces its origins back to the numerical simulation of blood flow through a heart. The essence of the IB method is to immerse a geometry inside a Cartesian mesh and impose surface boundary conditions upon

the nodes of the intersected cells using either a forcing function or a reconstruction scheme. The IB method has been an active area of research in computational fluid dynamics (CFD) analysis of engineering applications. IB method has also been proposed as an alternative meshing approach for geophysical flows (Tseng and Ferziger, 2003; Senocak *et al.*, 2004; Smolarkiewicz *et al.*, 2007).

Various approaches have been proposed to implement the IB method (Mittal and Iaccarino, 2005; Sotiropoulos and Yang, 2014). Despite the importance of heat transfer in many engineering or geophysical flows, majority of IB studies have focused on velocity boundary conditions with little coverage given to boundary conditions related to scalar transport. Boundary conditions can be imposed as Dirichlet (i.e. prescribed temperature), Neumann (i.e. prescribed heat flux) or a mixed type. Reconstruction schemes developed for velocity boundary conditions can be applied to impose a prescribed temperature boundary condition. On the other hand, there are not that many studies focusing on the implementation of Neumann type boundary conditions within an IB formulation. Pacheco-Vega *et al.* (2007) describe a generalized immersed boundary approach applicable to both the Dirichlet and Neumann type boundary conditions. For the Neumann type, a numerical stencil is created around a surface point where the Neumann boundary condition is required, not necessarily aligning with the numerical stencils on the Cartesian grid. Therefore, linear and bilinear interpolations are used to bring neighboring values to the imposed numerical stencil before calculating the Neumann boundary condition. Zhang *et al.* (2008) use a feedback-forcing type IB method where values at neighboring grid points are interpolated on to a surface point to determine heat transfer from the surface of a cylinder. Kang *et al.* (2009) performed a DNS of conjugate heat transfer consisting of a solid cylinder in a channel flow with a heated bottom plane using the IB reconstruction found in Fadlun *et al.* (2000). A Dirichlet type boundary condition was imposed for temperature field by linearly interpolating the temperature in the solid to the fluid boundary. Neumann boundary condition for temperature was satisfied by a one-sided finite difference in the fluid domain.

Our chief objective in this study is to develop an IB reconstruction scheme to impose a Neumann type boundary condition on the buoyancy field, and assess its performance in both laminar and turbulent flow regimes. To date, the issue of consistent performance in both flow regimes has not received any attention. To achieve our objectives we consider an idealized katabatic flow above an inclined surface as a canonical problem. Admittedly, an IB method does not provide any obvious advantage over a body-fitted mesh to simulate a flat surface. However, an idealized slope flow establishes a benchmark case that can be simulated with a body-fitted approach, which can then be used to assess the formal accuracy of reconstruction schemes and their predictive performance in both laminar and turbulent flow regimes.

5.3 Numerical Formulation

Cartesian grid nodes generally do not coincide with the immersed geometry or terrain. Therefore, a challenge in the IB method is to indirectly enforce the boundary conditions on the discretized form of the governing equations. The original IB method introduced a body force term in the momentum equations (Peskin, 1972). The method worked well for elastic surfaces, but issues with numerical stability limited the applicability of the original method. Mohd-Yusof (1997) proposed the *direct forcing* approach that alleviated numerical stability constraints by indirectly incorporating a body force term, F_i, into the discretized for of the momentum equations as

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = RHS_i + F_i, \tag{5.1}$$

where RHS_i includes advective, diffusive, and pressure gradient terms. If the velocity at the boundary can be prescribed as $u_i^{n+1} = V_i$ for all n, then the body force becomes

$$F_i = -RHS_i + \frac{V_i - u_i^n}{\Delta t} \tag{5.2}$$

from Eq. (5.1). Substituting Eq. (5.2) into Eq. (5.1), the body force can be taken into account implicitly by prescribing the velocity field, V_i. As the Cartesian grid is not coincident with the immersed surface, a reconstruction scheme is required to impose the boundary conditions on the Cartesian nodes near the surface, which in turn become boundary conditions for the rest of the flow.

In the present study, we adopt the GIN3D flow solver (Thibault and Senocak, 2012; Jacobsen and Senocak, 2011, 2013; DeLeon *et al.*, 2013). An IB method based on the aforementioned direct forcing approach was implemented in GIN3D for the velocity field (DeLeon *et al.*, 2012). A separate geometric preprocessor is available to immerse an arbitrarily complex geometry into a Cartesian grid (Senocak *et al.*, 2015). In the GIN3D solver, the velocity reconstruction scheme for laminar conditions follows a linear approach along the surface normal direction similar to the works of Gilmanov *et al.* (2003), Gilmanov and Sotiropoulos (2005) and Gilmanov and Acharya (2008). In wall-resolved LES of turbulent flows, in which the viscous sublayer is resolved, the same linear reconstruction schemes can be used albeit at modest Reynolds numbers and with a high computational cost. However, for wall-modeled LES, the linear reconstruction of the no-slip boundary condition is not adequate because spatial resolution is too coarse to resolve any part of the viscous sublayer. In those cases, GIN3D adopts the logarithmic reconstruction scheme proposed by Senocak *et al.* (2004) along with a subgrid-scale model (Senocak *et al.*, 2007) that is consistent with the logarithmic law-of-the-wall theory (Pope, 2000).

Figure 5.1 shows a schematic describing the implementation of the boundary reconstruction scheme where a line in the normal direction from the nearest triangular element of the immersed boundary (point a) is projected through the IB node, which is defined as the nearest fluid node to the surface (point b), and onto a Cartesian cell face in the fluid domain (point c). This line will be referred to as the IB line. Values at a are known as prescribed boundary conditions and values at c are reconstructed by bilinear interpolation from the neighboring Cartesian grid nodes marked by Greek letters α, β, γ, and δ. In the case of

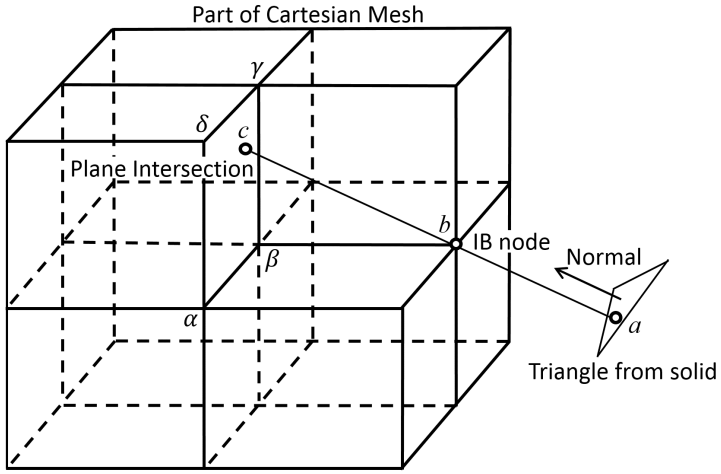


Figure 5.1: Sketch of the general indirect boundary reconstruction scheme at an IB node by projecting a line in the normal direction from the nearest triangular element of the boundary into the fluid domain.

linear velocity reconstruction, another linear interpolation along the IB line between a and c is performed to overwrite the current value at b. We emphasize that a linear reconstruction for velocity is appropriate for the present problem because the simulation resolves the flow field down to the surface.

5.4 Buoyancy Reconstruction Schemes

We investigate four reconstruction schemes to impose a Neumann type boundary condition within the context of a direct-forcing IB approach. The buoyancy quantity is solved for convenience of imposing boundary conditions for katabatic slope flows. Following the work of Fedorovich and Shapiro (2009), we prescribe a constant buoyancy flux at the surface to establish a canonical test case for both laminar and turbulent flow regimes. We note that constant surface buoyancy flux may not be ideal to study physics of actual katabatic flows over sloped terrain, but it serves the purpose of the present study well. A spatially-varying surface buoyancy flux (Shapiro and Fedorovich, 2007) could also be considered.

Scheme 1 makes use of the buoyancy gradient boundary condition at the surface and buoyancy gradient in the fluid node calculated with a central difference formula. These two

data points are then used to calculate the buoyancy at the IB node. Gilmanov *et al.* (2003) proposed this approach to impose a Neumann type boundary condition on the pressure field for an incompressible solution but it was not tested for the transport of a generic scalar field. Scheme 1 is outlined in the following steps for a general scalar ϕ with the help of Fig. 5.1:

1. Calculate $\left(\frac{\partial \phi}{\partial x_i}\right)$ at α-β-γ-δ in all three Cartesian coordinate directions ($i = 1, 2, 3$).
2. Linear interpolate these to c to get $\left(\frac{\partial \phi}{\partial x_i}\right)_c$.
3. Then the normal gradient of ϕ at c is

$$\left(\frac{\partial \phi}{\partial n}\right)_c = \mathbf{n}_a \cdot (\nabla \phi)_c \,, \tag{5.3}$$

   where $\mathbf{n}_a$ is the surface normal vector at a.

4. With the normal ϕ gradient boundary condition at the surface, $\left(\frac{\partial \phi}{\partial n}\right)_a$, and the known normal ϕ gradient at c, $\left(\frac{\partial \phi}{\partial n}\right)_c$, linear interpolate along the IB line to the midpoint between b and c to determine $\left(\frac{\partial \phi}{\partial n}\right)_{bc}$.
5. Linear interpolate to bring ϕ from α-β-γ-δ to c, giving ϕ_c.
6. The value of ϕ at the IB node is then calculated using a central difference approximation

$$\phi_b = \phi_c - \Delta s_{bc} \left(\frac{\partial \phi}{\partial n}\right)_{bc} , \tag{5.4}$$

   where Δs_{bc} is the distance along the IB line from b to c.

In step 1, Gilmanov *et al.* (2003) does not mention a method for calculating the first derivatives, so we approximate those derivatives with a second order accurate central difference formula because the scheme was originally applied to the pressure field. The central differ-

ence formula to compute the first derivative of ϕ with respect to z can be written as

$$\left(\frac{\partial \phi}{\partial z}\right)_k = \frac{\phi_{k+1} - \phi_{k-1}}{2\Delta z}, \tag{5.5}$$

where k represents grid index, and Δz is the uniform mesh spacing in the z direction. We note that Scheme 1 has not been validated for the transport of a generic scalar field. The remaining reconstruction schemes are unique to our present work to the best of our knowledge. In designing these reconstruction schemes we devote special attention to include only computational cells that are not intersected by the immersed boundary to avoid including unreconstructed values at the IB node. We find that this subtle issue is important and was not addressed in Scheme 1.

Scheme 2 assumes the ϕ gradient along the IB line from a to c is constant. Analogous to Eq. (5.4), a central difference using the ϕ gradient boundary condition is used to calculate the buoyancy at the IB node,

$$\phi_b = \phi_c - \Delta s_{bc} \left(\frac{\partial \phi}{\partial n}\right)_a. \tag{5.6}$$

Scheme 3 is identical to Scheme 1, except first derivatives in Step 1 are approximated with a one-sided difference formula, to avoid including IB nodes in the reconstruction, because values at those nodes may yet to be updated and can introduce errors into the scheme otherwise. We use a second-order-accurate, one-sided difference formula (Tannehill *et al.*, 1997), which can be written for a 1D uniform mesh as follows:

$$\left(\frac{\partial \phi}{\partial z}\right)_k = \frac{-3\phi_k + 4\phi_{k+1} - \phi_{k+2}}{2\Delta z}. \tag{5.7}$$

Scheme 4 adopts a nearest neighbor interpolation to compute the ϕ gradient at point b. To calculate ϕ at the IB node, a second-order-accurate, one-sided difference is used that includes the ϕ gradient at b and two cell face intersubsection points from the fluid side. This

reconstruction requires an additional fluid point not shown in Fig. 5.1, referred to here as d, where the IB line intersects the next cell face after point c. Values of ϕ are brought to d through linear interpolation from neighboring Cartesian grid points in the same way values are brought to c through linear interpolation from α, β, γ, and δ. The reconstruction is performed by solving for ϕ_b in the following equation:

$$\left(\frac{\partial\phi}{\partial n}\right)_b = \frac{-\phi_d\left(h_c - h_b\right)^2 + \phi_c\left(h_d - h_b\right)^2 - \phi_b[\left(h_d - h_b\right)^2 - \left(h_c - h_b\right)^2]}{\left(h_c - h_b\right)\left(h_d - h_b\right)\left(h_d - h_c\right)}, \tag{5.8}$$

where h refers to the distance from the surface of each respective point b, c, and d. Equation (5.8) is the version of Eq. (5.7) for non-uniform grid spacing (Ferziger and Perić, 2012).

A fifth scheme was also proposed in Umphrey *et al.* (2016). Combination of these schemes with different velocity reconstruction schemes was investigated in that study as well. We leave those options out in the present study for clarity and sake of space.

5.5 Results

We simulate katabatic flows above an infinite plane inclined at slope angle α to validate the aforementioned immersed boundary reconstruction schemes for imposing a Neumann type boundary condition for the buoyancy field. A one-dimensional model for laminar flow of a stratified fluid along a uniformly cooled or heated sloping plane was developed by Prandtl (1942). Figure 5.2 illustrates an instance from Prandtl's laminar solution for the case of a uniformly cooled sloping plane. A katabatic flow develops with a low-level down-slope jet topped by a weak up-slope return flow. Both velocity and bouyancy fields approach zero away from the surface. Fedorovich and Shapiro (Shapiro and Fedorovich, 2004; Fedorovich and Shapiro, 2009) published a non-dimensional form of the Prandtl model solution with

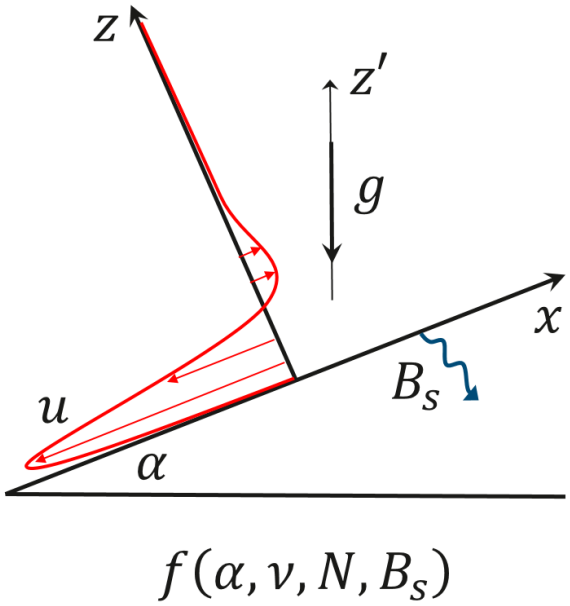


Figure 5.2: Sketch of an instance of the Prandtl's laminar solution of katabatic flow on an infinite plane with constant surface buoyancy flux. Note that in the simulations the gravity vector is rotated and a flat surface is considered.

the following formulation:

$$u_n = sin(z_n/\sqrt{2})exp(-z_n/\sqrt{2}), \tag{5.9}$$

$$b_n = cos(z_n/\sqrt{2})exp(-z_n/\sqrt{2}), \tag{5.10}$$

where

$$z_n = z\nu^{-1/2}N^{1/2}sin^{1/2}\alpha, \tag{5.11}$$

$$u_n = u\nu^{1/2}N^{3/2}B_s^{-1}sin^{1/2}\alpha, \tag{5.12}$$

$$b_n = b\nu^{1/2}N^{3/2}B_s^{-1}sin^{1/2}\alpha. \tag{5.13}$$

In the above equations, subscript n refers to normalised quantities, u is velocity parallel to the slope, z is the normal distance from the slope surface, b is buoyancy ($b = g\theta/\Theta_r$, g is the gravitational acceleration, θ is the potential temperature perturbation, Θ_r is the constant

reference potential temperature value), ν is the kinematic viscosity equal to the thermal diffusivity, N is the $Brunt - V\ddot{a}is\ddot{a}l\ddot{a}$ (or buoyancy) frequency ($N^2 = (g/\Theta_r)\left(d\Theta_e/dz^{'}\right)$, Θ_e is the environmental potential temperature), B_s is the surface buoyancy flux ($B_s = -\nu\left(db/dz\right)|_{z=0}$), and α is the slope angle. Furthermore, this analytical solution has the boundary conditions $u(0) = 0$, $\left(db_n/dz_n\right)|_{z_n=0} = -1$, and $u_n \to 0$, $b_n \to 0$ as $z_n \to \infty$. The above analytical solution serves as an excellent benchmark to evaluate various numerical methods and formal order of accuracy of numerical solutions. However, it does not guarantee success in the turbulent regime, as we show in the present study.

It is common to include buoyancy effects in the incompressible Navier-Stokes equations through the Boussinesq approximation, which is valid when density variations are small and vary linearly with temperature difference (Ferziger and Perić, 2012). Following the work of Fedorovich and Shapiro (2009) the momentum balance equations for the slope flow can be written as follows:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) + b\sin\alpha, \tag{5.14}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} = -\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right), \tag{5.15}$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} = -\frac{\partial p}{\partial z} + \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) + b\cos\alpha, \tag{5.16}$$

For convenience, the heat balance equation is written in terms of buoyancy

$$\frac{\partial b}{\partial t} + u\frac{\partial b}{\partial x} + v\frac{\partial b}{\partial y} + w\frac{\partial b}{\partial z} = \nu\left(\frac{\partial^2 b}{\partial x^2} + \frac{\partial^2 b}{\partial y^2} + \frac{\partial^2 b}{\partial z^2}\right) - N^2\left(u\sin\alpha + w\cos\alpha\right). \tag{5.17}$$

In the above equations, u, v, and w are velocity components in the up-slope, cross-slope, and slope-normal directions, respectively; p is pressure, $b\sin\alpha$ and $b\cos\alpha$ are buoyancy forcing terms. The Prandtl number is assumed to be unity.

Quantity	Symbol (Units)	Laminar Value	Turbulent Value
Slope angle	α (degrees)	30	60
Surface buoyancy flux	B_s (m^2 s^{-3})	-0.005	-0.5
$Brunt - Väisälä$ frequency	N (s^{-1})	1	1
Kinematic viscosity	ν (m^2 s^{-1})	0.0005	0.0001
Thermal diffusivity	γ (m^2 s^{-1})	0.0005	0.0001

Table 5.1: Simulation parameters for both laminar and turbulent katabatic slope flows.

5.5.1 Laminar Katabatic Flow Simulation

We adopt the nondimensional form of Prandtl's laminar solution to assess four different IB reconstruction schemes. We simulate an infinite flat plate with a slope angle of $\alpha = 30°$. To establish a body-fitted solution within a Cartesian flow solver, the gravity vector, instead of the horizontal surface, is rotated according to the slope angle. The flat plate is placed at the bottom of the domain such that IB Cartesian scalar nodes (staggered grid) were 0.25Δ above the flat plate, where Δ is the uniform grid spacing. The domain height in the z-direction is $H = 1.27$ m. The no-slip condition and a constant buoyancy flux are enforced at the immersed surface while the top of the domain receives a free-slip condition for velocity and zero-flux condition for buoyancy. Lateral boundaries are assumed to be periodic. Relevant simulation parameters are presented in Table 5.1. Three mesh spacings (Δ) are used throughout the laminar katabatic slope study: 0.005 m, 0.0025 m, 0.00125 m.

Figure 5.3 shows comparison for the normalised down-slope velocity for the schemes discussed in Sect. 5.4 using mesh spacing of $\Delta = 0.005$ m. Schemes 2–4 agree very well with the analytical solution. However, Scheme 1 as proposed by Gilmanov *et al.* (2003) clearly falls short for the peak velocity of the low level jet. In Step 1 of Scheme 1, buoyancy gradient approximation in the slope-normal direction of the fluid domain is performed using a central difference stencil that includes the IB node. This is problematic as the IB node needs to be reconstructed before it can be used in a stencil. Scheme 3 resolves this issue by using a second order accurate one-sided finite difference scheme that does not include the IB node. Central differences are still used in directions parallel to the slope as those stencils do not

include IB nodes.

Figure 5.4 shows the normalised buoyancy produced by the same schemes compared to the analytical solution. Discrepancies in buoyancy are less apparent, but Scheme 1 shows lower buoyancy values near the surface compared to the benchmark solution. Based on these results we exclude Scheme 1 from the rest of the present study.

A grid convergence study is often performed to identify the formal accuracy of a numerical solution. In computational analysis, a globally second-order-accurate solution is desirable at a minimum. To this end, we investigated the order of accuracy of Scheme 4 using three systematically refined grids as shown in Fig. 5.5. As expected from an L_∞ norm of numerical error, first order accuracy is achieved locally at the IB node. However, the global solution is second order accurate as evident by the slopes of the L_1 and L_2 norms. Formal accuracy of Scheme 2 and 3 was also studied in Umphrey *et al.* (2016) with the same outcome as Scheme 4. Note that we do not expect an order of accuracy higher than two, because the baseline flow solver adopts second order numerics in time and space.

5.5.2 Turbulent Katabatic Flow, First-Order Statistics

We hypothesize that reconstruction schemes that produce satisfactory results in the laminar regime may not readily apply to turbulent flow regimes. To validate our hypothesis, we consider Prandtl's model for katabatic slope flows in the turbulent regime. Because Scheme 1 failed to reproduce Prandtl's laminar solution, we only consider Schemes 2–4 in this subsection. Fedorovich and Shapiro (2009) performed DNS of idealized turbulent slope flows, in which they derived an integral slope-flow Reynolds number as $Re_I = |B_s|/\nu N^2 \sin\alpha$. Values of Re_I greater than 3000 are considered to have reasonably developed turbulence. For reference, the laminar simulation in Sect. 5.5.1 had an $Re_I = 20$. The flat plate is once again placed at the bottom of the domain at 0.25Δ below the scalar node in the staggered grid configuration. Relevant flow parameters are given in Table 5.1. The size of the domain is 0.64 m $\times$ 0.64 m $\times$ 1.6 m in the x-direction, y-direction and z-direction, respectively.

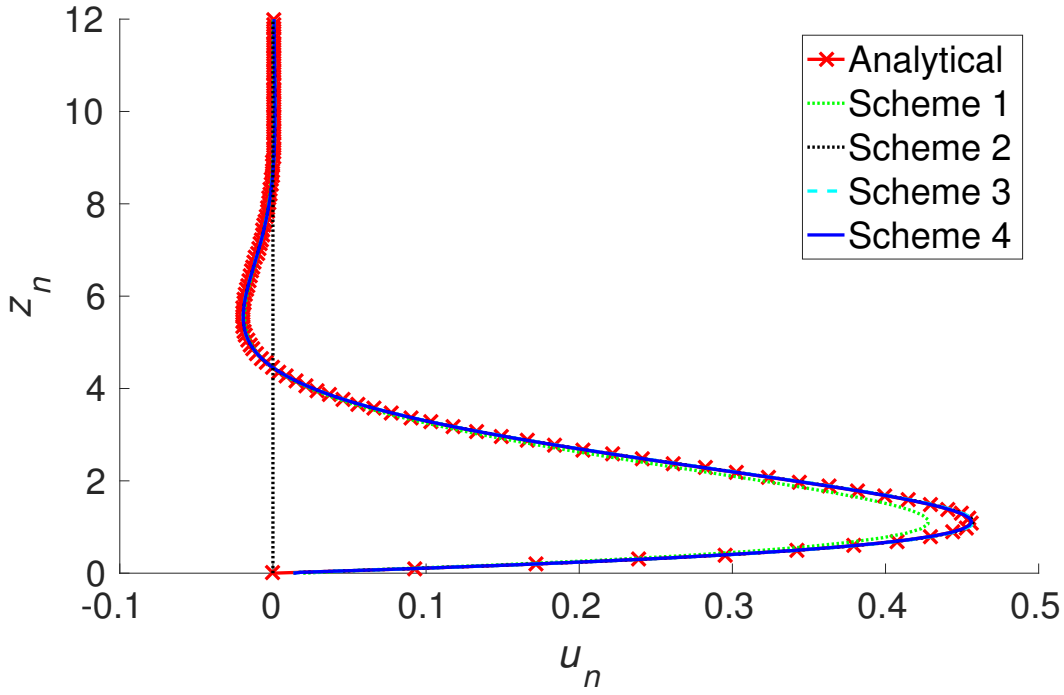


Figure 5.3: Normalized down-slope velocity. Comparison of results from different buoyancy reconstruction schemes with the analytical solution of the Prandtl model for laminar katabatic flow.

The x-direction, y-direction and z-direction correspond to the down-slope, cross-slope and slope-normal directions, respectively. Boundary conditions are the same as those used in the laminar case (see Sect. 5.5.1). The grid dimensions are $(NX \times NY \times NZ) = 257 \times 257 \times 641$ with uniform spacing of $\Delta = 0.00025$ m. The resulting integral slope-flow Reynolds number is $Re_I = 5773$. Given the analog of the Kolmogorov microscale $L_m = \nu^{3/4}|B_s|^{-1/4}$, the resolvability condition for DNS (Pope, 2000) is met by $\Delta \leq 2L_m$.

The performance of IB schemes are assessed relative to simulations performed with a body-fitted mesh using the same flow solver. Additionally, a second reference solution from Fedorovich and Gibbs (2015),designated as FG, and a third reference solution produced by the MicroHH code (van Heerwaarden *et al.*, 2016) as presented in Fedorovich and Gibbs (2015) are included as additional body-fitted mesh reference solutions to better assess the sensitivities of the simulations to numerical methods. Figure 5.6 shows a volume rendering of the instantaneous velocity magnitude obtained with DNS. The comparison of mean ve-

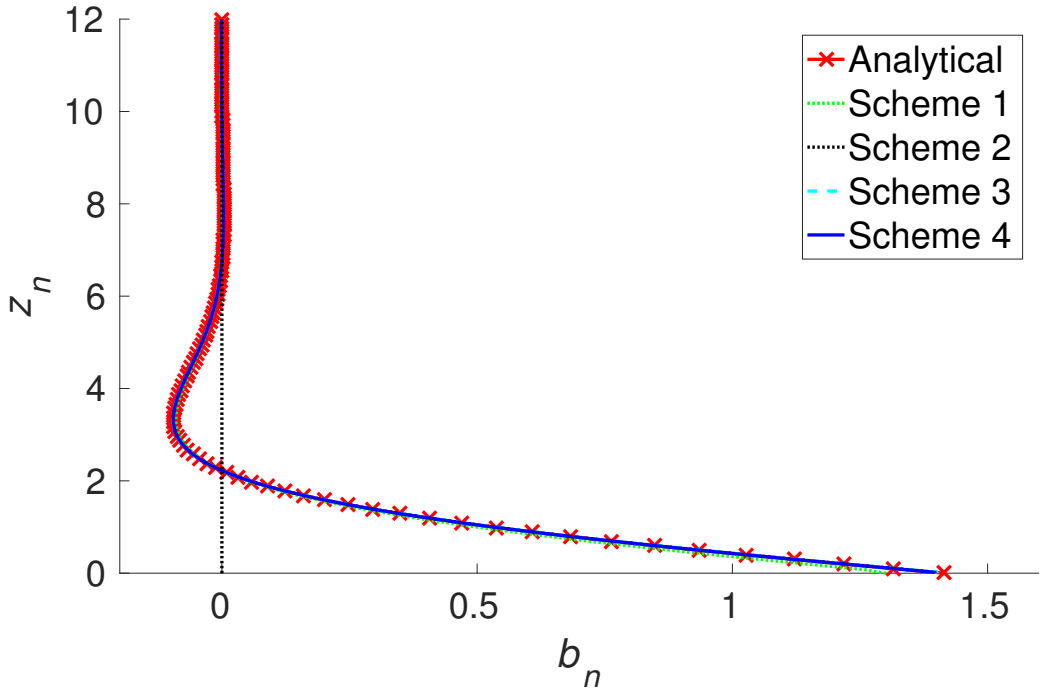


Figure 5.4: Normalized buoyancy. Comparison of results from different buoyancy reconstruction schemes with the analytical solution of the Prandtl model for laminar katabatic flow.

locity profiles is shown in Fig. 5.7. Although these schemes gave good agreement with the analytical solution in laminar flow simulations, errors in the peak velocity of the low level jet are easily seen for Schemes 2 and 3 for the turbulent case, validating our hypothesis on consistent performance in both flow regimes. Scheme 4 agrees well with the body-fitted mesh simulation and deviates slightly near the up-slope return flow region. As in the laminar case, discrepancies in buoyancy are less apparent in Fig. 5.8; however, Scheme 4 agrees well with the body-fitted mesh simulation.

5.5.3 Turbulent Katabatic Flow, Second-Order Statistics

Of the three schemes applied to the turbulent katabatic slope flow, Scheme 4 exhibited the best agreement with the first-order statistics of the body-fitted mesh reference solution. Therefore, only Scheme 4 is used for investigation of the second-order statistics. The first

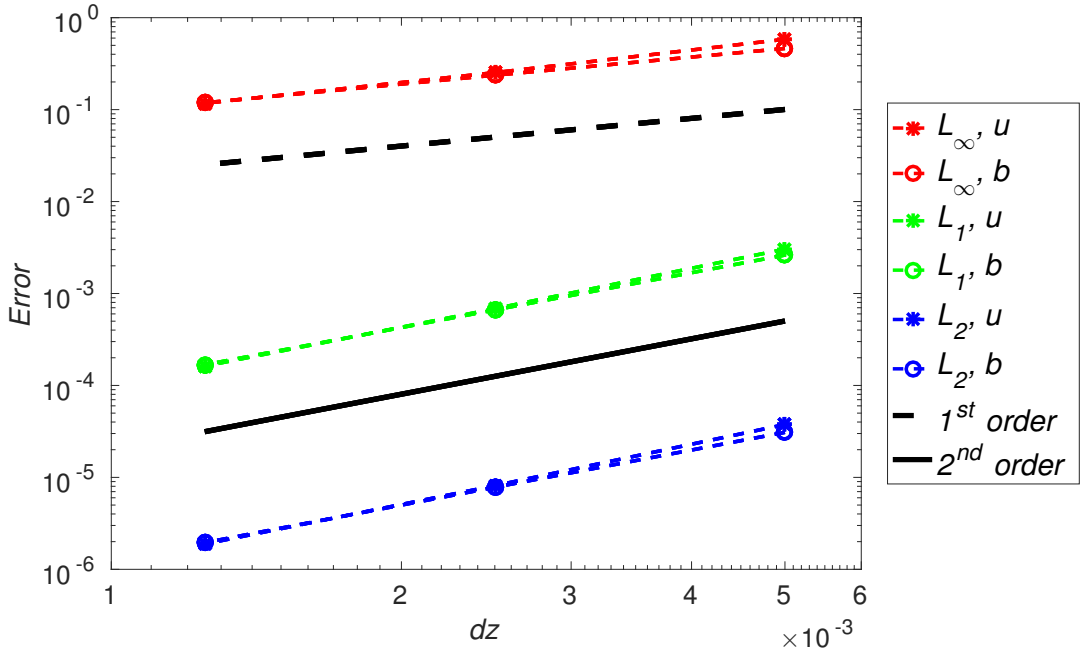


Figure 5.5: Grid convergence study of Scheme 4. The L_1 and L_2 norms show second order accuracy globally for both velocity and buoyancy fields. The L_∞ norm shows first order accuracy locally at the IB node.

quantity presented is the slope-normal buoyancy flux in Fig. 5.9. Scheme 4 agrees well with the body-fitted mesh approaches, deviating slightly from the reference solution in the up-slope return flow region. All four approaches exhibit similar trends with slight variations in this region. The slope-normal momentum flux in Fig. 5.10 shows a similar variation among the four approaches as well. Scheme 4 agrees best with FG in the low-level jet and best with our body-fitted approach in the up-slope return flow.

In Fig. 5.11, variation among the four approaches is more pronounced than variations observed in other figures. The comparison suggests that the down-slope velocity variance in the vicinity of the surface is sensitive to the details of the numerical approach. In the up-slope return flow region and above, the agreement of Scheme 4 with the other approaches is quite good. Of the other variances shown in Figs. 5.12, 5.13 and 5.14, all four approaches have excellent agreement among themselves. A critical inspection of Scheme 4 reveals the peaks near the surface for cross-slope and slope-normal velocity variances are slightly the

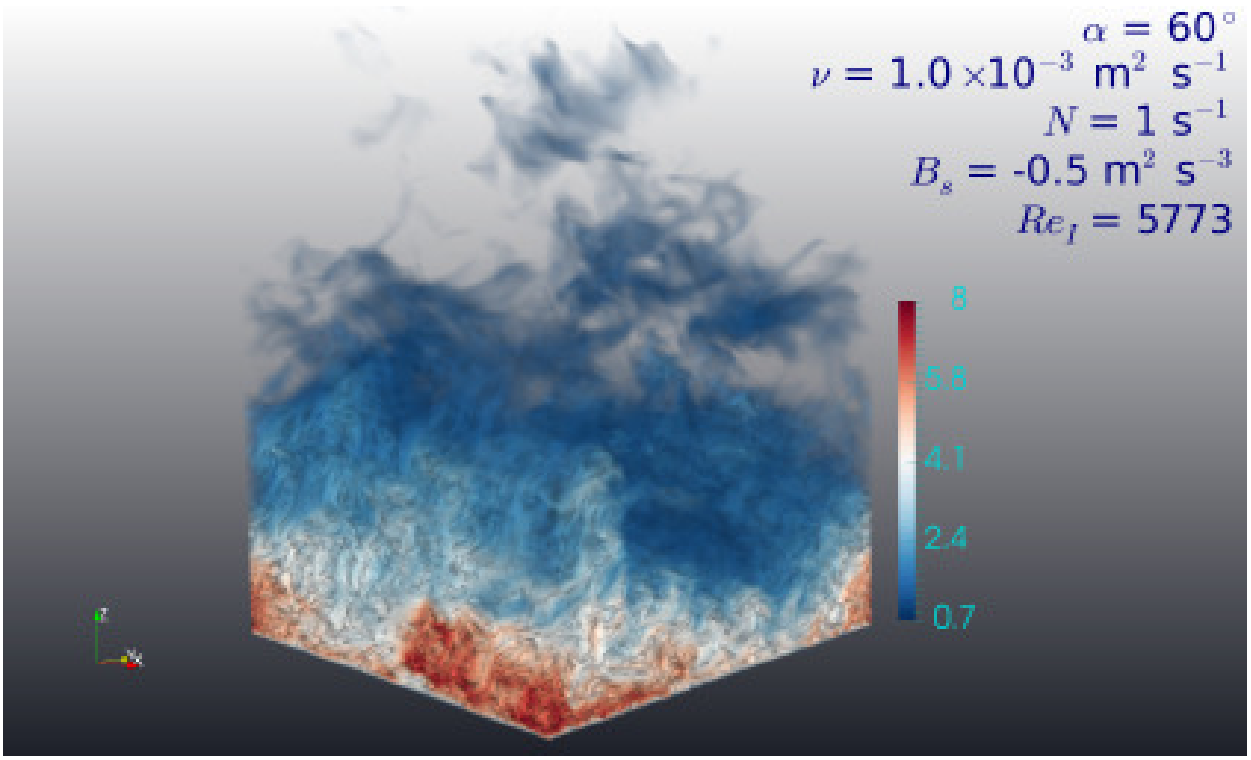


Figure 5.6: 3D volume rendering of instantaneous velocity magnitude from direct numerical simulation of turbulent katabatic flow. The simulation is for an infinite plane inclined at slope angle $\alpha = 60^o$, however, the slope is rotated down into an isometric view here to better show the turbulence throughout the domain.

lowest in magnitude compared to the other three DNS approaches. In the cross-slope and slope-normal velocity variances in Figs. 5.12 and 5.13, Scheme 4 produces the lowest variance magnitude in the up-slope return flow region and above, although this difference from the other three approaches can be assumed negligible. The buoyancy variance in Fig. 5.14 shows very little variation among the different solutions considered in the present study. Overall, the agreement of Scheme 4 with the other three approaches is very good in all the second-order statistics and encourages us to consider it in a future work to study turbulent katabatic flows over complex terrain with heterogeneous surface boundary conditions.

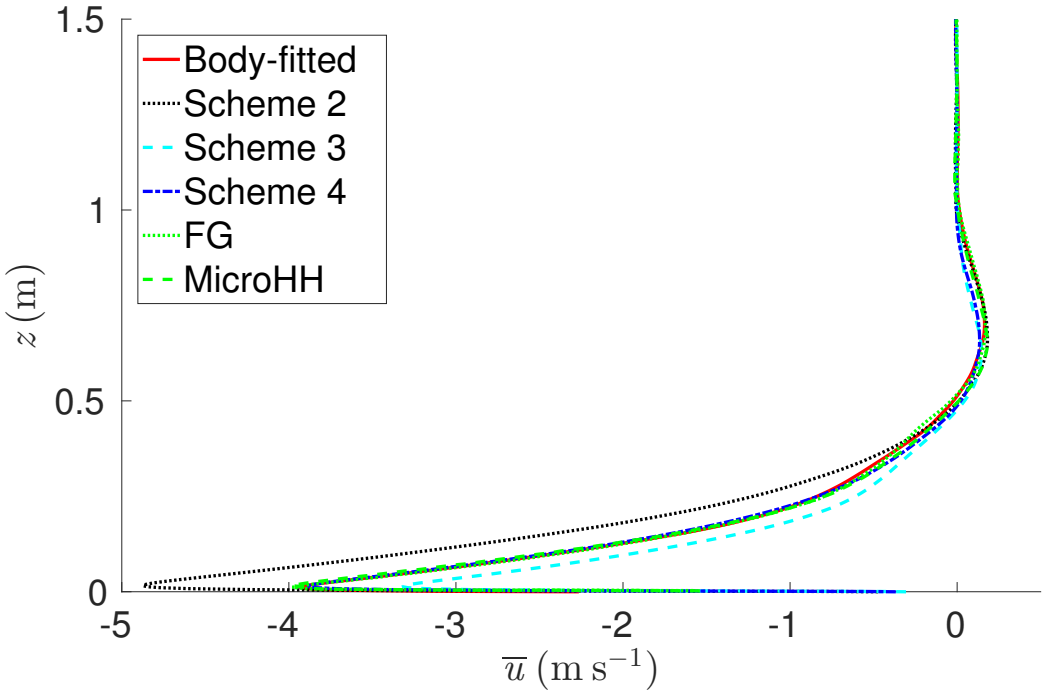


Figure 5.7: Down-slope velocity in a turbulent katabatic flow. Comparison of IB reconstruction scheme results and results from three different DNS studies using body-fitted meshes.

5.6 Conclusions

We studied the performance of four immersed boundary reconstruction schemes as a proof-of-concept for imposing a Neumann type boundary condition on the buoyancy field of an idealized katabatic slope flow. We have shown that the choice of IB reconstruction scheme for Neumann boundary conditions has a significant impact on the accuracy of the results. Despite having the same formal order of accuracy globally, second-order turbulence statistics obtained from direct numerical simulations showed significant differences among the schemes considered. Therefore we suggest validating IB reconstruction schemes in both laminar and turbulent flow regimes.

In designing an IB reconstruction scheme, we paid particular attention to exclude neighboring IB nodes in the reconstruction of a quantity at the IB node. We find this subtle issue to be important even for the laminar flow case. Consequently, we developed a reconstruction

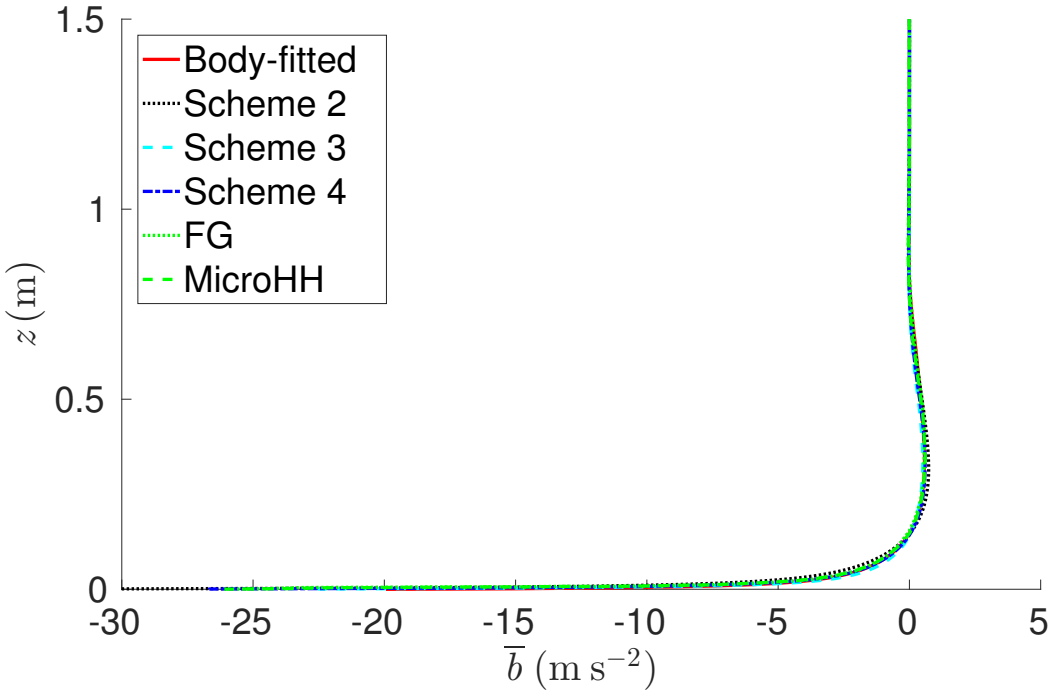


Figure 5.8: Buoyancy in a turbulent katabatic flow. Comparison of IB reconstruction scheme results and results from three different DNS studies using body-fitted meshes.

scheme (i.e. Scheme 4) for imposing a Neumann type boundary condition at an immersed boundary. *Scheme 4* performed well in both the laminar and turbulent flow regimes. First- and second-order turbulence statistics obtained from direct numberical simulations agree well with reference body-fitted solutions. The current flat terrain simulation constitutes a proof-of-concept to demonstrate critical issues pertinent to the IB method. In future work we will focus on extending and testing the proposed scheme for complex terrain with heterogeneous surface conditions.

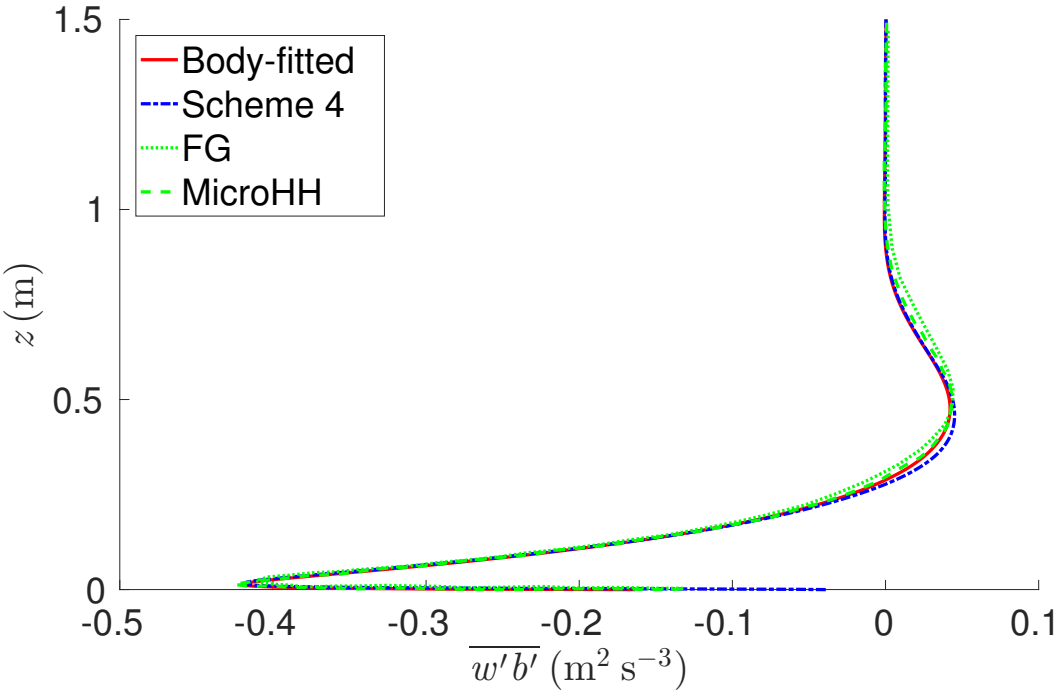


Figure 5.9: Slope-normal buoyancy flux in a turbulent katabatic flow. Comparison of IB reconstruction Scheme 4 with three different DNS studies using body-fitted meshes.

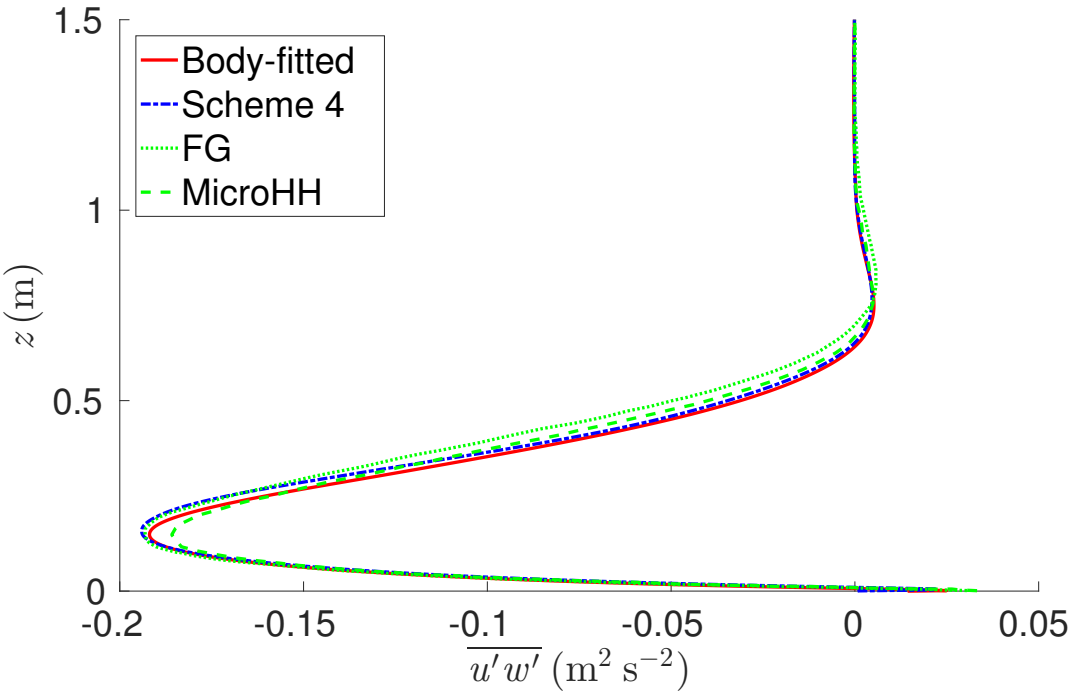


Figure 5.10: Slope-normal momentum flux in a turbulent katabatic flow. Comparison of IB reconstruction Scheme 4 with three different DNS studies using body-fitted meshes.

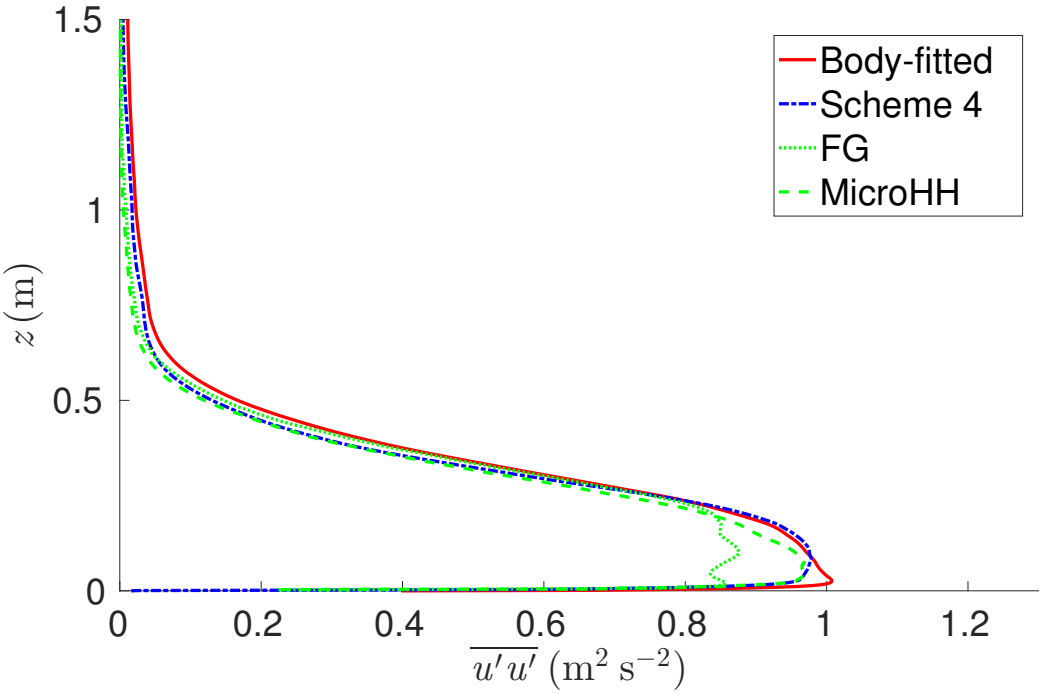


Figure 5.11: Down-slope velocity variance in a turbulent katabatic flow. Comparison of IB reconstruction Scheme 4 with three different DNS studies using body-fitted meshes.

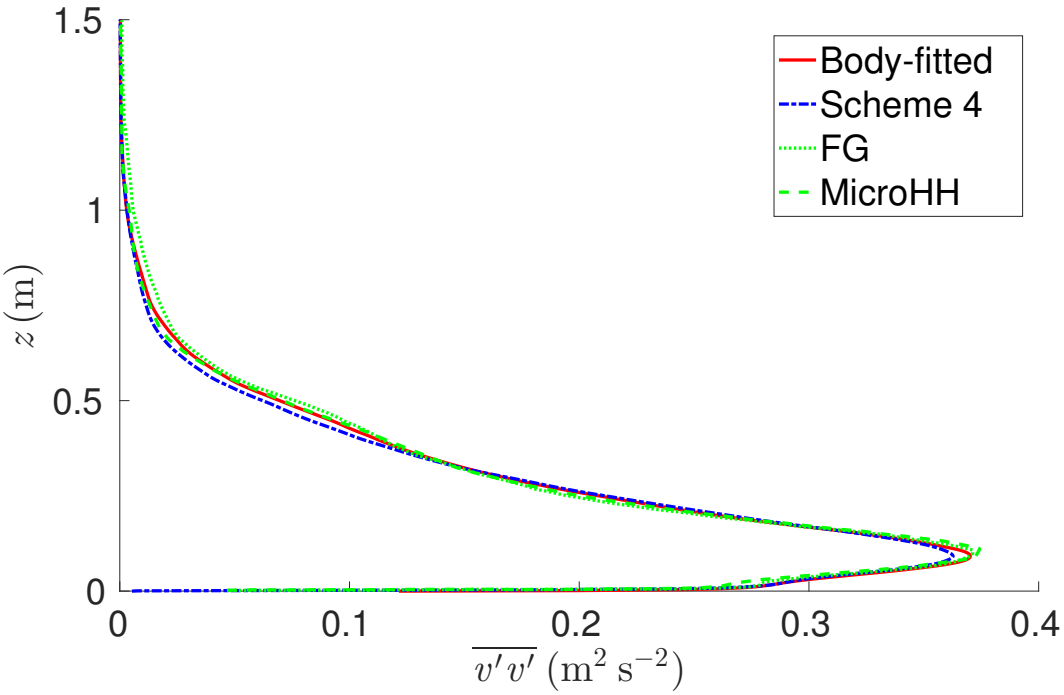


Figure 5.12: Cross-slope velocity variance in a turbulent katabatic flow. Comparison of IB reconstruction Scheme 4 with three different DNS studies using body-fitted meshes.

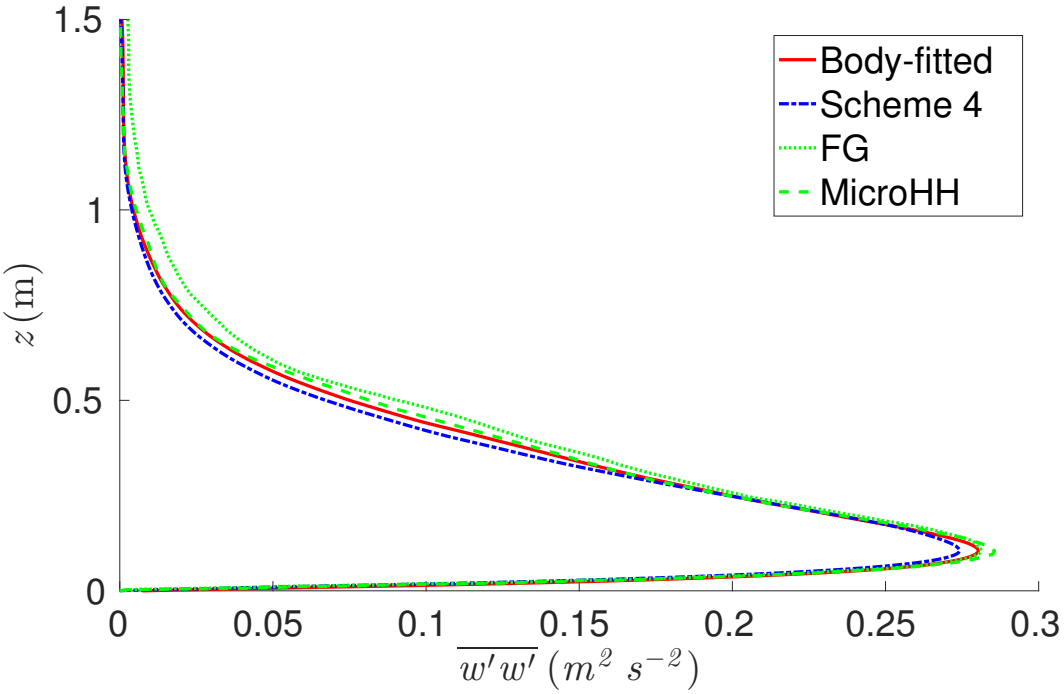


Figure 5.13: Slope-normal velocity variance in a turbulent katabatic flow. Comparison of IB reconstruction Scheme 4 with three different DNS studies using body-fitted meshes.

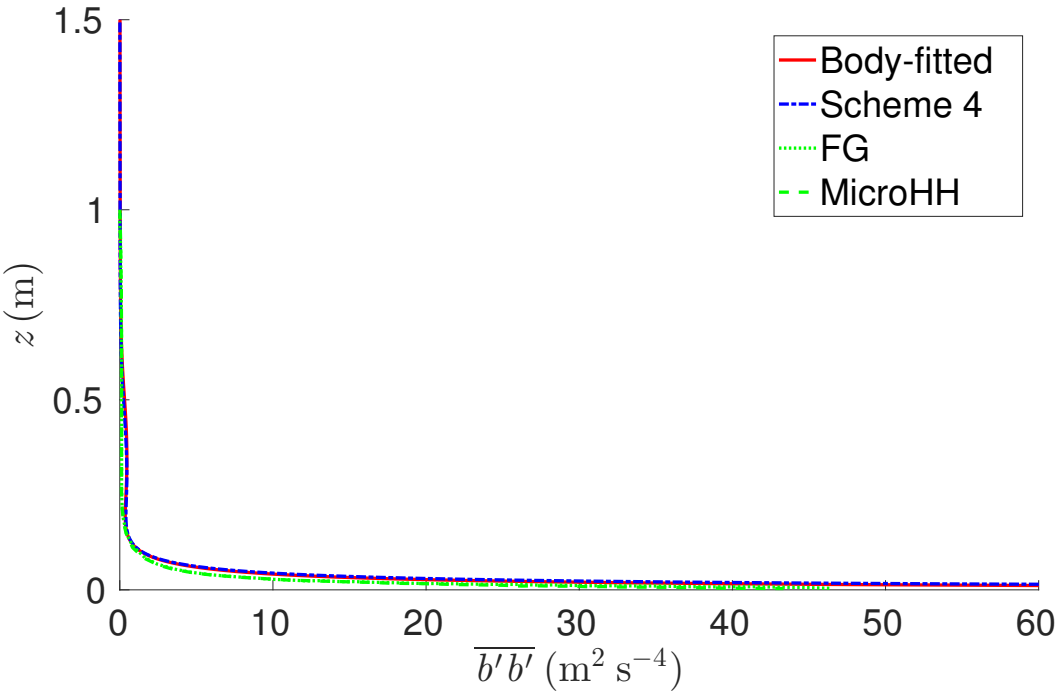


Figure 5.14: Buoyancy variance in a turbulent katabatic flow. Comparison of IB reconstruction Scheme 4 with three different DNS studies using body-fitted meshes.

CHAPTER 6

Simulations of Turbulent Winds over Complex Terrain Using an Immersed Boundary Method

6.1 Abstract

Technological advances in computing power enables us to simulate winds over complex terrain with finer spatio-temporal resolutions than ever before. As such, large-eddy simulation and immersed boundary techniques are finding applications in micro-scale modeling of complex terrain winds. In this study we investigate the Askervein Hill field study and the Bolund Hill study—both the field and wind tunnel cases—to extend an immersed boundary formulation to simulate neutrally-stratified turbulent winds over complex terrain. We reconstruct both the velocity and the eddy-viscosity fields in the terrain normal direction to produce the turbulent stresses as would be expected from a surface parameterization scheme based on the Monin-Obukhov similarity theory. We rely on the logarithmic profile for the wind field to formulate the eddy-viscosity in the vicinity of the surface and reconstruct the velocity field. We found that it is essential to be consistent in the underlying assumptions for the velocity reconstruction and the eddy-viscosity formula to produce good results. Our approach works well for attached turbulent wind fields and shows slight disagreement in separated regions, which we attribute to our reliance on the Monin-Obukhov similarity theory over the entire terrain. Further research is needed to improve the parameterization of turbulent stresses in separated wind regions, to account for varying aerodynamic roughness lengths, and incorporate the stability effects into the immersed boundary formulation.

6.2 Introduction

Modern computing hardware with multi- and many-core architectures has enabled researchers to deploy fine resolutions in studying the physics of atmospheric flow. The large-eddy sim-

ulation (LES) technique has become an obvious beneficiary of increased computing power. However, there are thorny issues to be resolved when the computational mesh resolution approaches on the order of meters. Chief among these is the resolution of terrain details. A complex terrain may be composed of many convex and concave geometric features with steep changes in elevation. If present, these features are a challenge for numerical methods that adopt terrain-following coordinates due to poor mesh quality. When meshing a complex terrain three options are available: an unstructured mesh, a terrain-following mesh, or a fixed Cartesian mesh. Immersing a geometry in a fixed Cartesian mesh and imposing the boundary conditions on grid points that do not coincide with the solid surface is referred to as immersed boundary (IB) methods. IB methods have desirable features such as a good mesh quality, semi-automation of the meshing step and a better fit to the memory layout of emerging computational accelerators. However, the technical challenge is to impose boundary conditions and surface parameterization accurately on non-coincident mesh points.

The original IB method was developed to simulate blood flow through a heart valve (Peskin, 1977). Since then researchers have proposed numerous versions to tackle issues related to the original method or to extend its domain of application. Majority of these methods have been tested in fluids engineering domain at low to moderate Reynolds numbers (Iaccarino and Verzicco, 2003; Mittal and Iaccarino, 2005; Sotiropoulos and Yang, 2014). Sharp-interface IB methods were developed to address some of the numerical stability issues related to the original IB method. One class of sharp-interface methods is the cut-cell method (Ye *et al.*, 1999). In this approach control volumes around the embedded geometry are created by reshaping the cells intersected by the boundary such that faces of the control volumes conform to the boundary exactly. A second class is referred to as immersed-interface methods, which incorporate the jump conditions of the discrete delta function directly into a finite difference stencil without relying on a continuous approximation of a discrete delta function (Leveque and Li, 1994; Lee and LeVeque, 2003). A third class is the direct forcing

approach (Mohd-Yusof, 1997; Verzicco *et al.*, 2000; Fadlun *et al.*, 2000; Tseng and Ferziger, 2003), which directly reconstructs flow variables in boundary-neighboring grid cells via interpolation or distribution functions between the known conditions at the boundary and the surrounding cells. This class of IB methods is also referred to as hybrid Cartesian/immersed boundary approaches when applied as a sharp-interface method Gilmanov and Sotiropoulos (2005) and has the distinguishing aspect that the forcing term in the governing equations does not need explicit calculation.

Most IB methods have been investigated for flow over aerodynamically smooth surfaces. However, IB methods extended to simulate atmospheric flows are much more involved as the boundary conditions take the form of surface stresses to parameterize surface roughness effects and compensate for the lack of resolution relative to the length scales of turbulent flow near the surface. Additionally, parameterization of heat and moisture fluxes are needed to better represent atmospheric flows. Therefore, a straightforward implementation of the IB method with no-slip boundary conditions in atmospheric flows can only produce visually appealing results.

Studies that have accounted for surface roughness over topographies include studies over a single (Tseng and Ferziger, 2003) or double (Bhaganagar *et al.*, 2004) undulations of a sinusoidal wave. Chester *et al.* (2007) developed an IB approach for spectral differentiation where the surface shear stress is determined from a rough-wall log-law using tangential velocity near the IB, then extrapolated and successively smoothed into the interior of the IB which results in a slight smearing of variables near the surface. Diebold *et al.* (2013) applied the approach of Chester *et al.* (2007) to the Bolund Hill complex terrain benchmark (Berg *et al.*, 2011). Anderson (2013) generalized the approach of Anderson and Meneveau (2010) to create a forcing term that creates momentum depletion for flow impinging upon the surface based upon the gradient of the distance field. Tests were performed over many surfaces such as prisms, mounds, sinusoids, fractal-like surfaces and synthetic cities. While this approach has unresolved roughness elements, this forcing term does not explicitly account for mass

conservation near the surface. Yuan and Piomelli (2014a) and Yuan and Piomelli (2014b) used a volume-of-fluid IB method where the velocities and eddy viscosities are reduced by an amount proportional to the volume of fluid contained within the grid cell intersected by the immersed boundary. This approach requires resolution of the roughness elements.

One of the desirable features of IB methods is its non-intrusive implementation in existing simulation codes, which makes it easy to parallelize as well. We made it a goal in our work to retain the ease of computer implementation. To this end, we pursue a direct-forcing IB approach as originally proposed by Mohd-Yusof (1997) and later extended to neutral atmospheric boundary layer over flat terrain by Senocak *et al.* (2004). Unlike other approaches where the stress tensor terms are reconstructed, we restrict ourselves to reconstruct only the primitive variables of the governing equations such as the velocity components and the scalar quantities and thereby retaining the simplicity of computer implementation.

Modeling the ABL at the microscale with the LES technique, even at the order of meters, requires wall-modeling. Therefore, the IB method needs to be formulated in a way to include wall-modeling. Many IB approaches have been proposed for wall-modeled LES (Tessicini *et al.*, 2002; Cristallo and Verzicco, 2006; Ji *et al.*, 2012; Roman *et al.*, 2009; Chang *et al.*, 2014). To this end, the IB method we are investigating for complex terrain will reconstruct the tangential velocity components using logarithmic reconstruction to be consistent with the log-law assumption embedded in the chosen near-surface grid-scale (SGS) model. Because the end goal is to produce the expected turbulent stresses and retain computational ease, we also reconstruct the eddy viscosity to avoid the calculation of velocity gradients at the immersed surface. We investigate the Askervein Hill and Bolund Hill cases to validate our approach. Because these are turbulent flow problems with separated flows region, our results are influenced by the underlying assumption of attached flow in the chosen subgrid-scale model. Some of the discrepancies cannot be directly attributed to the IB method at hand.

In what follows, we present the details of our IB method and simulate both the Askervein Hill and Bolund Hill cases. We compare against available field data, but we also simulate

the wind tunnel experiment for Bolund Hill (Yeow *et al.*, 2015), and compare our results against detailed measurements to better assess vertical velocity profiles across the complex terrain.

6.3 Immersed Boundary Methodology

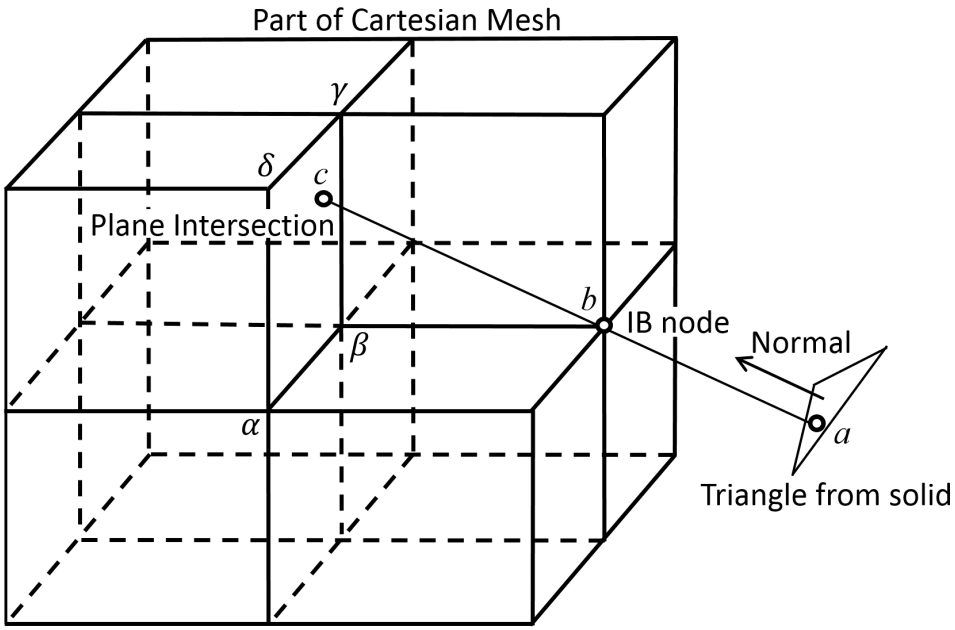


Figure 6.1: Sketch of the IB reconstruction in surface normal direction. The interpolation occurs along line projected from TC through IB until intersection with cell face away from surface is intersected at FI. Cartesian nodes α, β, γ, and δ are interpolated onto FI. TC - triangle centroid; IB - immersed boundary Cartesian node; FI - cell face intersection; α,β,γ,δ - Cartesian nodes that form cell face

The implementation of the IB method using a direct forcing approach, as detailed in Mohd-Yusof (1997); Fadlun *et al.* (2000), is based on solving the following discretized governing momentum equation,

$$\frac{u_i^{t+1} - u_i^t}{\Delta t} = RHS_i^t + f_i^t, \tag{6.1}$$

for f^t such that $u_i^{t+1} = u_{i,bc}^{t+1}$, where u^t is the velocity at time step t, $u_{i,bc}$ is the velocity boundary condition known at all time steps. RHS encompasses the convective, viscous and

pressure gradient terms, and f^t is a body-force term, which can be written as follows after rearranging the terms and enforcing that the forcing term produces the desired velocity at the boundary (i.e. $u_i^{t+1} = u_{bc}^{t+1}$)

$$f_i^t = -RHS^t + \frac{u_{bc}^{t+1} - u_i^t}{\Delta t}, \tag{6.2}$$

where the boundary condition value u_{bc}^{t+1} at the immersed surface is known for every time step. However, with Cartesian grids, u_{bc} does not necessarily coincide with the Cartesian grid points. Therefore, an interpolation scheme is required to reconstruct the velocities at the near-surface Cartesian grid points with respect to the boundary condition information at the surface. An example of reconstruction is sketched in Fig. 6.1. Here, a line is projected from the triangular element shown—used in representing the terrain—through the IB Cartesian node until the line intersects a cell face away from the solid surface. The Cartesian nodes forming the cell face are interpolated onto the face intersection. The velocity is then reconstructed via an interpolation scheme at the IB node by imposing the boundary condition at the surface. Note that by reconstructing the velocity field at the IB node, the forcing term f_i^t is implicitly accounted for in the governing equations.

When the flow is laminar, or the mesh resolves the viscous layer of a turbulent flow, a linear reconstruction scheme is adequate as the velocity varies linearly. However, in LES of atmospheric flows, the mesh is underresolved relative to turbulent length scales and a surface parameterization scheme, also known as a wall-model, is required. Therefore, a linear reconstruction of tangential velocity is no longer applicable for LES of atmospheric flows with a surface parameterization scheme.

Although the applicability of Monin-Obukhov similarity theory for complex terrain or the logarithmic law of the wall for separated flows are questionable, they are commonly used to formulate surface parameterization schemes or wall-models for turbulent stresses because of a lack of comprehensive theory. In this study, our goal is not to develop a new wall-model or

surface parameterization scheme, but rather develop an IB method that embeds an existing wall-model in its formulation. To this end, we adopt a reconstruction scheme upholding the logarithmic law of the wall to reconstruct the velocity field. Specifically, we use the log-law IB reconstruction developed for rough flat terrain in Senocak *et al.* (2004), who proposed a logarithmic reconstruction scheme for velocity tangent to the terrain surface and a linear reconstruction scheme for velocity normal to the surface to explicitly enforce the impermeability condition. The underlying assumptions in the log-law reconstruction scheme are equilibrium between turbulence production and destruction and a constant friction velocity in the vicinity of the surface. To demonstrate how the reconstruction is developed, we start with the rough log-law,

$$U = \frac{u_*}{\kappa} \ln\left(\frac{h + z_0}{z_0}\right), \tag{6.3}$$

where u_* is the friction velocity, κ is the von Kármán constant, where h is the height normal to the surface and z_0 is the aerodynamic roughness length. If the friction velocity is constant, the similarity between two heights in the logarithmic region allows the following relationship to be obtained by dividing the log-law at surface-normal heights, $h = h_{IB}$ and $h = h_{FI}$ (notation from Fig. 6.1 where location TC is $h = 0$),

$$\frac{U_{IB}}{U_{FI}} = \frac{u_*/\kappa \ln\left(\frac{h_{IB}+z_0}{z_0}\right)}{u_*/\kappa \ln\left(\frac{h_{FI}+z_0}{z_0}\right)}. \tag{6.4}$$

Then the logarithmic reconstruction scheme for the tangential velocity is

$$U_{IB} = U_{FI} \frac{\ln\left(\frac{h_{IB}+z_0}{z_0}\right)}{ln\left(\frac{h_{FI}+z_0}{z_0}\right)}. \tag{6.5}$$

where U_{FI} is obtained via bilinear interpolation among nodes α, β, γ, and δ in Fig. 6.1. The normal component is reconstructed at node IB with linear interpolation between location

FI and TC in Fig. 6.1,

$$U_{IB} = \frac{h_{IB}}{h_{FI}} U_{FI} + U_{TC}, \tag{6.6}$$

where U_{TC} is the no-slip condition in this study, i.e. $U_{TC} = 0$, and U_{FI} is the same value as in Eq. 6.5. Once the surface-normal and surface-parallel projections have been individually reconstructed, the projections are then formed into a vector with the same direction as the velocity vector at the face intersection. The underlying assumption here is that the direction of the flow remains constant along the surface normal in the close proximity of the surface. A similar procedure can also be found in Roman *et al.* (2009) where the velocity at an IB node is broken into tangential and normal components and reconstructed using different schemes to simulate a turbulent channel flow. In their study, smooth-wall logarithmic reconstruction was used for the tangential component and a quadratic polynomial was used to reconstruct the normal component.

As we are reconstructing the velocity near the surface and setting the velocity field to zero inside the terrain, one can expect incorrect velocity gradients calculated on the faces of an IB Cartesian grid cell that then go into the calculation of the eddy viscosity. The reconstruction of the eddy viscosity in IB simulation of turbulent flows has not received much attention. As shown later in §6.4, we adopt an eddy-viscosity-based SGS model which relies on the existence of a logarithmic law for the velocity field. The intent here is to be consistent with the logarithmic reconstruction for the velocity field. If the goal is to impose the correct turbulent stresses with the wall model, the incorrect velocity gradients will inject errors into the eddy-viscosity model and produce incorrect results. To avoid this problem, we use the same linear reconstruction scheme given in Eq. 6.6 on the near-surface eddy viscosity. We justify the use of linear reconstruction in §6.4.

For any IB method, it is essential to use accurate geometric information related to the immersion of complex boundaries within a Cartesian domain. For this study, we use the

geometric pre-processing algorithms detailed in Senocak *et al.* (2015), which describes a procedure to immerse a complex geometry with convex and concave regions inside a three-dimensional Cartesian domain. The preprocessor also computes the distance field from a complex terrain using an Eikonal equation. A distance field is an integral part of the grid-scale model used in this study. We refer the reader to Senocak *et al.* (2015) for details, and stress the importance of correct geometric information to implement an IB method for flow computations.

6.4 Numerical Formulation

We solve the filtered form of the incompressible Navier-Stokes equations

$$\frac{\partial \overline{u}_j}{\partial x_j} = 0, \tag{6.7}$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u}_i \overline{u}_j\right) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\nu \overline{S}_{ij} - \tau_{ij}\right), \tag{6.8}$$

where the $\overline{S}_{ij}$ is the strain rate tensor and τ_{ij} are the filtered Reynolds stresses which are modeled with an eddy-viscosity assumption as follows

$$\tau_{ij} = \nu_t \overline{S}_{ij}, \tag{6.9}$$

$$\nu_t = l_{mix}^2 |\overline{S}|. \tag{6.10}$$

We use a second-order Adams-Bashforth scheme for time advancement and discretize the spatial derivatives with a second-order central difference scheme on a directionally-uniform Cartesian grid. The three dimensional incompressible flow solver, for which density is set to unity, has been parallelized for fast computing on a cluster of graphics processing units (GPU) (Thibault and Senocak, 2012; Jacobsen and Senocak, 2013; DeLeon *et al.*, 2013). The Poisson equation for pressure is solved with an amalgamated parallel geometric multigrid technique (Jacobsen and Senocak, 2011).

Case	z_0 (m)	Periodic Shift (m)	Target Mass Flow Rate (kg s^{-1})
Askervein FS	0.03	1000	9.01×10^7
Bolund FS	6.0×10^{-4}	150	1.26×10^6
Bolund WT	1.02×10^{-5}	0.4	38.39

Table 6.1: Summary of simulation parameters used for the three test cases. FS, Field Study; WT, Wind Tunnel.

We use the near-surface model described in Senocak *et al.* (2007), which is a hybrid RANS-LES that blends the mixing length of the LES -grid scale (SGS) eddy viscosity model with the Prandtl mixing length (Prandtl, 1925), such that l_{mix} in Eq. 6.10 becomes

$$l_{mix} = (1 - \exp(-h/h_B))\, C_S \Delta + \exp(-h/h_B)\, \kappa h, \tag{6.11}$$

Note that we blend the length scales directly instead of the squared quantities that was originally proposed in Senocak *et al.* (2007). The Smagorinsky coefficient C_S is computed using the localized Lagrangian dynamic Smagorinsky SGS model (Meneveau *et al.*, 1996). In Eq. 6.11, h is the surface-normal distance from the terrain, h_B represents the RANS-LES blending height, and Δ is the LES filter width defined from the cube root of a grid cell volume as $\sqrt[3]{\Delta x \Delta y \Delta z}$. At the IB Cartesian nodes close to the surface, the hybrid RANS-LES scheme will be exclusively RANS meaning the SGS model will be described by the Prandtl mixing length model, which can be written as,

$$\nu_t = (\kappa h)^2 \, |\overline{S}| = \kappa h u_*. \tag{6.12}$$

The far right-hand-side of Eq. 6.12 shows a linear dependence on h indicating we can use a linear IB reconstruction.

Case	Mesh 1	Mesh 2	Mesh 3
Askervein FS	321 × 257 × 129	513 × 385 × 193	641 × 513 × 257
Bolund FS	385 × 193 × 129	513 × 257 × 193	769 × 385 × 257
Bolund WT	N/A	N/A	1025 × 193 × 513

Table 6.2: Summary of simulation mesh sizes used for the three test cases. FS, Field Study; WT, Wind Tunnel.

6.5 Askervein Hill

We first apply the IB method described in section 6.3 to the Askervein Hill case, which was the focal point of a field measurement campaign in the 1980s (Taylor and Teunissen, 1987; Mickle *et al.*, 1988; Salmon *et al.*, 1988; Walmsley and Taylor, 1996). The experiment has a very high density of measurements making it one of the most widely used isolated complex terrain benchmarks. The topography, as shown in Fig. 6.2, can be described as gently sloping that is suitable to be represented by a terrain-following grid. Lopes *et al.* (2007) performed a wall-modeled LES (i.e., hybrid RANS-LES) using a Lagrangian dynamic Smagorinsky model coupled with the rough surface model of Marusic *et al.* (2001) and the near-surface derivative correction of Porté-Agel *et al.* (2000). Chow and Street (2009) used the Advanced Regional Prediction System (ARPS) (Xue *et al.*, 2003) to evaluate LES SGS models with a specified surface stress boundary condition (Raithby *et al.*, 1987). Golaz *et al.* (2009) used the Coupled Ocean/Atmosphere Mesoscale Prediction System (COAMPS ®) with LES (Golaz *et al.*, 2005) using a rough-surface log-law boundary condition. The aforementioned studies perform LES on terrain-fitted meshes. In general, the windward side of the hill is captured well with little sensitivity to mesh resolution. The leeward side where intermittent flow separation occurs exhibits significant sensitivity to SGS model parameters and mesh resolution, which can be attributed to the use of equilibrium assumptions in the models when intermittent flow separation is not in equilibrium (Lopes *et al.*, 2007).

In our simulations, we set the aerodynamic roughness length, z_0, to 0.03 m (Taylor and Teunissen, 1987). The domain size was 8218.0 m × 5800.0 m × 1000.0 m in the stream-

wise, horizontal spanwise and vertical directions with a starting grid resolution of 641 $\times$ 513 $\times$ 257 in the same respective directions. The top boundary condition was a zero-shear-stress wall, the spanwise lateral boundary conditions were periodic, and all the streamwise boundary conditions were the shifted periodic boundary conditions of Munters *et al.* (2016a). The shifted periodic boundary conditions are designed to avoid the spanwise locking of flow structures by shifting the outflow condition a set distance in the spanwise direction during the periodic recycling. The end result is shorter domains than are required with standard periodic conditions, because the flow structures near the boundary become more decorrelated. For Askervein, we use a shift of 1000.0 m for the shifted periodic boundary conditions and drive the flow by dynamically calculating the driving mean pressure gradient to keep a constant target mass flow rate of approximately 9.01×10^7kg s^{-1} using the formulation of Benocci and Pinelli (1990). We use the terrain map that does not include neighboring hills (Walmsley and Taylor, 1996) and we compare results to the case of MF03-D taken between 1400-1700 BST (British Summer Time = UTC + 1h) on 3 October 1983 (Taylor and Teunissen, 1985) that is commonly used for the close to neutral atmospheric conditions. For this field study, and the rest of the results presented in this study, we use the fractional speed-up ratio (Jackson and Hunt, 1975) to compare with the experimental benchmarks. Fractional speed-up ratio is the horizontal velocity speed up relative to an undisturbed mean velocity profile,

$$S = \frac{U(h_{agl}) - U_{ref}(z)}{U_{ref}(z)}, \tag{6.13}$$

where h_{agl} and z are both vertical heights above ground level of the terrain except z contains the global elevation datum.

We first probe the effects of the turbulence model before investigating the impact of different mesh resolutions on the performance of the IB method. Figures 6.3 and 6.4 show fractional speed-up ratios of the IB method compared to the Askervein Hill field study

measurements. We tested RANS-LES blending heights of 15.0 m, 25.0 m 35.0 m , and 45.0 m. Overall, little sensitivity occurs on the windward side of the hill for both A and AA lines, indicating the blending height does not have much effect on the fractional speed-up ratio of the wind velocity. Good agreement is achieved with the field data (Taylor and Teunissen, 1985) on the windward side on both A and AA lines. This is consistent with attached flow assumptions that are integral to the formulation of the current IB method. We observed some disagreement with measured data on the leeward side at about 200 m after the hill centre point (CP in figure 6.2), where intermittent flow separation is present. The disagreement within separated flow regions is not unexpected as the current method is based on attached flow assumptions. The A line shows a large dependency on the RANS-LES blending height past 200 m past the hilltop (HT in figure 6.2). As seen in Fig. 6.4, increasing the blending height increases the fractional speed-up ratio on the leeward side of the hill. By analyzing the results from these two figures, we decided to use $h_B = 35.0$ m in our simulations.

The difference in sensitivities observed on the leeward side between the A and AA lines can be explained by the A line being located behind the tallest and steepest part of the hill while the leeward locations along the AA line are behind a much shorter part of the hill. Regarding the direction of the mean incoming flow, the leeward locations of line A are in a larger shadow of hill than line AA. Consequently, the flow separation is expected to be larger at the leeward locations along line A. Figure 6.5 is a visualization of the wake region behind the hill. Clearly, line A passes through a large region of separated flow while line AA passes through a much smaller region. We note that the attached flow assumptions in the turbulence model and the log-law reconstruction do not prevent us from capturing separation, but the accuracy is affected by such assumptions.

Next, we proceed to discuss the sensitivity of the IB method mesh resolution. We use three grids given in Tab. 6.2. As suggested in Senocak *et al.* (2007), we keep the relation $h_B(2\Delta)^{-1} > 1$ to ensure the LES technique is not applied in regions of insufficient resolution. $h_B = 35.0$ m satisfies this relation on all three meshes. Figures 6.7 and 6.6 are the fractional

speed-up ratios at 10 m above ground level along the A and AA lines in the mesh resolution study. The IB method mostly shows good agreement along the AA line in Fig. 6.6 with minor sensitivity to mesh resolution for most of the terrain.

Similar to the situation with line AA, we also observe good agreement on the windward side of the hill along line A in Fig. 6.7, more so with the two finer meshes, Mesh 2 and Mesh 3. The coarsest mesh, Mesh 1, experiences a noticeable reduction in speed 200 m before HT. Careful examination of the hill elevation map (Walmsley and Taylor, 1996) reveals that the STL representation has a slight convex feature on the windward face. Mesh 1 has horizontal resolution of approximately 25 m, which is too coarse to correctly capture this convexity, leading to the observed dip in fractional speed-up ratio. Mesh sensitivity is low at the hill top and approximately 200 m downstream of HT. However, mesh sensitivity becomes more pronounced further downstream, and more so than what we observe along line AA. This is consistent with the sensitivities observed in the blending height study. A careful examination of the terrain profiles (Walmsley and Taylor, 1996) reveal small convexities approximately 200 m after HT along line A. A much finer adaptive resolution is needed to capture this terrain feature correctly. The results from figures 6.7 and 6.6 indicate that the immersed boundary method performs well on various meshes, provided that terrain features are adequately resolved.

We show the wind speed profile at the Askervein Hill reference site (RS) in Fig. 6.8 for all three meshes. All three meshes produced nearly-identical mean profiles that agree better with the field measurements at RS than the recommended log-law profile with $z_0 = 0.03$ m and $u_* = 0.654$ m s^{-1}. These results confirm that the shifted periodic boundary condition driven by a constant mass flow rate forcing is able to produce the expected wind profile for the lowest 100 m. The lack of agreement roughly above 100 m is not unexpected as we do not account for the vertical structure of the atmospheric boundary in our formulation.

The fractional speed-up ratio at hill top (HT) is given in Fig. 6.9. The profiles at HT show little difference among the three meshes. This is consistent with the same observation

for the wind profiles at the reference site. The only significant difference is below 20 m. In this particular case, because the complex terrain is immersed and of fixed position in the domain, the IB node ended up being closer to the terrain on the coarsest mesh. At sequent refinements the IB node moved away from the surface for Mesh 2 and Mesh 3, producing a counter-intuitive result of better prediction on a coarser mesh. Attention must be given to details such as this if prediction near the surface is of importance.

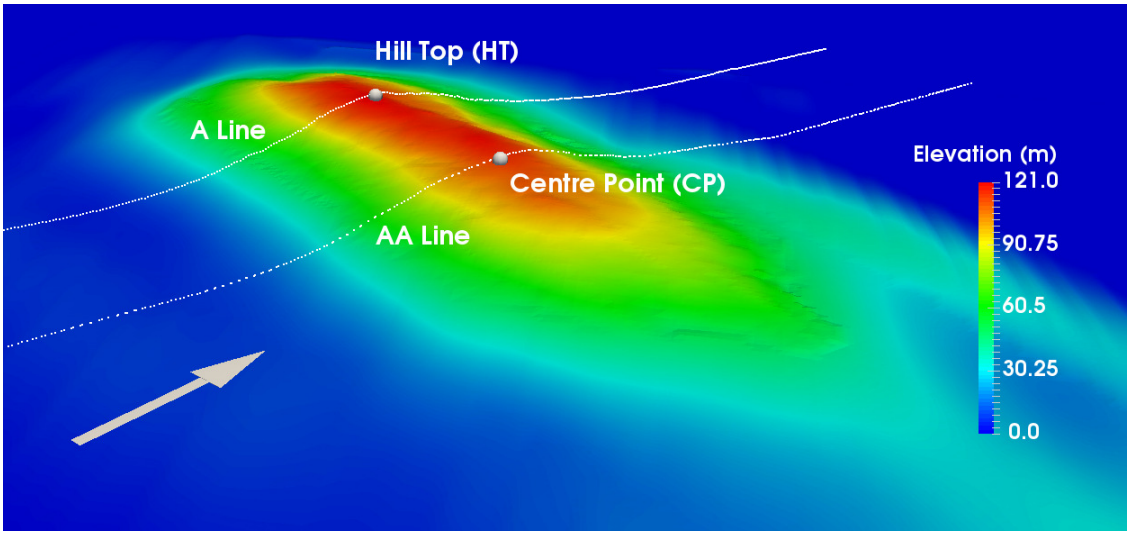


Figure 6.2: Askervein Hill elevation map. The lines designated A and AA are the sampling locations for comparison to the actual Askervein Hill field experiment. The large spheres represent locations designated hill top (HT) and centre point (CP) during the measurement campaign. The arrow indicates mean flow direction as well as positive x-direction.

6.6 Bolund Hill Full Scale

Bolund Hill is a coastal approximately 12-meter-high hill located in Denmark. It was the focus of a field study (Berg *et al.*, 2011) followed by a blind evaluation of computer models (Bechmann *et al.*, 2011). The hill is completely surrounded by water at certain times of the year and features a steep escarpment which makes it a challenging case for wind solvers as seen by the large span of results in the blind evaluation study. IB methods have previously been applied to Bolund Hill in RANS (Jafari *et al.*, 2012) and LES simulations (Diebold *et al.*, 2013). Jafari *et al.* (2012) used the approach of Fadlun *et al.* (2000) to enforce the

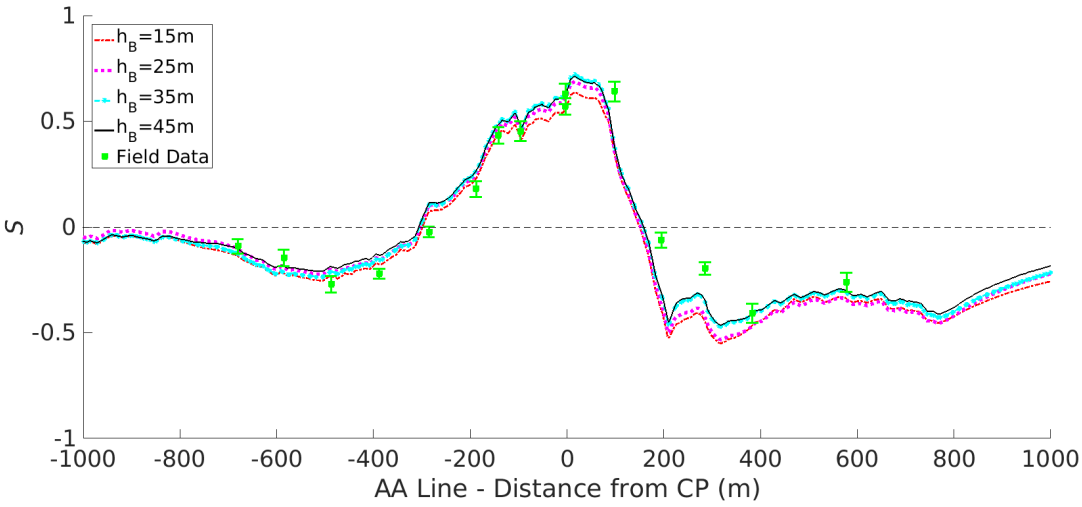


Figure 6.3: Fractional speedup sensitivity to RANS-LES transition heights along the AA Line at 10 m above ground level. The Askervein Hill field data is from Walmsley and Taylor (1996).

no-slip condition at the surface while using wall functions to reduce computational expense in the vicinity of the surface. Diebold *et al.* (2013) used the approach of Chester *et al.* (2007) and imposed surface stresses derived from the logarithmic law of a rough wall. The study found sensitivity to using only one roughness length rather than two roughness lengths for land and water. Conan *et al.* (2016) performed a comparison between a wind tunnel, an LES with wall functions Vuorinen *et al.* (2015) and the field experiment of Bolund Hill. Both Diebold *et al.* (2013) and Conan *et al.* (2016) found sources of error in the turbulent flow close to the ground that did not significantly affect the flow away from the surface. Along with the wind tunnel experiment of Conan *et al.* (2016), Bolund Hill was also the ject of a wind tunnel experiment by Yeow *et al.* (2015), which we use to assess our simulations in addition to the field data.

We use a computational domain of 1533.0 m $\times$ 765.0 m $\times$ 127.0 m in the streamwise, horizontal spanwise and vertical directions to simulate actual-scale Bolund Hill. We tested three grid resolutions given in Tab. 6.2. The finest mesh used gave resolution of 2.0 m $\times$ 2.0 m $\times$ 0.5 m, in the x, y, z directions, respectively. The RANS-LES blending height, h_B, was set to 5.3 m to maintain $h_B(2\Delta)^{-1} > 1$ with all three meshes as recommended in Senocak *et al.*

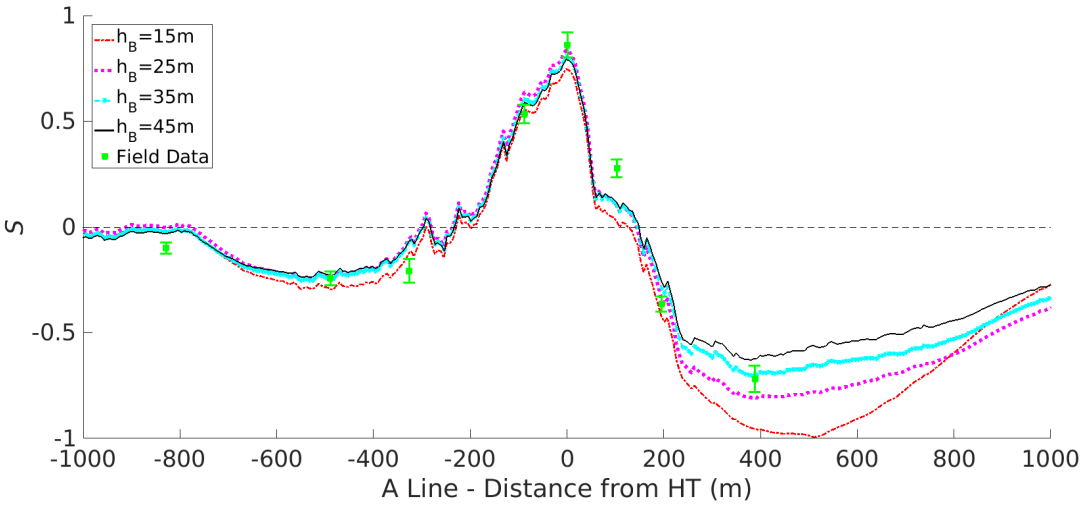


Figure 6.4: Fractional speedup sensitivity to RANS-LES transition heights along the A Line at 10 m above ground level. The Askervein Hill field data is from Walmsley and Taylor (1996).

(2007). The boundary conditions are identical to the Askervein Hill setup except that the shift for the periodic boundary conditions (Munters *et al.*, 2016a) was set to 150 m and the incoming mass flow rate was approximately 1.26×10^6 kg s^{-1}. The aerodynamic roughness length, z_0, was set to 0.0006 m. We chose to compare with Line B for the 270-degree wind direction as described in Berg *et al.* (2011) and shown in Fig. 6.10.

Figure 6.11 shows streamlines of instantaneous velocity over the hill. By using the mean horizontal wind speed at the location for the reference mast designated M0 in Berg *et al.* (2011), we calculated the fractional speedup ratio along line B, consisting of masts M7, M6, M3, and M8 in the field study, for each mesh. Figure 6.12 compares these results to the field data from Berg *et al.* (2011). Overall, the sensitivity between the different mesh resolutions is minor except for near the surface at M6. The regions of attached flow where M7 and M3 reside show very good agreement among all three meshes. As we discuss later, the IB results in Fig. 6.13 show a better agreement with the wind and water tunnel data than they do with field study measurements. This is not unexpected, because it is much easier to represent the conditions in a laboratory experiment than in a field study. Also, our single roughness length is more similar to a laboratory setting than the field experiment.

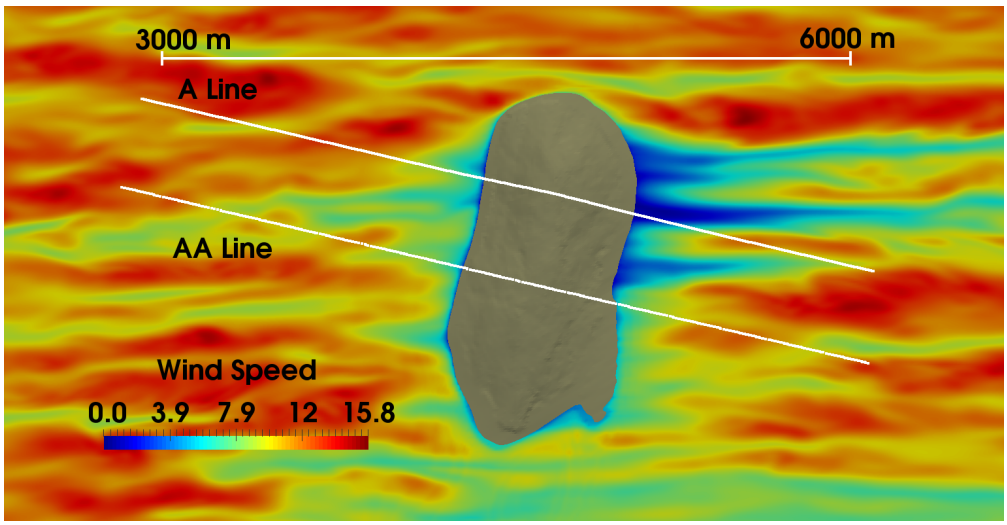


Figure 6.5: Visualization of the flow around Askervein Hill. Horizontal wind speed given in units of m s^{-1}. Visualization is aligned with mean flow direction, left to right and is only a set of the entire domain, with a 3000 m ruler given for reference.

The mast M6 lies directly behind the escarpment making it the most uncertain to capture, as evidenced by the spread in benchmark data and disagreement between field and laboratory experiments as shown in Fig. 6.13. As observed in the mesh sensitivity results for the Askervein Hill case, the placement of the surface relative to the Cartesian mesh can influence results obtained from the first few grid points off of the surface. At some locations, the first off-surface grid points show a slight increase in fractional speed-up ratio. At M6 for example, the coarser resolutions of Mesh 1 and Mesh 2 have grid points closer to the surface than Mesh 3 and produce different behavior than is expected. M6 is in an area of flow separation and the attached flow assumptions affect the accuracy in this region, which has also been observed on terrain-fitted approaches (Lopes *et al.*, 2007).

6.7 Bolund Hill Wind Tunnel

The wind tunnel experiment of Yeow *et al.* (2015) recreated the Bolund Hill on a 1:115 scale. An elevation map is shown in Fig. 6.10. The reported Reynolds number was 4.15×10^5 based on the hill height. We generate a computational domain that is the same size as the wind tunnel experiment at 10.5 m $\times$ 2.2 m $\times$ 2.2 m in the streamwise, horizontal spanwise,

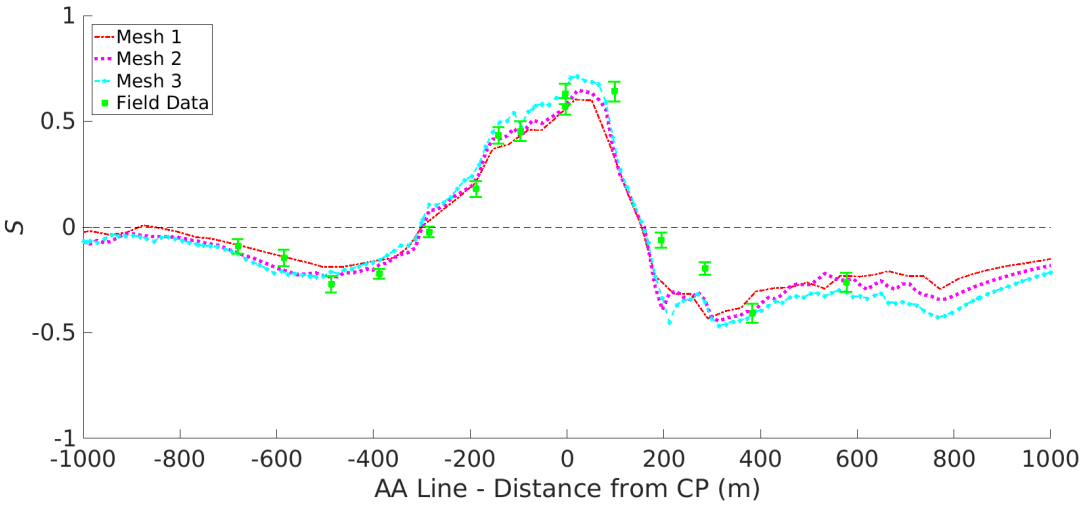


Figure 6.6: Fractional speedup sensitivity to mesh resolution along the AA Line at 10 m above ground level. Mesh 1, 2, and 3 refer to the three meshes in Tab. 6.2. The Askervein Hill field data is from Walmsley and Taylor (1996).

and vertical directions with a grid resolution of 1025 $\times$ 193 $\times$ 513 in the same respective directions. The mesh resolution used to replicate the wind tunnel study corresponds to 1.2 m $\times$ 1.2 m $\times$ 0.5 m when scaled to the actual hill size.

The IB method was applied to the geometry with an aerodynamic surface roughness of $z_0 = 1.02 \times 10^{-5}$ (Yeow *et al.*, 2015). The top wall of the wind tunnel was modelled as a smooth-surface, wall-shear-stress boundary conditions via the Schumann-Grötzbach model Schumann (1975); Grötzbach (1987) with standard values for the smooth-wall log-law with von Kármán constant being 0.41 and log-law intercept constant being 5.2 Pope (2000). We applied the same lateral boundary conditions we did for Askervein Hill but with periodic shift of 0.4 m and target mass flow rate of 38.39 kg s^{-1}. While not shown in this paper, we did test the sensitivity in fractional speedup ratios using wall models for the spanwise boundaries and standard periodic boundary conditions for the streamwise direction. The results did not show significant difference in fractional speedup with the boundary conditions presented in this study, and we chose the shifted periodic boundary conditions for the decorrelation advantage. The RANS-LES blending height, h_B, was set to 0.0217 m, which is equivalent to 2.5 m when scaled to the actual hill size.

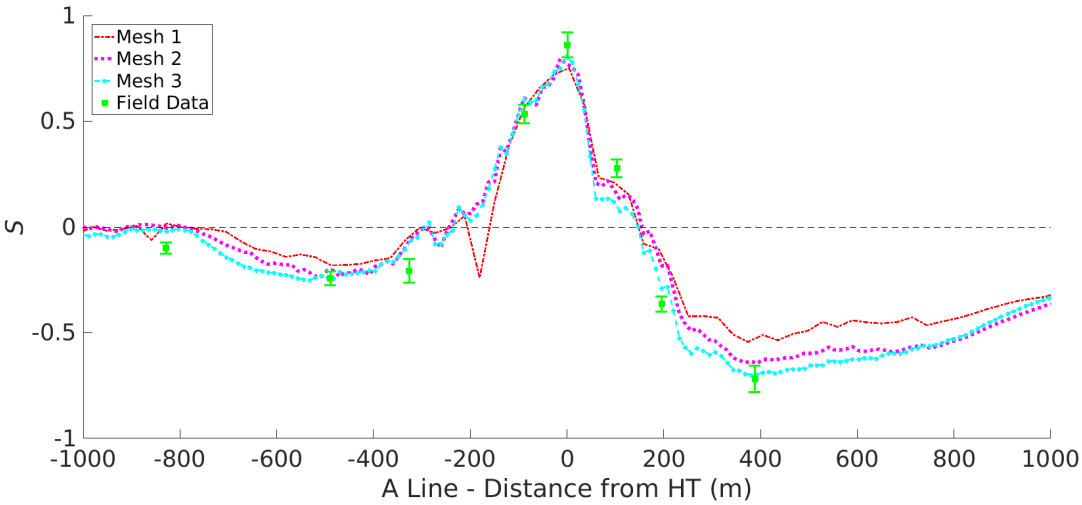


Figure 6.7: Fractional speedup sensitivity to mesh resolution along the A Line at 10 m above ground level. Mesh 1, 2, and 3 refer to the three meshes in Tab. 6.2. The Askervein Hill field data is from Walmsley and Taylor (1996).

Figure 6.13 shows the fractional speed-up ratios compared to the experimental results of Yeow *et al.* (2015), the field measurement campaign Berg *et al.* (2011) and the water tunnel experiment Bechmann *et al.* (2011). The reference location corresponds to the same location used in the study by Yeow *et al.* (2015). The mast locations correspond to those shown in Fig. 6.10. Mast M7 is captured well by the IB method, agreeing well with all three benchmarks. The most challenging mast, M6, is directly behind the escarpment of the hill. The challenge is evident as all three benchmark cases each have different fractional speed-up ratios near the surface at M6. Yeow *et al.* (2015) attributes this to the intermittent recirculation found after the escarpment as well as the ratio of the boundary layer height to the hill height being different than at full scale. The IB method falls between the field study and the wind tunnel near the surface. Good agreement with all three benchmarks is achieved further away from the surface where the benchmark results collapse.

Comparing the IB method results for the wind-tunnel scale and the actual hill scale shows some differences. We also note the differences among laboratory and field experiments, highlighting the challenges to validate complex terrain wind simulations. Our results for the actual hill scale and the laboratory scale show differences as well. One contributing factor is

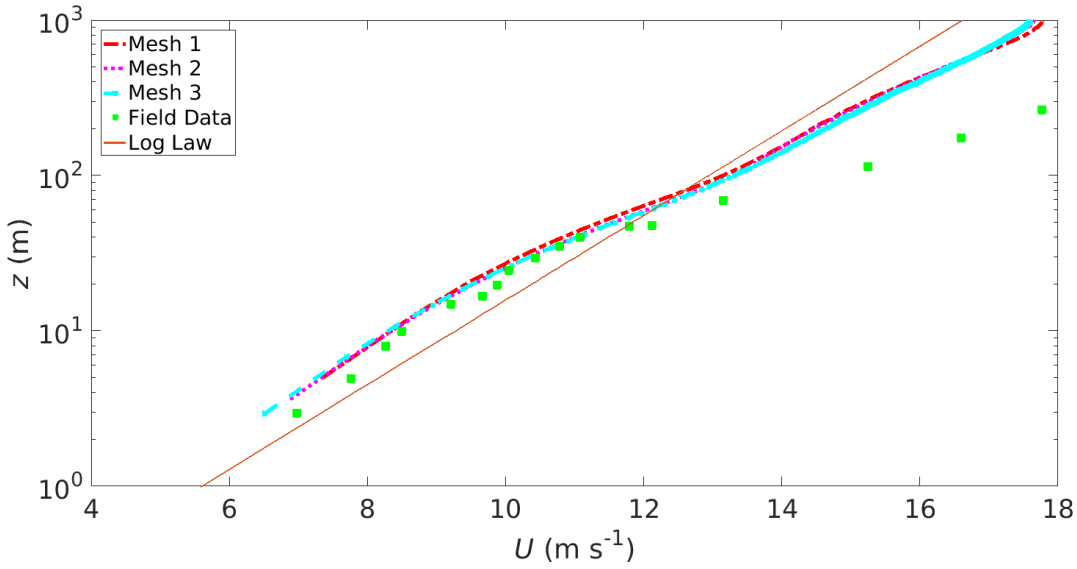


Figure 6.8: Mean velocity at the reference site (RS). Mesh 1, 2, and 3 refer to the three meshes in Tab. 6.2. The Askervein Hill field data is from Walmsley and Taylor (1996). The log-law is plotted with Eq. 6.3 with $z_0 = 0.03$ m and $u_* = 0.654$ m s^{-1}.

the top boundary condition is different between those two simulations (i.e. zero-shear-wall vs. wall-stress boundary condition). Another major factor, perhaps the most influential, is the slightly finer mesh resolution (when scaled to actual size) in our wind tunnel simulation. As was observed in Figs. 6.6 and 6.7, the finer mesh produced better agreement in the wake of the hill, which we again observe in Fig. 6.13 at M8 compared to Fig. 6.12. Reynolds number and incoming flow profile also differ between actual hill scale and laboratory scale simulations.

6.8 Summary and Conclusions

We investigated the use of a direct-forcing immersed boundary method to perform LES of neutrally stratified turbulent winds over complex terrain. In our approach we reconstructed the tangential velocity using a rough-wall logarithmic wind profile. We reconstructed the normal component of the velocity using a linear interpolation to enforce the impermeability condition. We adopted a near-surface eddy-viscosity model based on the logarithmic

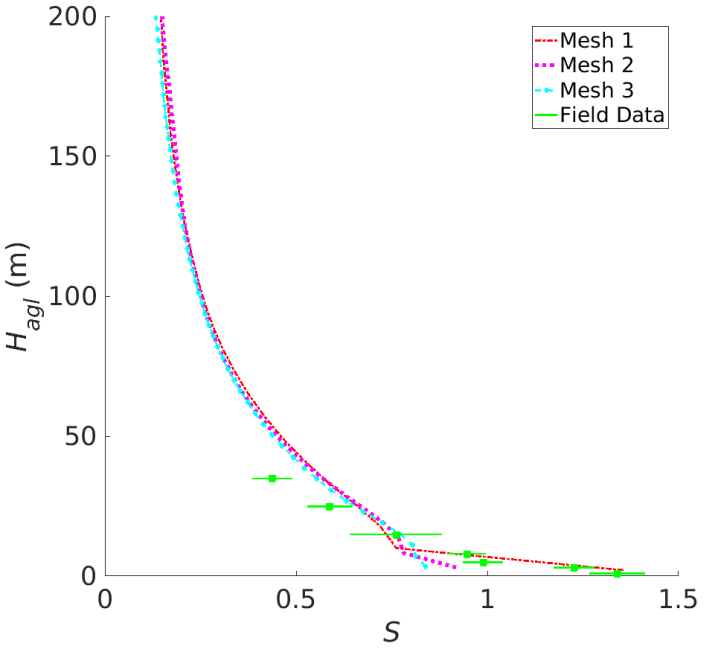


Figure 6.9: Fractional speedup sensitivity to different meshes at the hill top (HT). Mesh 1, 2, and 3 refer to the three meshes in Tab. 6.2. The Askervein Hill field data is from Walmsley and Taylor (1996).

wind profile and used a linear reconstruction on the eddy viscosity to produce the expected turbulent stresses. Consistent adoption of the logarithmic wind assumption in the velocity reconstruction and in the definition of the eddy viscosity was paramount to the success of the simulations. Through this consistent formulation, we were able to retain ease of computer implementation of the direct-forcing IB method.

We simulated wind fields over Askervein Hill and Bolund Hill to validate our approach. The current IB method works well in attached flows when terrain features are adequately resolved. In our study the vertical mesh resolution was about 4.0 m for the Askervein Hill case and 0.5 m for the Bolund Hill case. Results showed minor sensitivity to the RANS-LES blending height in attached flow regions. Fractional speed-up ratios in regions of separated flow showed some sensitivity to both mesh resolution and turbulence model parameters. This is not unexpected as the assumptions behind both the IB method and the near-surface eddy viscosity model are not intended for flow separation.

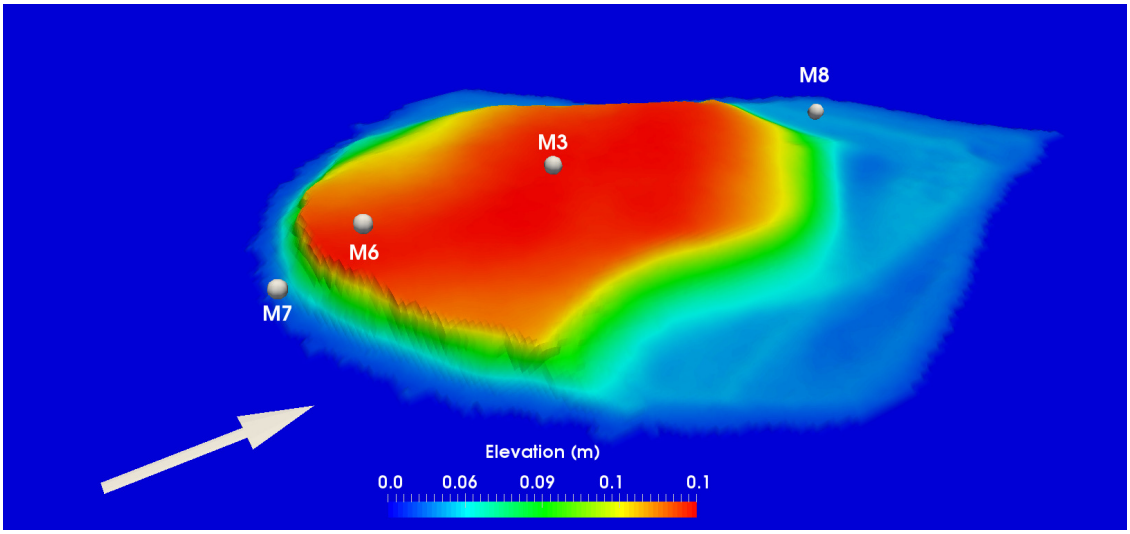


Figure 6.10: Bolund Hill elevation map. The scale is that of the wind tunnel experiment of Yeow *et al.* (2015). The measurement locations, represented by the large spheres, are those used in the field measurement campaign.

The current IB reconstruction scheme and the eddy viscosity model could benefit from theories applicable to separated flow. Similar conclusions about separated flow have been made in other LES studies using terrain-fitted approaches (Lopes *et al.*, 2007). Nevertheless, the current IB methods showed good agreement in fractional speed-up ratios for attached flow regions, while showing some disagreement in regions with flow separation. In future work, we intend to extend the current IB framework to study both stable and unstable stratification over complex terrain.

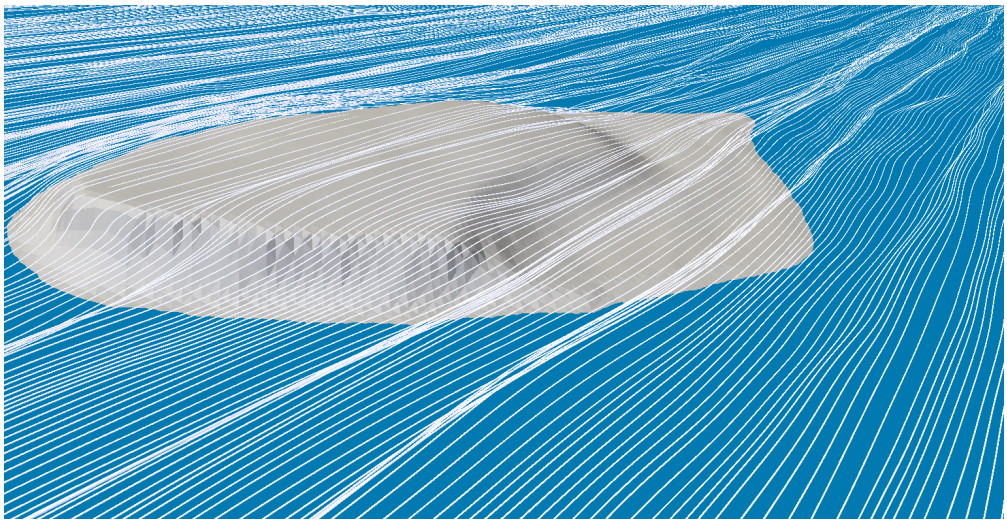


Figure 6.11: Streamlines of instantaneous velocity over Bolund Hill at the field study scale. A triangulated stereolithography (STL) representation of the hill is used for the IB method.

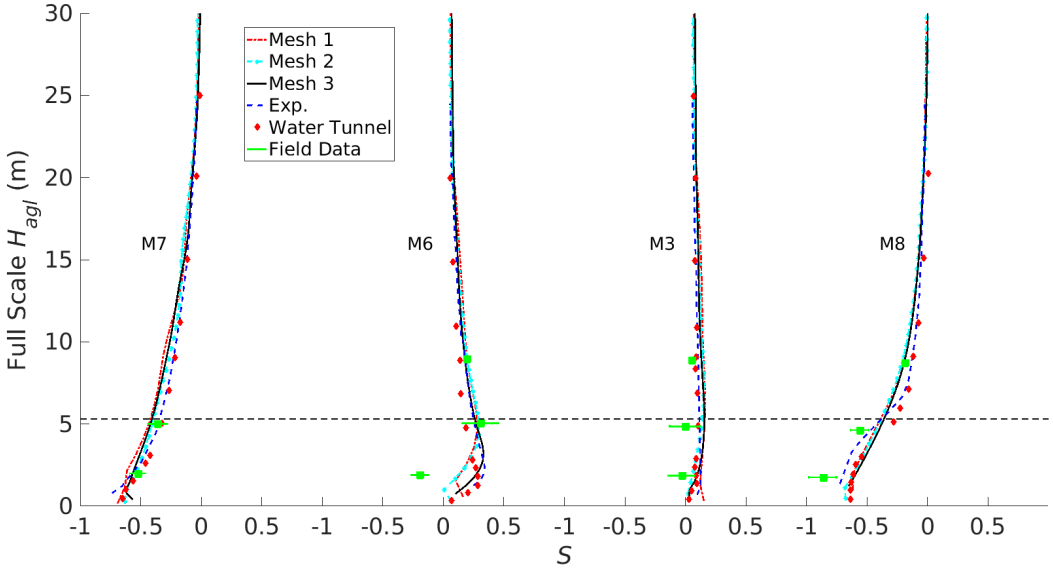


Figure 6.12: Fractional speedup along line B for the 270-degree wind direction in the Bolund Hill field study. Masts M7, M6, M3, and M8 respectively as described in Berg *et al.* (2011). Horizontal dashed line is the RANS-LES blending height. Field data obtained from Berg *et al.* (2011).

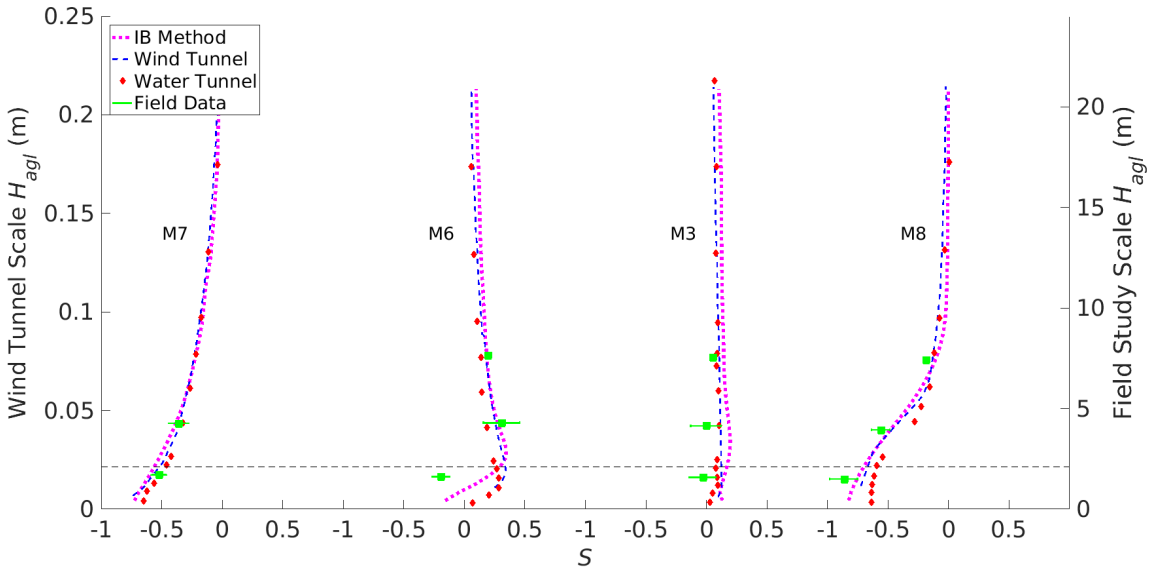


Figure 6.13: Fractional speedup comparison at M7, M6, M3, and M8 for Bolund Hill wind tunnel domain. Wind tunnel data are from the experiment of Yeow *et al.* (2015); water tunnel and field data are from Bechmann *et al.* (2011). Dashed horizontal line is the RANS-LES blending height.

CHAPTER 7

A Turbulent Inflow for Large-eddy Simulation Using Stochastic Buoyancy Perturbations

7.1 Introduction

The essential idea of large-eddy simulation (LES) is to resolve the large scales of turbulence while modeling the small scales, the separation of scales achieved with a low-pass filtering operation. For accurate results, the resolved scales of the LES require the correct turbulent structure and correlations at the boundaries of the domain. The standard periodic boundary conditions can accomplish this once a flow has reached a developed turbulent state. However, standard periodic boundary conditions are not desirable as generalized boundary conditions as periodicity is not applicable to all flows. Inflow and outflow boundary conditions have much more general applicability but offer a challenge when used in LES. Unlike Reynolds-averaged Navier-Stokes (RANS), smooth velocity information at the LES inflow does not work as the flow will need an unreasonable distance to develop into realistic turbulence. Therefore, the inflow condition for LES needs to be able to generate turbulent flow structure within a practical distance efficiently. Such an turbulence inflow generation technique is highly desirable for a broader adoption of LES in engineering applications.

Concerning turbulent boundary layers, perhaps the most basic way to provide accurate turbulent flow to an LES is to allow a flow to transition from a smooth mean profile to a fully-developed turbulent flow. As wall-bounded LES is computationally-expensive, the domain size and mesh resolution requirements would make this basic approach impractical at higher Reynolds numbers. Therefore, a method to shorten the computational domain by providing accurate turbulent information at the inflow is an area of active research (Tabor and Baba-Ahmadi, 2010; Wu, 2017). Turbulent inflow methods fall into two broad categories: precursor methods and synthetic methods (Tabor and Baba-Ahmadi, 2010). The essential idea of precursor methods is to fully develop a boundary layer in an auxiliary simulation

using periodic boundary conditions, then subsequently feed sampled flow field information from the developed auxiliary simulation into the target simulation. The obvious advantage is that the inflow boundary condition will have developed turbulent flow field information but the obvious disadvantage is having extra computational cost (i.e. two domains and data transfer between them).

Synthetic methods seek to avoid the extra computational cost by synthetically producing turbulent flow information or accelerating the onset of turbulence at the inflow of the domain. This is very challenging, more so than imposing small white-noise random perturbations on a mean profile. In a periodic turbulent channel flow, imposing stochastic white-noise on a mean profile as an initial will quickly dissipate as the turbulent energy cascade cannot develop, and the same will happen for a turbulent inflow boundary condition. Initialization of periodic channel flow needs to be such that large flow structures can trip and engage the energy cascade. For inflow conditions, correct Reynolds stresses will engage the energy cascade, but these stress tensors are typically not known *a priori* for general flows. Synthetic methods thus require a development fetch before flow exhibits the correct turbulent flow structure and correlations. The development fetch may add significant cost to a simulation, possibly even more than an auxiliary domain. Research in synthetic methods aim to reduce the length of the fetch.

The challenges of turbulent inflows are evident in reviews of the subject. Keating *et al.* (2004) evaluated several generation methods and found that turbulent information needs to contain scales larger than the integral length scale to achieve the turbulent energy cascade. The recent comprehensive review of Wu (2017) highlights many of the future challenges with turbulent inflow generation, including the need to develop more robust methods without requiring ad-hoc calibrations. Also highlighted is the need for robust methods for atmospheric flows with stable or unstable stratification and three-dimensional mean flow as well as high-Mach-number flows and the related acoustic disturbances caused by inflow techniques.

Lund *et al.* (1998) proposed an improved precursor method for flat-plate boundary layers

where the velocity field is sampled several boundary-layer thicknesses downstream in the precursor simulation. The sampled velocity is rescaled using inner and outer layer scaling laws before being reintroduced at the inlet to create a spatially-developing boundary layer. In order to use similarity laws to perform the rescaling, the flow would need to be in equilibrium. For general flows, this methodology may create a large domain assuming a downstream region if equilibrium exists.

Klein *et al.* (2003) applied a digital filtering synthetic approach to normally-distributed random fluctuations to achieve proper two-point correlations. As two-point correlations are not known in general cases, the filter coefficients were calculated assuming the two-point correlations were Gaussian in shape. The purpose of the filter is to have the larger scales become more prominent in the random fluctuations to have a more natural energy cascade. di Mare *et al.* (2006) further investigated the digital filtering approach by finding exact filter coefficients from known two-point correlations. Better agreement was achieved with the exact coefficients, however the disadvantage still remained of not knowing the Reynolds stress tensor *a priori*.

Jarrin *et al.* (2006) proposed the synthetic eddy method that creates a box of artificial eddies of prescribed shape but random sign and position around the inflow area. These artificial eddies are convected at a constant velocity through the box. The actual velocity field then has fluctuations imposed on it by a Cholesky decomposition of the vortex field produced by the synthetic eddies. The method performed well in channel flows and has also been tested on a variety of flows that transition from upstream Reynolds-averaged Navier-Stokes (RANS) simulations to LES Jarrin *et al.* (2009). However, the synthetic eddy method does not produce divergence-free turbulent fields. Poletto *et al.* (2013) improved upon this shortcoming of the original method.

Pamiès *et al.* (2009) also expanded upon the method of Jarrin *et al.* (2006) by splitting the single stochastic signal in the original synthetic eddy method into multiple modes. The idea was to closer replicate the length and time scales of wall-bounded turbulence with multiple

modes than is possible with a single mode. The modified synthetic eddy method worked well when calibrated to a turbulent flat-plate boundary layer.

Most turbulent inflow methods for LES are designed for stationary mean flow direction. However, growing interest has led to recent proposals for turbulent inflows that accommodate time-varying mean flow directions. Munters *et al.* (2016b) proposed a precursor methodology that rotates the sampling regions of a concurrent precursor simulation (Stevens *et al.*, 2014) according to a known mean flow direction before feeding the flow field information into the target simulation. The method was applied as an inflow condition for an LES of an atmospheric boundary layer flow over an off-shore wind farm, with good agreement in wind velocity compared to other LES simulations of the same wind farm. The approach has the limit that if the mean inflow direction varies too drastically, turbulent structures may be compressed or stretched when fed into the target domain.

Muñoz-Esparza *et al.* (2014) proposed a turbulent inflow that accelerates the onset of turbulence in neutrally-stratified atmospheric boundary layers by introducing stochastic buoyancy perturbations. The buoyancy is generated by uniformly-random perturbations directly imposed on the potential temperature field at a uniform frequency. Muñoz-Esparza *et al.* (2014) dub the approach the cell perturbation method where perturbations are applied in cells of 8×8 grid points in the horizontal and one grid cell in height. The maximum amplitude of the temperature perturbations and the perturbation frequency were optimized in Muñoz-Esparza *et al.* (2015) by a single Eckert number and a single convective time scale. The method agrees well with other synthetic methods and a periodic simulation when used in a compressible, non-hydrostatic numerical weather prediction model.

The cell perturbation method has shown potential in neutral atmospheric boundary layers. In this study, we investigate and modify the cell perturbation method for wall-resolved LES of wall-bounded turbulent flows. These modifications originate from Umphrey and Senocak (2016), who attempted using the Eckert-number guidance in Muñoz-Esparza *et al.* (2015) but found that the optimal value for atmospheric boundary layers did not pertain to

an incompressible turbulent channel flow. Umphrey and Senocak (2016) reformulated the direct application of temperature perturbations as a source term in the temperature transport equation, converting the white-noise perturbations into colored-noise perturbations as the temperature transport equation is solved. The stochastic noise is colored in the sense that the LES filter operation will have some effect on the noise. The buoyancy is introduced through a Boussinesq approximation. The original perturbation cell shape was modified to have height greater than one grid cell to make the source term effective. Umphrey and Senocak (2016) referred to the technique as the *perturbation box method*. We build upon the perturbation box method by developing a non-dimensional formulation that can be applied to variety of flows. The improvements to the perturbation box method are tested in both a turbulent channel flow and turbulent flow over a backward-facing step, showing good agreement with DNS benchmarks.

7.2 Perturbation Box Technique

In this subsection, we describe the cell perturbation method proposed by Muñoz-Esparza *et al.* (2014). We then describe the modifications performed by Umphrey and Senocak (2016) to the cell perturbation method. Finally, we propose further modifications to the perturbation box method found in Umphrey and Senocak (2016).

7.2.0.1 Cell Perturbation Method for Atmospheric Flow

The cell perturbation method of Muñoz-Esparza *et al.* (2014) was applied to neutrally-stratified atmospheric boundary layers over flat terrain in the Weather Research and Forecasting (WRF) model (Skamarock *et al.*, 2008), a compressible, non-hydrostatic numerical weather prediction model with LES capabilities. The cell perturbation method applies uniform-random distribution of finite-amplitude perturbations to the temperature field in a buffer region near the inlet that are continuously updated at a certain frequency. The tem-

perature perturbations create vertical instabilities through buoyancy effects that transition rapidly into realistic turbulence within an amplitude range of ±0.5 K. The perturbations are applied directly to the temperature field as white-noise in cells of 8 × 8 grid points in the horizontal and one grid cell in the vertical, i.e. horizontal planes of 8 × 8 grid points. The optimum value of maximum perturbation amplitude and perturbation frequency were studied in Muñoz-Esparza *et al.* (2015). The optimal value for perturbation amplitude was formulated as a perturbation Eckert number given by

$$Ec = \frac{U_g^2}{\rho c_p \Delta\phi_{max}}, \tag{7.1}$$

where U_g is the geostrophic wind for the atmospheric boundary layer, c_p is the specific heat at constant pressure, and $\Delta\phi_{max}$ is the maximum perturbation amplitude defined as $(\phi - \phi_0)_{max}$ where ϕ_0 is the reference temperature. The optimum value was found to be $Ec = 0.16$, which gave rapid transition to turbulence that did not dissipate or create unrealistically long flow structures. The recommendation for perturbation frequency was a dimensionless time scale set to unity that took into account the time the mean flow is convected through a cell by the weakest velocity,

$$\frac{t_p U_1}{d_c} = 1, \tag{7.2}$$

where t_p is the perturbation update interval, U_1 is the velocity at the first grid point off of the surface, and d_c is the diagonal of the 8 × 8 perturbation cell given by $d_c = \sqrt{2} n_c \Delta x$, where n_c is the number of grid points on a perturbation cell side and Δx is the mesh spacing.

7.2.0.2 Perturbation Box Technique for Wall-bounded Flows

Umphrey (2015) and Umphrey and Senocak (2016) attempted adopting the cell perturbation method for incompressible, turbulent channel flow at $Re_\tau = 395$ by adding a Boussinesq approximation for buoyancy into the governing momentum equations. Using the bulk veloc-

ity in the perturbation Eckert number in Eq. 7.1 with a value of 0.16, the performance of the cell perturbation method was not adequate in the channel flow. An Eckert number of very small magnitude was needed. However, values that improved the first-order moments created more disagreement among the second-order moments and no optimal Eckert number could be found.

As the Eckert number does not appear to be an applicable non-dimensional number for turbulent channel flow, Umphrey and Senocak (2016) reformulated the maximum perturbation amplitude as a small fraction of the reference temperature. Umphrey and Senocak (2016) also formulated the temperature perturbations as a source term in the temperature transport equation rather than directly applying the perturbations to the temperature field. Due to the an implicit filtering operation, the white-noise perturbations of the cell perturbation method then became colored-noise perturbations. The noise is colored in the sense that the LES filter operation and the numerical solution of a transport equation mean the noise is no longer white. The horizontal planes of single-grid-cell height in the cell perturbation method then became volumes containing more than a single grid cell in height to make the source term effective, and the method was referred to as the *perturbation box* method. The value of the temperature perturbation is constant in a perturbation box just like in a perturbation cell, the difference being that more than one grid is contained in the vertical direction with the perturbation box method. The maximum perturbation amplitude, $\Delta\phi_{max}$, was specified as a constant value that was a fraction of the reference temperature. The performance was better than with the Eckert number cell perturbation method as second-order moments exhibited a consistent response to parameter tuning, which was not the case with the cell perturbation method. While Umphrey and Senocak (2016) found better response characteristics, the technique was not generalized to work with other flows and improvements in performance could still be made.

Wall-bounded turbulence has length and time scales varying with distance from the wall Pope (2000). Therefore in this study, we propose a formulation that adjusts the perturbation

amplitude and perturbation frequency based on the incoming mean velocity profile. Before presenting these formulations, we first discuss the sizing of the perturbation boxes. As shown in Keating *et al.* (2004), a proper turbulent cascade is achieved when turbulent scales larger than four times the integral length scale are generated by the turbulent inflow. Our proposition is to size the box height such that they are larger than the integral length scale but similar in scale and order of magnitude. Larger boxes may create large motions that take a significant development fetch to breakdown non-turbulent vortices, so we rely on the combinations of multiple boxes to create the larger vortical structures. The idea is that the large energy-containing range and the larger length scales in the inertial subrange are energized through the large buoyant motions. Therefore, we recommend sizing the wall-normal height of the box, H_{PB}, to 0.125 δ, where δ is the boundary-layer height at the inlet or the channel half-height. The reasoning is grounded in a discussion in Pope (2000, pg 240-42) explaining the demarcation between the energy-containing scales and the inertial subrange as one-sixth of the longitudinal integral length scale at high Reynolds numbers. Our suggested H_{PB} should fall within the lower range of the energy-containing scales and the combination of boxes should create necessary large-scale motions observed in Keating *et al.* (2004). We recommend maintaining a square shape in the horizontal directions, or close to a square shape, to preserve the advantage of being able to have time-varying mean flow directions with this method. To introduce some degree of anisotropy into the perturbations, we suggest $2H_{PB}$ for the length, L_{PB}, and width, W_{PB}, of the perturbation boxes. Muñoz-Esparza *et al.* (2015) used three perturbation cells in the inlet-normal direction. We adopt the same number perturbation boxes in the inlet-normal direction.

We propose to decide on the perturbation amplitude and perturbation frequency based on the incoming mean velocity profile while attempting to maintain simplicity in the method. For the amplitude, we propose a perturbation Richardson number, Ri_{PB}, formulation as the definition includes the buoyancy and mean shear. The bulk Richardson number to determine

the maximum perturbation amplitude is

$$Ri_{PB} = \frac{g\beta\Delta\phi_{max}H_{PB}}{\langle U(z)\rangle_{PB}^2}, \tag{7.3}$$

where g is the gravitational constant, β is the thermal expansion coefficient, and $\langle U(z)\rangle_{PB}$ is the incoming mean velocity profile averaged to the center of the perturbation box. Note that given a constant Ri_{PB} value, $\Delta\phi_{max}$ will vary with distance from the wall unlike previous approaches that used a uniform maximum amplitude for the entire perturbation zone. For the perturbation frequency, we adopt the same form of time scale given in Eq. 7.2 and set it to unity, but replace the velocity with $\langle U(z)\rangle_{PB}$ resulting in

$$\frac{t_p \langle U(z)\rangle_{PB}}{d_c} = 1. \tag{7.4}$$

For this study, $d_c = L_{PB}$. With this perturbation frequency, the update time will vary with wall normal distance, albeit the update time will be largest in the first off-wall perturbation box. While not coinciding with physical turbulent time scales, our proposition is to update the perturbation amplitude within a box once the previous perturbation has been convected through the streamwise length of the box. Note, Umphrey and Senocak (2016) formulated the update frequency in the same manner.

A plan view of the computational domain for the perturbation box method is sketched in Fig. 7.1. The perturbation zone was offset five grid cells from the inlet where a mean velocity profile is prescribed. To ensure that the buoyancy effects do not appear in the target domain, the Boussinesq approximation was removed from the momentum equations 6 L_{PB} after the initial offset. The region where the Boussinesq approximation was present is referred to as the buoyant region in Fig. 7.1. To ensure the global energy balance was not disrupted, all temperature perturbations were uniformly shifted every time step such that integration of the buoyant forces leads to thermal energy input. In this study, we used the properties of air at a reference temperature of $\phi_0 = 293$ K. As an ideal gas, $\beta = \phi_0^{-1}$.

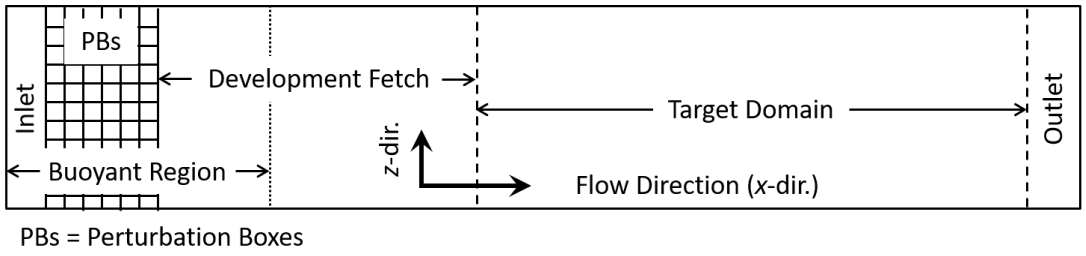


Figure 7.1: Plan-view sketch of the computational domain for the perturbation box technique. y-direction is orthogonal to x and z in the sketch.

7.3 Numerical Formulation

We solve the filtered Navier-Stokes equations for incompressible flows, starting with the continuity equation,

$$\frac{\partial \overline{u}_j}{\partial x_j} = 0, \tag{7.5}$$

and the momentum equations,

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u}_i \overline{u}_j\right) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\nu \overline{S}_{ij} - \tau_{ij}\right) + g\beta\left(\overline{\phi} - \phi_0\right)\delta_{i3}, \tag{7.6}$$

where δ is the Kronecker delta in this equation, the strain rate is defined as,

$$S_{ij} = \frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right), \tag{7.7}$$

and the residual Reynolds stresses are defined as,

$$\tau_{ij} = \overline{u_i u_j} - \overline{u}_i \overline{u}_j. \tag{7.8}$$

We also solve a temperature transport equation,

$$\frac{\partial \overline{\phi}}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{\phi}\overline{u}_j\right) = \frac{\partial}{\partial x_j}\left(\Gamma \frac{\partial \overline{\phi}}{\partial x_j} + \tau_{\phi j}\right) + \phi_{PB}, \tag{7.9}$$

where the residual turbulent heat flux is defined as,

$$\tau_{\phi j} = \overline{\phi u_j} - \overline{\phi}\overline{u}_j, \tag{7.10}$$

and ϕ_{PB} is the source term create by the perturbation box method.

The governing equations are solved using a GPU-accelerated, three dimensional incompressible flow solver (Thibault and Senocak, 2012; Jacobsen and Senocak, 2013; DeLeon *et al.*, 2013; Umphrey *et al.*, 2017). The code uses the projection algorithm (Chorin, 1968) on directionally-uniform Cartesian grids. The finite-difference schemes used were second-order central differences for spatial derivatives in the momentum equation and the diffusion term in Eq. 7.9, a first-order upwind difference for the convective term in Eq. 7.9, and a second-order Adams-Bashforth scheme for time advancement. The pressure Poisson equation is solved by an amalgamated parallel 3D geometric multigrid solver specifically designed for GPU clusters (Jacobsen and Senocak, 2011). Turbulence closure of the momentum equations is achieved using the localized Lagrangian dynamic Smagorinsky eddy viscosity model Meneveau *et al.* (1996) for LES. The sub-grid-scale (SGS) Reynolds stresses become

$$\tau_{ij} = -2\nu_t S_{ij} = -2\left(C_S\Delta\right)^2 |S| S_{ij}, \tag{7.11}$$

where the Smagorinsky model coefficient, C_S, is calculated through the Lagrangian dynamic procedure and the filter width, Δ, is set to the cube root of the grid cell volume. Turbulence closure for the energy equation is accomplished using a simple definition for turbulent

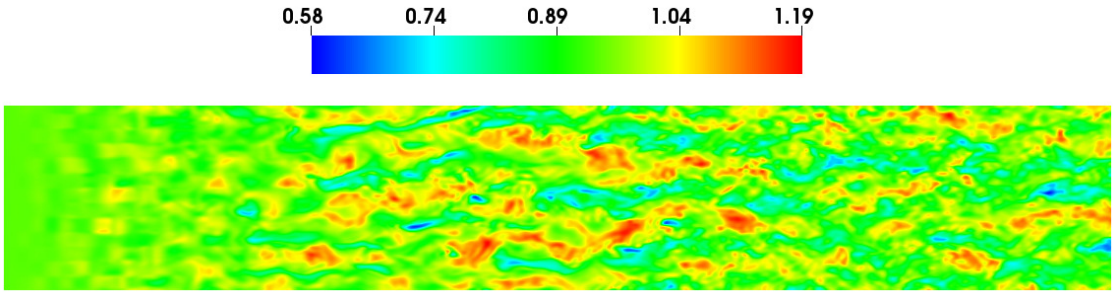


Figure 7.2: A wall-normal layer of the streamwise velocity component in the $Re_\tau = 395$ channel flow located at $y/\delta \approx 0.25$. The perturbation box method corresponds to $Ri = 0.042$ and $H_{PB} = 0.125\ \delta$. Streamwise velocity is normalized by bulk velocity.

diffusivity of

$$\tau_{\phi j} = \Gamma_t \frac{\partial \overline{\phi}}{\partial x_j} = \frac{\nu_t}{Pr_t} \frac{\partial \overline{\phi}}{\partial x_j}, \tag{7.12}$$

where Pr_t is a turbulent Prandtl number set to 0.85.

7.4 Channel Flow

In this subsection, we apply the perturbation box method described in § 7.2.0.2 to a turbulent channel flow at $Re_\tau = 395$. The entire computational domain was $6\pi\delta \times\ 2\ \delta \times \pi\delta$ in the streamwise, wall-normal, and spanwise directions, respectively, where δ is the channel half-height. The target domain to gather statistics was $3\pi\delta \times\ 2\ \delta \times \pi\delta$ and was placed such that 30 grid cells were between the downstream end of the target domain and the outlet (see Fig. 7.1). A directionally-uniform Cartesian grid of size $641 \times 257 \times 129$ was used, placing the first off-wall grid point at $y^+ = 1.5$. A mean velocity profile obtained from the direct numerical simulation (DNS) of Moser *et al.* (1999) was imposed at the inlet and the outlet was a convective type using the bulk velocity as the characteristic velocity scale. Spanwise boundary conditions were periodic and the no-slip condition was enforced at the walls. The simulation was allowed to develop for 13 convective time units, L_x/U_{bulk}, from a quiescent field and turbulent statistics were sampled for 17 convective time units thereafter.

As discussed in § 7.2.0.2, $H_{PB} = 0.125\ \delta$ and $L_{PB} = W_{PB} \approx 2\ H_{PB}$, where L_{PB} and W_{PB} are equal but slightly adjusted such that dividing the spanwise dimension by W_{PB} results in an integer value. After a systematic sensitivity analysis(DeLeon and Senocak, 2017), we found that $Ri_{PB} = 0.042$ gave good agreement with the DNS benchmark. A visualization of the developing channel flow is given in Fig. 7.2 under these criteria. The volumetric buoyancy perturbations are observed as faded boxes near the inlet as they perturb the flow in the wall-normal direction and subsequently trigger the formation of three-dimensional vortices. Once the turbulent energy cascade begins, the vortices start to break down into realistic turbulence.

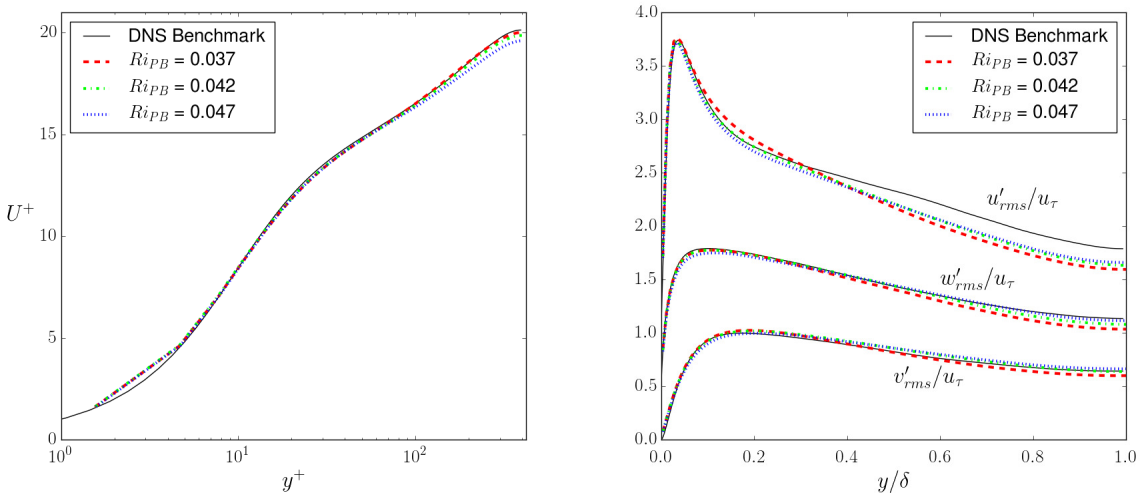


Figure 7.3: Perturbation box method sensitivity to Ri_{PB} in $Re_\tau = 395$ channel flow. (a) is the mean velocity and (b) is the root-mean-square of the turbulent fluctuations.

The mean velocity profile produced from Ri_{PB} of 0.037, 0.042, and 0.047 are presented in Fig. 7.3a. All three Ri_{PB} showed little to no sensitivity below $y^+ \approx = 80$. Past $y^+ \approx = 80$, sensitivity to Ri_{PB} is observed with an increasing velocity deficit at the core as Ri_{PB} increases. $Ri_{PB} = 0.037$ performs best regarding mean velocity. Figure 7.3b shows the root-mean-squares of the turbulent fluctuations with the same set of Ri_{PB}. Overall, good agreement is achieved with the spanwise and wall-normal components. The streamwise

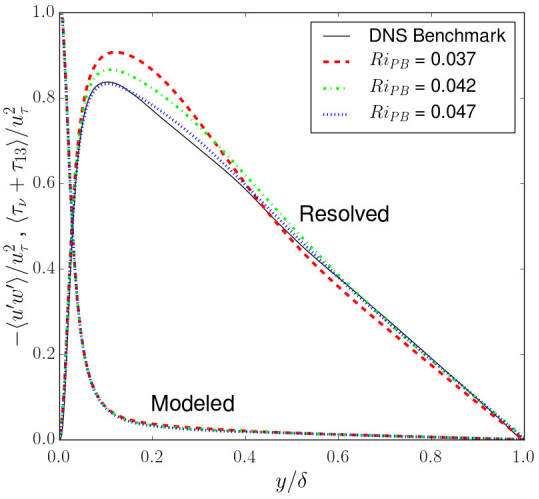


Figure 7.4: Shear stress sensitivity to Ri_{PB} in $Re_\tau = 395$ channel flow. Profile labeled modeled includes viscous shear stress.

component was captured well near the wall but the turbulence intensity falls short of the DNS in the core region, perhaps due to insufficient grid resolution. Also in the channel core, the fluctuation intensities increase with increasing Ri_{PB}. Near the wall, good agreement is achieved by all three Ri_{PB} but $Ri_{PB} = 0.042$ shows the best agreement among all three components in Fig. 7.3b, particularly with the streamwise intensity below $y\ /\ \delta \approx 0.3$. Figure 7.4 shows both the resolved and modeled shear stresses. $Ri_{PB} = 0.047$ clearly shows the best agreement with the DNS benchmark regarding the resolved Reynolds shear stress. One possible explanation is that the increased buoyancy perturbations create an 'artificial' turbulence production effect that transfers too much energy from the mean flow to the fluctuations. Hence, the decreased mean velocity in Fig. 7.3a with $Ri_{PB} = 0.047$. However, too small of buoyancy perturbations tends to not distribute the correct amount of energy to each of the components. Thus, we observe good agreement with mean velocity but too high Reynolds shear stresses and too low Reynolds normal stresses. A larger domain with a longer development fetch will most likely form the correct turbulent flow structure with the given buoyancy perturbations but the goal of synthetic methods is to reduce the required development length.

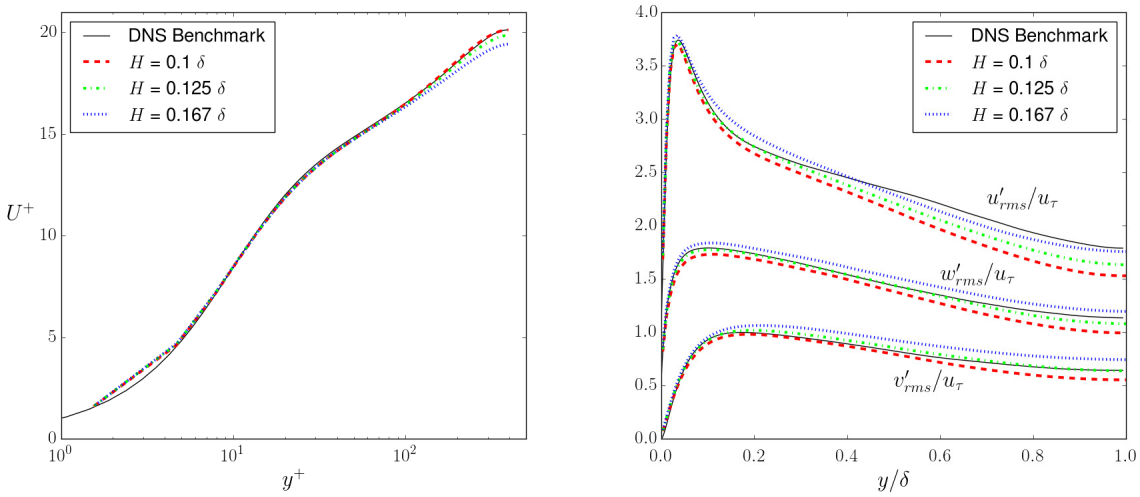


Figure 7.5: Perturbation box method sensitivity to H_{PB} in $Re_\tau = 395$ channel flow using $Ri_{PB} = 0.042$. (a) mean velocity (b) root-mean-square of the turbulent fluctuations

To gain a sense of the sensitivity varying perturbation-box size, we performed simulations with $H_{PB} = 0.1\ \delta$ and $H_{PB} = 0.167\ \delta$. Judging from Figs. 7.3 and 7.4, $Ri_{PB} = 0.042$ had the best balance in agreement among the mean velocity and Reynolds stresses and use that value here. The same relation of $L_{PB} = W_{PB} \approx 2\ H_{PB}$ in the Ri_{PB} sensitivity study is kept as well. Figure 7.5a shows the effect of altering the box size. Decreasing H_{PB} has a very similar effect as decreasing Ri_{PB} on the mean velocity. Again, the larger buoyancy perturbations could create an artificial production-term effect by transferring too much energy away from the mean velocity as the large vortical structures act against the mean velocity gradients. This hypothesis is reinforced by Fig. 7.5b as the larger perturbation box size exhibits a drastic increase in turbulent intensities relative to the baseline while the smaller perturbation box size produces too low of intensities. This means that too large of perturbation sizes will lead to very large waves in the flow, possibly requiring unreasonably long distances for realistic turbulence to develop. If the perturbations are too small, then there will not be enough production of turbulent kinetic energy.

Another objective is the reduction of the development fetch (see Fig. 7.1). For the

aforementioned perturbation box sizes, the development fetches were 10.52 δ, 10.77 δ, and 10.92 δ for $H_{PB} = 0.1\ \delta$, $H_{PB} = 0.125\ \delta$, and $H_{PB} = 0.167\ \delta$, respectively. We note that these numbers are not the exact lengths needed to obtain realistic turbulence. However, in the case of $Ri_{PB} = 0.042$ with $H_{PB} = 0.125\ \delta$, the exact length to fully-developed turbulent channel flow may not be that much greater than 10.77 δ. For reference, Poletto *et al.* (2013) was able to recover the streamwise normal stress in approximately 3.6 δ and the wall-normal stress in approximately 20 δ. Our hypothesis is that further optimization of the perturbation box method can lead to shorter development fetches.

7.5 Backward-facing Step

Applicability to a variety of flows is desirable for a turbulent inflow generation technique. We tested the same perturbation box parameters in §7.4 to a turbulent boundary-layer flow experiencing a sudden expansion when flowing over a backward-facing step. We compare to the $Re_h = 5100$ DNS results of Le *et al.* (1997), where Re_h is based on the freestream velocity, U_0, and h. Our domain is the same as in the DNS benchmark except with a smaller spanwise direction such that the domain size is 30 h × 6 h × 4 h, h being the step height, in the streamwise, wall-normal, and spanwise directions, respectively. The Cartesian resolution was 513 × 513 × 65. The step expansion was located at 10 h from the inlet. A zero-shear wall was applied to the top boundary while periodic boundary conditions were applied to the homogeneous spanwise direction. A convective outlet was used opposite the perturbation box inflow with a characteristic velocity equal to the U_0 scaled by the expansion ratio of 1.2. A mean profile from the flat-plate boundary-layer DNS of (Spalart, 1988) was imposed at the inlet corresponding to $Re_\theta = 670$ (θ being the momentum thickness). The bottom wall boundary conditions were imposed with an immersed boundary method using linear reconstruction schemes (Mohd-Yusof, 1997; Fadlun *et al.*, 2000). To capture the large separation region accurately, the first off-wall grid point was placed at $y^+ = 0.3$ (normalization by viscous scales determined from the incoming mean profile of the $Re_\theta =$

670 boundary layer).

The incoming turbulent boundary layer height can be expected to grow over a 10 h distance. The perturbation box method produces realistic turbulence after a development fetch but the question regarding a spatially-developing turbulent boundary layer is how far from the wall does the flow need to experience buoyancy perturbations to achieve the correct turbulent structure at the expansion? Therefore, two perturbation box heights were tested. The first height, $H_{PB,1}$, was 0.125 δ where δ is the boundary layer height based on the mean velocity profile at the inlet, where δ is 0.99 U_0 or $\delta^+ = 0.99\ U^+$ in nondimensional form. From nondimensional plots of mean velocity in $Re_\theta = 670$ turbulent boundary layer flow (Spalart, 1988), the nondimensional height corresponding to 0.99 U^+ would be $y^+ \approx 273$. The second height, $H_{PB,2}$, was $0.125 \times 1.2\ h$, which was the boundary layer height reported in Le *et al.* (1997). There is approximately a 10% difference between the two heights. This leaves approximately 8.90 h as a development fetch before the expansion. Based on results from the channel flow, $Ri_{PB} = 0.042$ showed the best balance between mean velocity and turbulent statistics. $Ri_{PB} = 0.047$ could have been used as well but we opted to use $Ri_{PB} = 0.042$ below the chosen heights, i.e. δ and 1.2 h. To maintain the freestream flow, we do not apply perturbation above the two respective heights.

Figure 7.6 is a snapshot of instantaneous streamwise velocity using $Ri_{PB} = 0.042$ and $H_{PB,1}$. The turbulence generates but does not seem to reach a realistic state before the step. Figure 7.7 shows the mean streamwise velocity at various streamwise locations in the domain and shows good agreement with the DNS benchmark with both perturbation box heights. The recirculation is captured very well in the mean sense. There are some overshoots in the streamwise velocity near $y\ /\ \delta \approx 1.5$ after the step and the profile has some disagreement with the DNS before the step is reached. Figure 7.8 shows the second-order moments at the same streamwise locations. Overall, there is very little sensitivity to the perturbation box heights except for the spanwise and wall-normal turbulent intensities before the step. This indicates that the perturbation box size may not be a major factor, although the difference was only

10%. The second-order moments before the step indicate that the development fetch was not adequate prior to the expansion. As observed in the channel flow, the streamwise turbulence intensity is underestimated relative to the DNS more so than the spanwise or wall-normal intensities. The Reynolds shear stress before the step shows good agreement. Regardless, the streamwise and spanwise intensities show good agreement below $y\ /\ \delta \approx 1.25$ after the step, and the wall-normal and Reynolds shear stress show good agreement for the entire domain height after the step. Possible explanations for the streamwise and spanwise discrepancies in the region between $y\ /\ \delta \approx 1.25$ and 2.25 could be attributed to the way the perturbation box method generates the turbulence at the top of the boundary layer and the interaction of the dynamic SGS model with the generated vortices that have yet to breakdown in the upper regions of the incoming turbulent boundary layer.

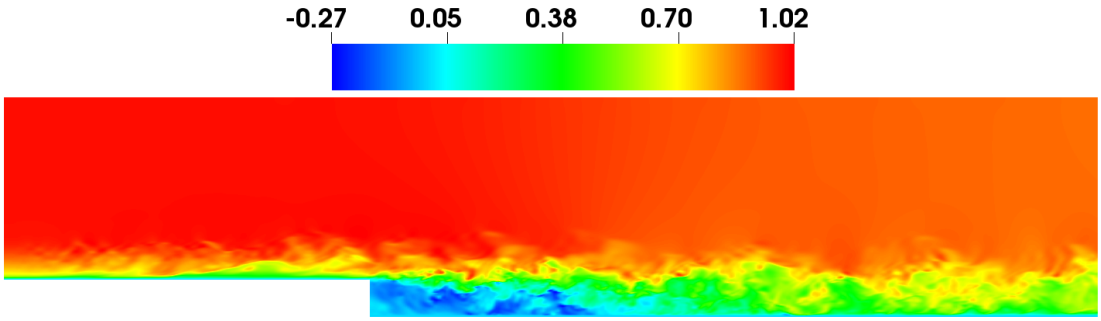


Figure 7.6: Instantaneous streamwise velocity generated from the perturbation box turbulent inflow in flow over a backward-facing step at $Re_h = 5100$ using $Ri_{PB} = 0.042$ and $H_{PB,1}$. Velocity normalized by U_0.

7.6 Summary and Conclusions

The improved version of the perturbation box turbulent inflow was presented. We proposed a Richardson number, Ri_{PB}, based on the incoming mean velocity profile and perturbation box height to specify the maximum temperature-perturbation amplitude. The temperature differences created by applying the stochastic temperature perturbations generate buoyancy effects near the inflow that accelerate the onset of realistic turbulence. The height of the

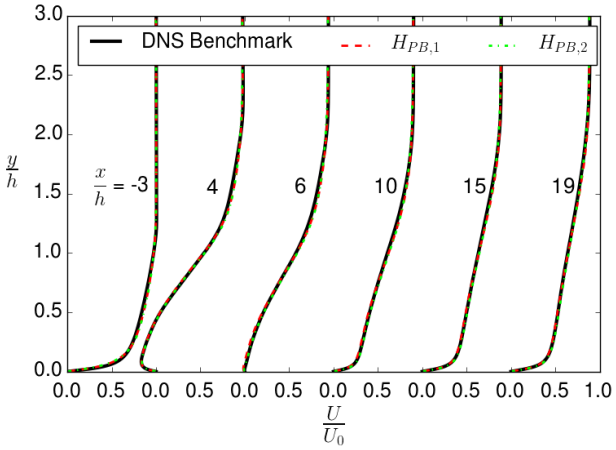


Figure 7.7: Mean streamwise velocity in flow over a backward-facing step at $Re_h = 5100$ using $Ri_{PB} = 0.042$. $H_{PB,1} = 0.125\ \delta$; $H_{PB,2} = 0.125 \times 1.2\ h$

perturbation boxes was suggested to be 0.125 δ and the length and width of the boxes be 0.25 δ, δ being a channel half-height or a turbulent boundary-layer height. The improved perturbation box method was applied to turbulent channel flow and turbulent boundary-layer flow over a backward-facing step. The results show the potential of the perturbation box method as a turbulent inflow generation technique for LES or DNS. For the channel flow, the mean velocity and the second-order moments showed overall good agreement with a DNS benchmark. Of the Ri_{PB} tested, $Ri_{PB} = 0.042$ was chosen as having the best balance between accuracy of first and second-order moments, particularly for its performance near the wall with turbulence intensities and Reynolds shear stress in the core. Using the same perturbation Richardson number for the backward-facing step, overall good agreement with a DNS benchmark was achieved among first and second-order moments after the step. The length before the step was not adequate to achieve good agreement with benchmark results before the expansion was reached.

The perturbation box method as presented in this study was intended to be a practical, low-cost model that could be used with a time-varying mean flow direction. The model does not require knowledge of turbulent moments beyond the first order. The method could benefit from introducing a recycling-type mechanism to reduce the development length.

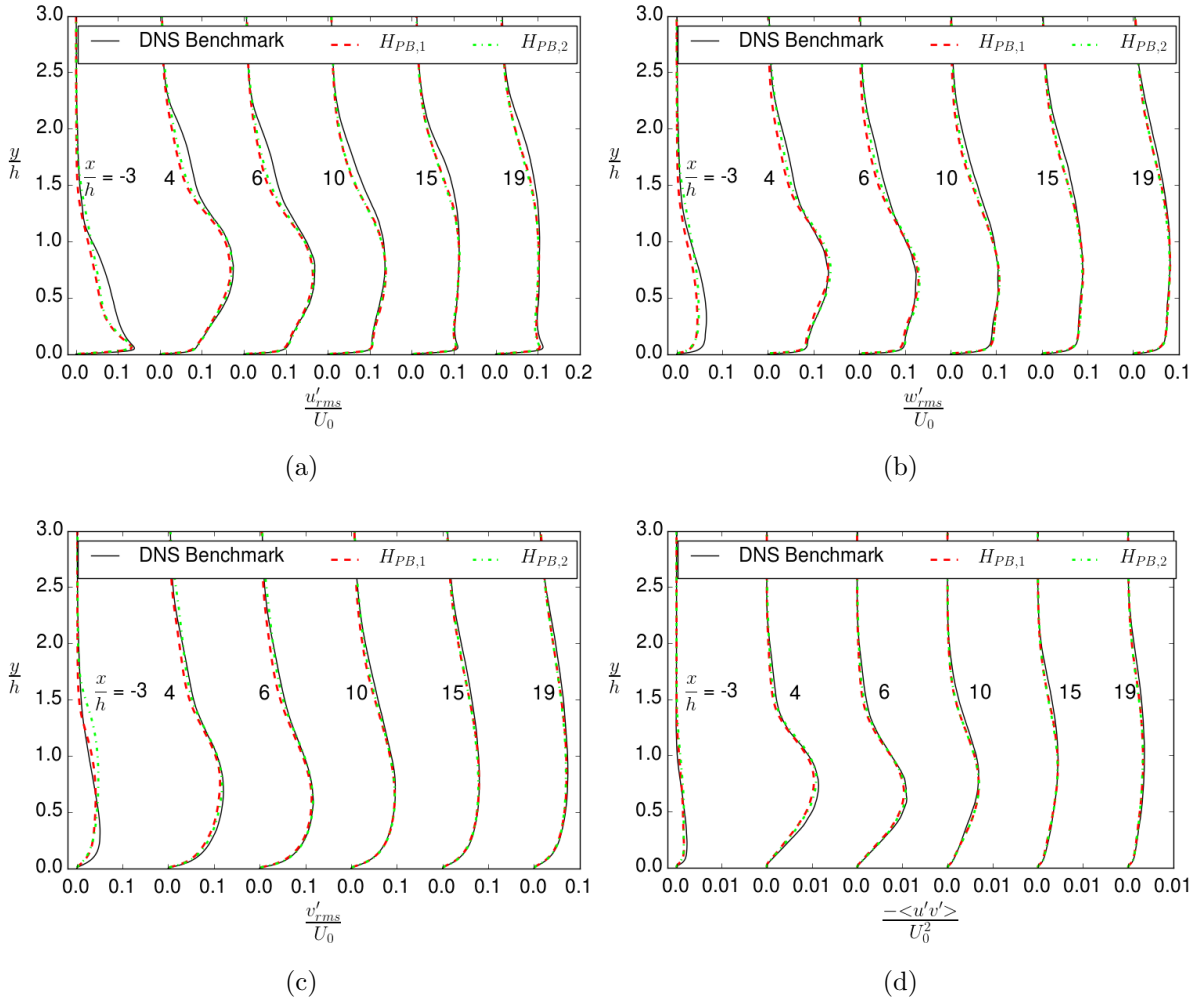


Figure 7.8: Second-order moments in flow over a backward-facing step at $Re_h = 5100$ using $Ri_{PB} = 0.042$. (a) streamwise intensity (b) spanwise intensity (c) wall-normal intensity (d) Reynolds shear stress. $H_{PB,1} = 0.125\ \delta$; $H_{PB,2} = 0.125 \times 1.2\ h$

Further testing is required for use in wall-modeled LES. Both will be part of future efforts.

CHAPTER 8

Globally-neutral Buoyancy Perturbations to Generate Turbulent Winds over Complex Terrain

8.1 Introduction

Large-eddy simulation (LES) is increasingly becoming the turbulence modeling technique of choice when modeling wind flow in the atmospheric boundary layer (ABL). The increasing interest in modeling wind farm power production and wind turbine wake dynamics in stratified or neutral conditions is a major driving factor in the numerous LES studies of the ABL (Mehta *et al.*, 2014). LES presents many challenges, e.g. modeling the residual sub-grid-scale Reynolds stresses, reducing the near-surface grid resolution requirements, and incoming lateral boundary conditions. The focus of this paper will be regarding the turbulent inflow conditions for LES.

The basic idea behind LES is to resolve turbulent eddies larger than a particular scale while modeling eddies with isotropic behavior that are smaller than said scale, the separation of scales performed by a low-pass filtering operation. The resolved flow field then has the correlations and flow structure of the turbulent flow and for LES to perform well with inflow and outflow boundary conditions, the inflow condition must contain or generate accurate turbulent correlations for the rest of the LES domain. Turbulent inflow conditions have been the focus of numerous studies over the past 25 years as discussed in the reviews of Keating *et al.* (2004), Tabor and Baba-Ahmadi (2010), and Wu (2017). In particular, Wu (2017) highlights the need for more robust inflow generation techniques for LES of atmospheric flows that take into account surface roughness and the three-dimensionality of the mean wind flow, i.e. time-varying mean flow directions.

Turbulent inflow generation techniques generally fall into two categories: precursor methods and synthetic methods (Tabor and Baba-Ahmadi, 2010). Precursor methods run an auxiliary boundary-layer simulation and feeds fully-developed turbulent flow information

into a target simulation at the boundaries. The obvious advantage is correct turbulent flow correlations are introduced into the target simulation but the obvious disadvantage is having to perform an auxiliary simulation. Synthetic methods seek to generate realistic turbulence in the incoming flow. However, as the turbulent flow correlations are typically not known in advance for general flows, synthetic methods require a development fetch before accurate turbulence is realized. The potential of synthetic methods is that a development fetch shorter than a separate auxiliary domain would greatly reduce computational overhead for generating turbulence.

A common turbulent inflow generation technique for atmospheric-related LES is the precursor method (Lopes *et al.*, 2007; Diebold *et al.*, 2013; Porté-Agel *et al.*, 2013; Munters *et al.*, 2016b). The auxiliary domain stage in the precursor method consists of obtaining a converged solution in a doubly periodic domain driven by either a constant mass flow rate (Lopes *et al.*, 2007) or a constant mean pressure gradient (Diebold *et al.*, 2013). Often, the auxiliary domain is stored as an independent database that can be read in by the target simulation. Stevens *et al.* (2014) proposed having a concurrent precursor methodology that transferred the inflow data directly to the target simulation from a synced auxiliary simulation without having to write a database. Munters *et al.* (2016b) proposed a concurrent precursor that mapped the auxiliary flow field onto the target domain dependent on mean flow direction.

A category of precursor methods that can also avoid the creation of inflow databases are those referred to as recycling methods. Some authors refer to recycling methods as a different category of turbulent inflows but recycling methods can be used to improve basic precursor methods. The most well-known of the recycling methods is that of Lund *et al.* (1998) applied to engineering-type boundary layers. The flow is sampled several boundary-layer thicknesses downstream of the target and the developed turbulent field is then introduced at the inlet after being scaled by inner-layer and outer-layer similarity laws to account for the growing turbulent boundary layer. However, equilibrium is required for similarity laws

to apply. Mayor *et al.* (2002) applied the recycling idea to develop homogeneous turbulence that would then become the inflow condition for an inhomogeneous flow field when regarding mesoscale internal boundary layers. The original method of Lund *et al.* (1998) used empirical relationships when applying the scalings between the recycling location and the inflow which led Araya *et al.* (2011) to propose using a test plane to determine flow parameters dynamically.

Most synthetic methods attempt to apply turbulent flow structure directly on the velocity field to generate turbulence. However, recent wind farm studies have found that applying mean shear and turbulent fluctuations independently can lead to erroneous results (Mehta *et al.*, 2014; Park *et al.*, 2014). Regardless, synthetic methods have shown potential. The synthetic method of Mann (1998) constructs a spectral tensor by applying rapid distortion theory to linearized Navier-Stokes equations. While turbulence is generated, it may not necessarily be physical. Regardless, the Mann synthetic method has been used in a variety of ABL-related studies (Munters *et al.*, 2016b). Xie and Castro (2009) proposed generating a two-dimensional turbulent plane using digital filters on zero-mean, unit-variance random data. Using the observation that correlation functions in shear flows follow a decaying exponential, each two-dimensional plane was correlated with planes at previous time steps using a decaying exponential weighting function. Good agreement was achieved with wind-tunnel experiments of urban canopies.

However, the aforementioned synthetic methods, as well as most synthetic methods, rely on knowing the Reynolds stress tensors or turbulent spectral content in advance. If Reynolds stresses aren't known, then time-varying mean flow directions may not be possible. Additionally, a synthetic method may not be divergence-free depending on the imposition of the correlations. A synthetic method that avoids these shortcomings is the cell perturbation method of Muñoz-Esparza *et al.* (2014) for the Weather and Research Forecasting model (Skamarock *et al.*, 2008). Stochastic temperature perturbations, update at a certain frequency, are directly imposed as white noise onto planes of the potential temperature field in

a buffer region near the inlet. Through buoyancy created by the stochastic temperature differences, the flow field experiences perturbations that develop into three-dimensional vortices which will break down into realistic turbulence. The parameters of an Eckert number for temperature perturbation amplitude and a time scale for the perturbation update frequency were optimized in Muñoz-Esparza *et al.* (2015). The method has the advantages requiring minimal knowledge of the turbulent flow structure and ability to adjust to time-varying mean flow directions.

Umphrey and Senocak (2016) tested the proposition of Muñoz-Esparza *et al.* (2015) in a canonical incompressible turbulent channel flow using a Boussinesq buoyancy approximation and found that the Eckert number was not the appropriate nondimensional number for that flow case. Umphrey and Senocak (2016) then proposed the perturbation box method. The Eckert number guideline for maximum temperature perturbation amplitude was replaced with a specified uniform maximum temperature perturbation. The direct imposition of temperature perturbations on horizontal planes were reformulated as volumetric source terms in a temperature transport equation, converting the white-noise perturbation into colored-noise perturbation as the low-pass LES filter operation and solution of the transport equation result in the noise no longer being white. The result was improved but was not formulated in a general manner that could be applied to various wall-bounded flows. In the turbulent channel flow study of DeLeon and Senocak (2017), a bulk Richardson number is formulated based on an incoming mean velocity profile, meaning the temperature perturbation amplitude varies with distance from a wall.

In this study, we present and explore the formulation in DeLeon and Senocak (2017), originally derived for channel flows, and apply it to turbulent wind flow over complex terrain. We apply the Richardson-number-based perturbation box method in a wall-modeled LES with an immersed boundary method for complex terrain. We use the commonly-known complex terrain benchmarks of Askervein Hill (Walmsley and Taylor, 1996) and Bolund Hill (Berg *et al.*, 2011) and compare the perturbation box method to a periodic solution and ex-

perimental data. The focus of the study is to investigate perturbation box sizes and observe the length of the development fetch. Good agreement is acheived with the perturbaiton box technique with regard to acceleration of turbulent wind over complex terrain with larger perturbation boxes producing larger-scale flow structures that play a significant and necessary role in the transport of momentum.

8.2 Perturbation Box Technique

Before presenting the perturbation box method, we first discuss the cell perturbation method of Muñoz-Esparza *et al.* (2015) that was implemented in the Weather and Research Forecasting numerical weather prediction model(Skamarock *et al.*, 2008). First proposed by Muñoz-Esparza *et al.* (2014), the cell perturbation method directly applied uniformally-random, zero-mean perturbations as white noise to the potential temperature field in a buffer region near the inlet. The perturbation cells were updated at a certain frequency. The perturbations were applied in horizontal planes of 8×8 grid cells with the maximum perturbation amplitude uniform in all planes. The size of the planes was motivated by the numerical dissipation present in the WRF model(Skamarock *et al.*, 2008). The inflow condition was tested and compared to the synthetic method of Xie and Castro (2009) in neutral ABL flow over flat terrain. Muñoz-Esparza *et al.* (2015) further optimized the perturbation cell method by formulating a global perturbation Eckert number that governed the maximum perturbation amplitude and a global perturbation time scale that controlled the update frequency. Every perturbation cell had the same amplitude and frequency. The perturbation Eckert number was defined as

$$Ec_{CP} = \frac{U_g^2}{\rho c_p \Delta\phi_{max}}, \tag{8.1}$$

where the subscript CP is for cell perturbation, U_g is the geostrophic wind for the atmospheric boundary layer, c_p is the specific heat at constant pressure, and $\Delta\phi_{max}$ is the

maximum perturbation amplitude. The optimum value was found to be $Ec = 0.16$, which gave rapid transition to turbulence that did not dissipate or create unrealistically long flow structures. The dimensionless time scale was defined as

$$\Gamma_{CP} = \frac{t_p U_1}{d_c}, \tag{8.2}$$

where t_p is the perturbation update interval, U_1 is the velocity at the first grid point off of the surface, and d_c is the diagonal of the 8 $\times$ 8 perturbation cell given by $d_c = \sqrt{2} n_c \Delta x$, where n_c is the number of grid points contained along a perturbation cell side that were spaced Δx apart. The optimum time scale was reported as unity.

Umphrey and Senocak (2016) attempted using the perturbation cell method with Ec_{CP} = 0.16 and $\Gamma_{CP} = 1$ in an incompressible turbulent channel flow discovering that the Eckert number needed to be very small in magnitude to work in this flow. The temperature field was coupled to the momentum field by using a Boussinesq approximation for buoyancy. Even so, tuning the parameter to agree well with the mean velocity resulted in increased disagreement with second-order moments. This inconsistency motivated a reformulation of the perturbation amplitude formulation and introduction of perturbations into the temperature field. Instead, a fraction of the reference temperature, ϕ_0, was used as the maximum perturbation amplitude. The perturbation was formulated as a source term into the temperature transport equation by dividing the stochastic perturbation by the update frequency, converting the white-noise operation of direct addition of the perturbations into a colored-noise operation due to LES filtering and numerical solution of a transport equation. To make a source term effective, the horizontal planes of one grid cell in height become volumes of multiple vertical grid cells. The time scale in Eq. 8.2 was also reformulated to replace U_1 with $U(z)$ and d_c with the streamwise length of the perturbation box. The reformulation was referred to as the *perturbation box* method.

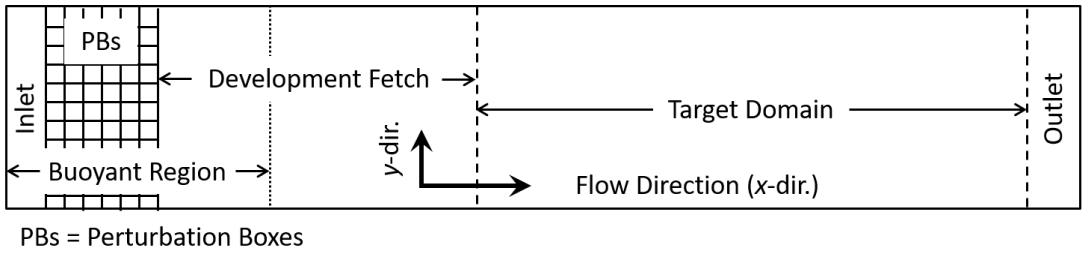


Figure 8.1: Plan-view sketch of the computational domain for the perturbation box technique. z-direction is orthogonal to x and y in the sketch.

DeLeon and Senocak (2017) advanced the perturbation box method by proposing a non-dimensional formulation that could be applied to a variety of flows. A sketch of an example domain is given in Fig. 8.1. The non-dimensional proposition for perturbation amplitude was the following bulk Richardson number,

$$Ri_{PB} = \frac{g\beta\Delta\phi_{max}H_{PB}}{\langle U(z)\rangle^2_{PB}}, \tag{8.3}$$

where g is the gravitational constant, β is the thermal expansion coefficient, and $\langle U(z)\rangle_{PB}$ is the incoming mean velocity profile averaged to the center of the perturbation box. This Richardson number formulation allows the temperature perturbation amplitudes to vary with distance from a wall, consistent with the behavior of wall-bounded turbulence. The same dimensionless time scale as in Umphrey and Senocak (2016) was used with $\langle U(z)\rangle_{PB}$ and set to unity. When a perturbation is convected fully through the length of the box at the average velocity at the box center, the box receives a new stochastic perturbation. To ensure zero thermal energy input, we shift each perturbation amplitude uniformly such that integration yields a global energy balance and the flow remains neutral. This perturbation box method was tested in turbulent channel flow and turbulent flow over a backward-facing step. For engineering flows, we only applied the Boussinesq buoyancy approximation to a portion near the inflow, twice the streamwise length of the perturbation box buffer zone, described as the buoyant region in Fig. 8.1. In this study, we adopt this perturbation box

method as described here. An ABL flow simulation of complex terrain would typically not remove the momentum-temperature interactions. However, the temperature perturbations quickly dissipate past the buffer region and would not impact the neutral condition.

8.3 Numerical Formulation

We solve the filtered form of the incompressible Navier-Stokes equations

$$\frac{\partial \overline{u}_j}{\partial x_j} = 0, \tag{8.4}$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u}_i \overline{u}_j\right) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\nu \overline{S}_{ij} - \tau_{ij}\right), \tag{8.5}$$

where the $\overline{S}_{ij}$ is the strain rate tensor and τ_{ij} are the filtered Reynolds stresses which are modeled with an eddy-viscosity assumption as follows

$$\tau_{ij} = \nu_t \overline{S}_{ij}, \tag{8.6}$$

$$\nu_t = l_{mix}^2 |\overline{S}|. \tag{8.7}$$

We also solve a temperature transport equation,

$$\frac{\partial \overline{\phi}}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{\phi}\overline{u}_j\right) = \frac{\partial}{\partial x_j}\left(\Gamma \frac{\partial \overline{\phi}}{\partial x_j} + \tau_{\phi j}\right) + \phi_S, \tag{8.8}$$

where $\tau_{\phi j}$ is the filtered turbulent heat flux and ϕ_S is a source term, created by the perturbation box method in this study. The turbulent heat flux is modeled by

$$\tau_{\phi j} = \frac{\nu_t}{Pr_t}\frac{\partial \overline{\phi}}{\partial x_j}, \tag{8.9}$$

where Pr_t is a turbulent Prandtl number set to 0.85.

We use a second-order Adams-Bashforth scheme for time advancement and discretize the spatial derivatives with a second-order central difference scheme on a directionally-uniform

Cartesian grid. The temperature equation uses a first-order upwind for the convection term. The three dimensional incompressible flow solver, for which density is set to unity and the pressure Poisson equation is solved by a geometric multi-grid method, has been parallelized for fast computing on a cluster of graphics processing units (Jacobsen and Senocak, 2011; Thibault and Senocak, 2012; Jacobsen and Senocak, 2013; DeLeon *et al.*, 2013; Umphrey *et al.*, 2017).

We make use of Reynolds-averaged Navier-Stokes (RANS) to reduce the surface resolution requirements by using the hybrid RANS-LES model described in Senocak *et al.* (2007). The mixing lengths of the LES sub-grid scale (SGS) eddy viscosity model are blended with the Prandtl mixing length (Prandtl, 1925), such that l_{mix} in Eq. 8.7 becomes

$$l_{mix} = (1 - \exp(-h/h_B))\, C_S \Delta + \exp(-h/h_B)\, \kappa h, \tag{8.10}$$

Note that we blend the length scales directly instead of the squared quantities that was originally proposed in Senocak *et al.* (2007). The Smagorinsky coefficient, C_S, is computed using the localized Lagrangian dynamic Smagorinsky SGS model (Meneveau *et al.*, 1996). In Eq. 8.10, h is the surface-normal distance from the terrain, h_B represents the RANS-LES blending height, and Δ is the LES filter width defined from the cube root of a grid cell volume as $\sqrt[3]{\Delta x \Delta y \Delta z}$.

For complex terrain boundary conditions, we use a direct-forcing immersed boundary (IB) method (Mohd-Yusof, 1997; Fadlun *et al.*, 2000; Umphrey *et al.*, 2017). We use the same procedure found in Umphrey *et al.* (2017) except we will perform wall-modeled LES rather than direct numerical simulation, thus requiring an IB reconstruction scheme applicable to logarithmic mean velocity profiles, i.e. linear reconstruction will give erroneous results. We adopt the approach of Senocak *et al.* (2004) that applies a logarithmic reconstruction scheme to velocity tangent to the surface and a linear reconstruction scheme to the surface-normal velocity to ensure the impermeability condition is enforced. Eddy viscosity is also linearly

reconstructed near the surface as the calculation of $\overline{S}_{ij}$ will result in large errors at the immersed surface.

8.4 Askervein Hill

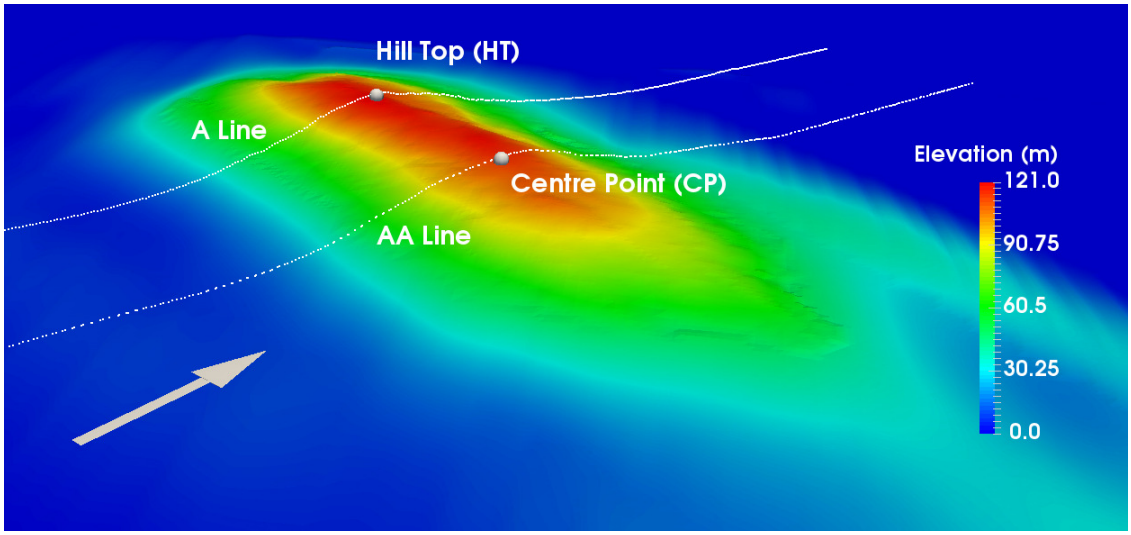


Figure 8.2: Askervein Hill elevation map. The lines designated A and AA are the sampling locations for comparison to the actual Askervein Hill field experiment. The large spheres represent locations designated hill top (HT) and centre point (CP) during the measurement campaign. The arrow indicates mean flow direction as well as positive x-direction.

We first apply the perturbation box method to the Askervein Hill case, which was the focal point of a field measurement campaign in the 1980s (Taylor and Teunissen, 1987; Mickle *et al.*, 1988; Salmon *et al.*, 1988; Walmsley and Taylor, 1996). The topography is shown in Fig. 8.2. The hill has been the focus of many LES studies including the hybrid RANS-LES of Lopes *et al.* (2007), the Advanced Regional Prediction System (ARPS) (Xue *et al.*, 2003) LES of Chow and Street (2009), and the Coupled Ocean/Atmosphere Mesoscale Prediction System (COAMPS ®) (Golaz *et al.*, 2005) LES of Golaz *et al.* (2009). We compare the perturbation box method to the field data measurements as well as our own periodic simulation. The RANS-LES blending height, h_B, was set to 16.5 m to maintain the condition, $h_B(2\Delta)^{-1} > 1$, that ensures regions that lack of sufficient resolution for LES are modeled with RANS instead Senocak *et al.* (2007).

Case	L_{PB} (m)	W_{PB} (m)	H_{PB} (m)	Boxes in x	Boxes in y	Boxes in z
Ask1	23	23	12	3	247	84
Ask2	76	76	38	3	76	26
Ask3	123	123	62	3	47	16

Table 8.1: Summary of perturbation box technique parameters used for flow over Askervein Hill. L_{PB}, W_{PB}, and H_{PB} are the length, width, and height of the perturbation boxes, respectively.

In our simulations, the domain size for the Askervein simulation was 8218.0 m $\times$ 5800.0 m $\times$ 1000.0 m in the streamwise (x), horizontal spanwise (y) and vertical (z) directions with a starting grid resolution of 641 $\times$ 513 $\times$ 257 in the same respective directions. We use the terrain map that does not include neighboring hills (Walmsley and Taylor, 1996) and immersed it such that the surrounding flat surface is located 14.7 m above the bottom of the domain. As suggested by Taylor and Teunissen (1987), we use an aerodynamic roughness length, z_0, of 0.03 m. The top boundary condition was a zero-shear-stress wall and the terrain surface boundary conditions are calculated by the IB method described in §8.3. Spanwise lateral boundary conditions were standard periodic boundary conditions.

For the periodic reference case, the streamwise boundary conditions were the shifted periodic boundary conditions of Munters *et al.* (2016a). The shifted periodic boundary conditions are designed to avoid the spanwise locking of flow structures by shifting the outflow condition a set distance in the spanwise direction during the periodic recycling. The end result is shorter domains than are required with standard periodic conditions, because the flow structures near the boundary become decorrelated. For Askervein Hill, we use a shift of 1000.0 m for the shifted periodic boundary conditions and drive the flow by dynamically calculating the driving mean pressure gradient to keep a constant target mass flow rate of approximately 9.01×10^7kg s^{-1} using the formulation of Benocci and Pinelli (1990).

For the perturbation box method, a mean profile was prescribed at the inlet based on a neutrally-stable, rough-surface log-law using $u_* = 0.68$ m s^{-1}. Note the higher value than the recommended $u_* = 0.654$ (Walmsley and Taylor, 1996) is due to dissipation over the development fetch. The inlet profile was set to a constant value approximately 850 m above

the surrounding flat terrain to reduce the numerical stability constraints in our explicit scheme. Testing with the full log-law extending to the top did not show any sensitivity in the solution near the surface. The outflow was the convective type with the bulk velocity as the characteristic velocity. ϕ_0 was set to 293 K. Air was assumed meaning that $\beta = \phi_0^{-1}$ in Eq. 8.3. We focus our study on the effect of perturbation box size, the values and number of boxes in each respective direction are given in Tab. 8.1. Based on results from DeLeon and Senocak (2017), we set $Ri_{PB} = 0.042$ and hold this constant for all trials. DeLeon and Senocak (2017) recommended the perturbation box height, H_{PB}, be approximately one-eighth of the turbulent boundary layer height and the perturbation box length and width are twice the height, i.e. $L_{PB} = W_{PB} = 2H_{PB}$. Regarding the ABL, setting H_{PB} to one-eighth the boundary layer height would result in very large buoyancy perturbations that would take a significant development fetch to break down into fine-scale turbulence. Therefore, we seek perturbation box sizes based on fractions of other length scales. In Lopes *et al.* (2007), the Obukhov length of 600 m was used in their analysis. Cases Ask2 and Ask3 in Tab. 8.1 are approximately one-sixteenth and one-eighth of that Obukhov length scale, respectively. As the turbulent stresses greatly affect the mean flow in the inner layer, we set the height of H_{PB} in Case Ask1 to the inner layer height as a physically-small buoyancy perturbation will break down quickly. The value for the height of the inner layer is also given in Lopes *et al.* (2007).

The turbulent eddies generated at 100 m above the flat terrain by each of the perturbation box dimensions in Tab. 8.1 are shown in Fig. 8.3. Unsurprisingly, the smallest perturbation boxes break down rapidly into smaller turbulent eddies while the larger boxes takes a much longer development fetch. Given the current domain, case Ask1 and Ask 2 break down into realistic turbulence before the wind flows over the hill. Case Ask3 does not, obviously requiring a longer domain than the one used in the present study. To assess the impact of the turbulent inflow on the mean flow, we compare results to the case of MF03-D taken between 1400-1700 BST (British Summer Time = UTC + 1h) on 3 October 1983 (Taylor

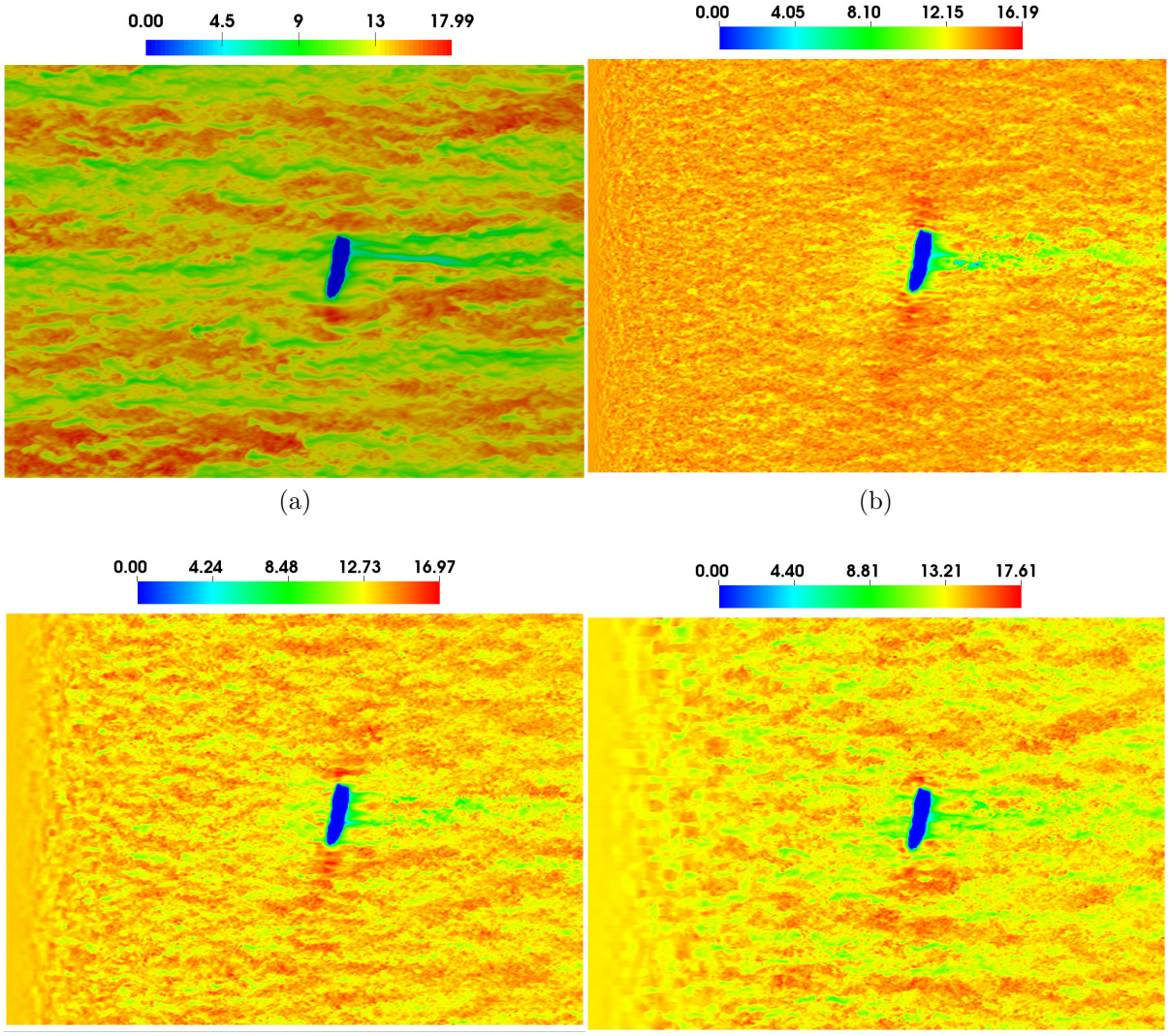


Figure 8.3: Visualization of the different perturbation box methods used for the Askervein Hill case. Horizontal wind speed given in units of m s^{-1} and is taken at approximately 100 m above the surrounding flat surface. (a) Periodic (b) Case Ask1, $H_{PB} = 12$ (c) Case Ask2, $H_{PB} = 38$ (d) Case Ask3, $H_{PB} = 62$ (see Tab. 8.1)

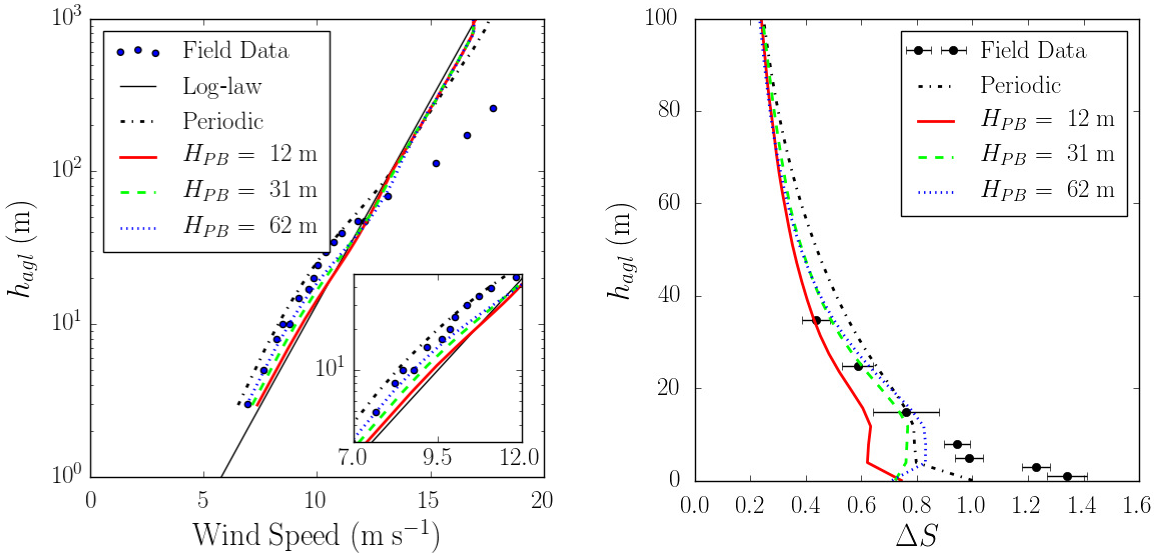


Figure 8.4: (a) Mean velocity at the reference site (RS). (b) Fractional speedup ratio at hill top (HT). The rough-surface log-law in (a) is plotted with $z_0 = 0.03$ m and $u_* = 0.654$ m s^{-1}. The insert in (a) is a focus between $h_{agl} = 3$ m to $h_{agl} = 50$ m.

and Teunissen, 1985) that is commonly used for near-neutral atmospheric conditions. For comparison, we use the fractional speed-up ratio (Jackson and Hunt, 1975) to compare LES results with the experimental benchmark. Fractional speed-up ratio is the horizontal velocity speed up relative to an undisturbed mean velocity profile,

$$\Delta S = \frac{U(h_{agl}) - U_{ref}(z)}{U_{ref}(z)}, \tag{8.11}$$

where h_{agl} and z are both vertical heights above ground level of the terrain except z contains the global elevation datum.

Figure 8.4 shows the horizontal wind speed at the reference site (RS) and the fractional speedup ratio at the hill top (HT). Locations RS and HT can be found in Walmsley and Taylor (1996). Given that the inlet was prescribed a larger u_* value than the recommended value in Walmsley and Taylor (1996), the mean wind speed at RS produced by the perturbation box method in Fig. 8.4a agrees well with both the experimental data and the surface-layer

logarithmic profile for horizontal velocity with $z_0 = 0.03$ m and $u_* = 0.654$ m s^{-1}. The perturbation box method produces better agreement with the neutral-stability log-law than does the reference periodic simulation. Upon close inspection, the larger perturbation boxes produces more variation away from the neutrally-stable log-law when $h_{agl} < 60$ m. Above 60 m, the profiles collapse onto the log-law. The deviation away from the log-law below 60 m could be attributed to the vortices created by the stochastic buoyancy requiring more development.

The fractional speedup ratio in Fig. 8.4b shows the effect of the flow structure size on how the flow accelerates over the gentle hill. The two largest perturbation boxes show a significant impact on the flow acceleration close to the ground at HT. Both the two largest perturbation box sizes also show the best agreement with the experimental data after a height of $h_{agl} \approx 15$ m, outperforming the periodic reference simulation even though case Ask3 has not fully developed into realistic turbulence. The smallest perturbation box is drastically lower in fractional speedup ratio. The size of the flow structures has a significant effect on the acceleration over the hill most likely due to the transfer of momentum away from the ground being diminished with the fine-scale turbulent eddies. The larger eddies seem to play an important role in the momentum transfer and are needed to observe the acceleration of the experimental data. Note, we do not seek good agreement below the RANS-LES interface in our hybrid RANS-LES approach as the RANS eddy viscosity will dampen the turbulence near the surface.

Figures 8.5 and 8.6 show the fractional speedup ratio along the AA and A lines at 10 m above ground level. On the windward side of the hill, both the two largest perturbation boxes and the periodic simulation show good collapse with minor sensitivity, even though the largest perturbation box requires a longer development fetch. On the other hand, the smallest perturbation box, which developed into full turbulence, shows a significantly lower fractional speedup ratio. This could be due to the missing role large-scale eddies play in flow over complex terrain. The periodic reference simulation and the two largest perturbation

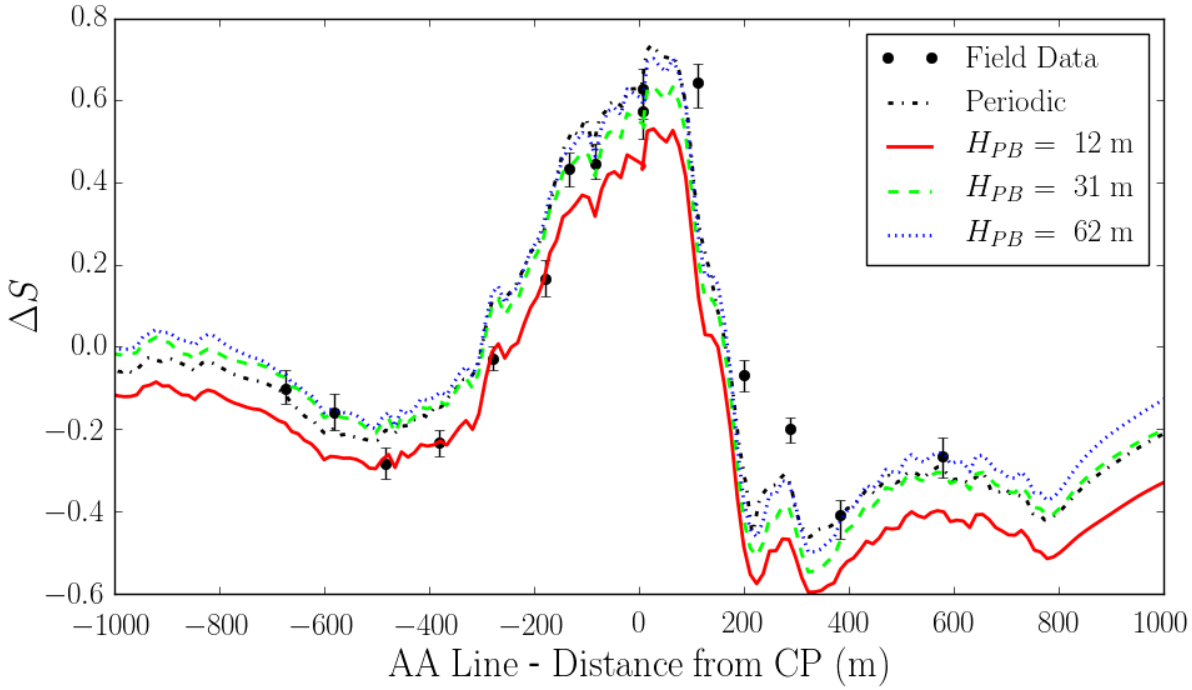


Figure 8.5: Fractional speedup ratio along the AA Line at 10 m above ground level.

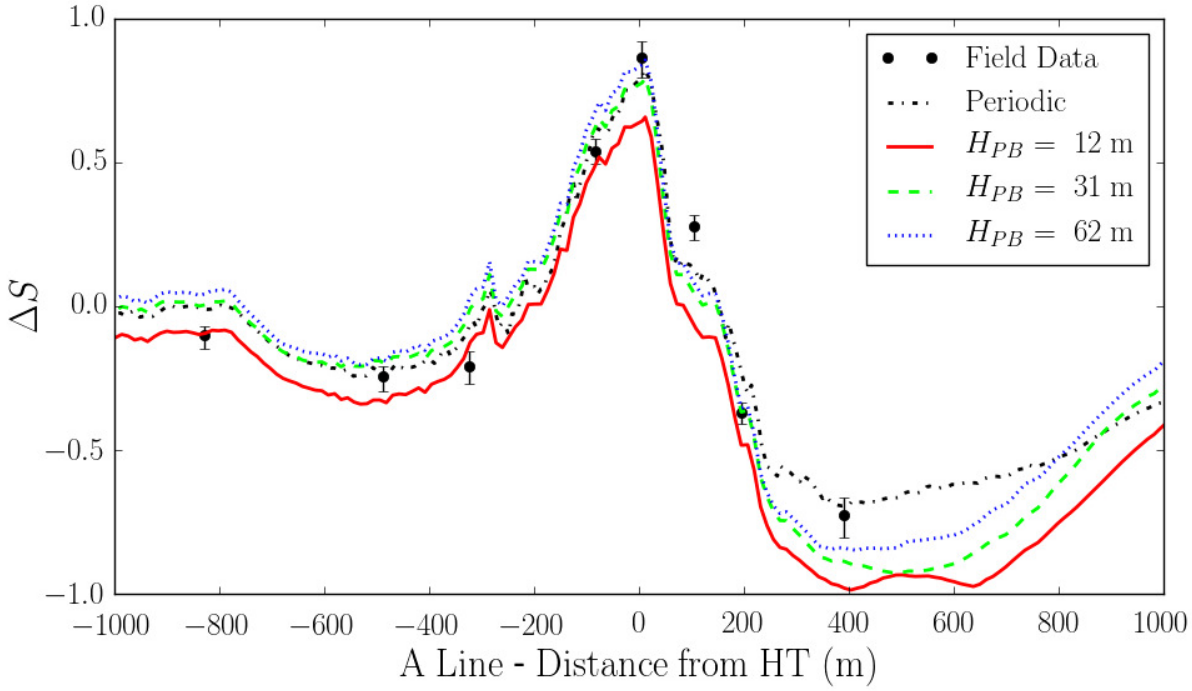


Figure 8.6: Fractional speedup ratio along the A Line at 10 m above ground level.

Case	L_{PB} (m)	W_{PB} (m)	H_{PB} (m)	Boxes in x	Boxes in y	Boxes in z
Bol1	4	4	2	3	188	50
Bol2	35	35	17	3	22	7

Table 8.2: Summary of perturbation box technique parameters used for flow over Bolund Hill. L_{PB}, W_{PB}, and H_{PB} are the length, width, and height of the perturbation boxes, respectively.

boxes do not show much sensitivity on the leeward side of the hill as seen in Fig. 8.5. The separation behind the hill along the AA line is not quite captured, but explanation lies with the attached-flow assumptions in the turbulence model. The leeward side of the A line is experiences the full shadow of the hill and more flow separation as shown in Fig. 8.3. Thus, we see large sensitivity on the leeward side of the hill along the A line. Understanding the sensitivity in the wake of the hill would be of great benefit but such a study would require a turbulence model more suitable for separated flow, which is outside the scope of the this study.

8.5 Bolund Hill

We wish to observe the development of turbulence when a finer grid is used and the impact that smaller-scale structures have on turbulent wind flow over a hill. A terrain benchmark smaller than Askervein is Bolund Hill, a coastal hill approximately 12 m tall and located in Denmark. It was the focus of a field study (Berg *et al.*, 2011) followed by a blind evaluation of computer models (Bechmann *et al.*, 2011). The hill is completely surrounded by water at certain times of the year and features a steep escarpment which makes it a challenging case for wind solvers as seen by the large span of results in the blind evaluation study. Two examples of LES of wind flow over Bolund Hill are Diebold *et al.* (2013) and Conan *et al.* (2016). Both Diebold *et al.* (2013) and Conan *et al.* (2016) found that the flow away from the surface was not significantly affected by errors near the surface.

We use a computational domain of 1533.0 m $\times$ 765.0 m $\times$ 127.0 m in the streamwise, horizontal spanwise and vertical directions to simulate Bolund Hill. The surrounding flat

surface at sea level is immersed at 5.25 m from the bottom of the domain. The RANS-LES blending height, h_B, was set to 2.5 m to maintain $h_B(2\Delta)^{-1} > 1$ as recommended by Senocak *et al.* (2007). The aerodynamic roughness length, z_0, was set to 0.006 m. Note, this is not the recommended value in Bechmann *et al.* (2011) nor is it truly representative of the actual Bolund Hill as two roughness lengths should be used, one for the surrounding water and another for the short-grass surface of the hill. However, we are only studying the effect the perturbation box inflow technique has on speedup ratio, so only having a single roughness should still be indicative of the sensitivity to perturbation box size. The boundary conditions for the periodic reference and the perturbation box method are mostly identical to the Askervein Hill simulation. The exceptions are that the shift for the periodic boundary conditions (Munters *et al.*, 2016a) was set to 150 m and the incoming mass flow rate was approximately 1.26×10^6 kg s^{-1} for the periodic reference case. The perturbation box sizes are given in Tab. 8.2. Case Bol1 is based on the inner layer height of the Bolund Hill measurement campaign, reported as ≈ 2 m in Bechmann *et al.* (2011). Case Bol2 follows the Obukhov length recommendation found in §8.4. As reported in Bechmann *et al.* (2011), one of the criteria to determine if measurements contributed to the neutrally-stratified database was an Obukhov length greater than 250 m. Let's say the Obukhov length is 300 m, then the one-eighth recommendation would result in three perturbation boxes in the vertical given the vertical height of the domain. Thus, we only pursue the one-sixteenth recommendation, giving an $H_{PB} = 17$ m. Again, Ri_{PB} is set to 0.042 and air is once again used at 293 K.

We chose to compare with Line B for the 270-degree wind direction as described in Berg *et al.* (2011) and shown in Fig. 8.7. Figure 8.8 shows the development of turbulence from buoyancy perturbations generated by the two perturbation box sizes. The largest of the perturbation boxes is similar in size to the smallest perturbation box in our Askervein trials in §8.4. The stochastic buoyancy perturbations developed into turbulence within an approximate 1000 m fetch. Given that the development region before the hill is less than 1000 m, it is not surprising that the flow has not reached a developed state. Turbulence

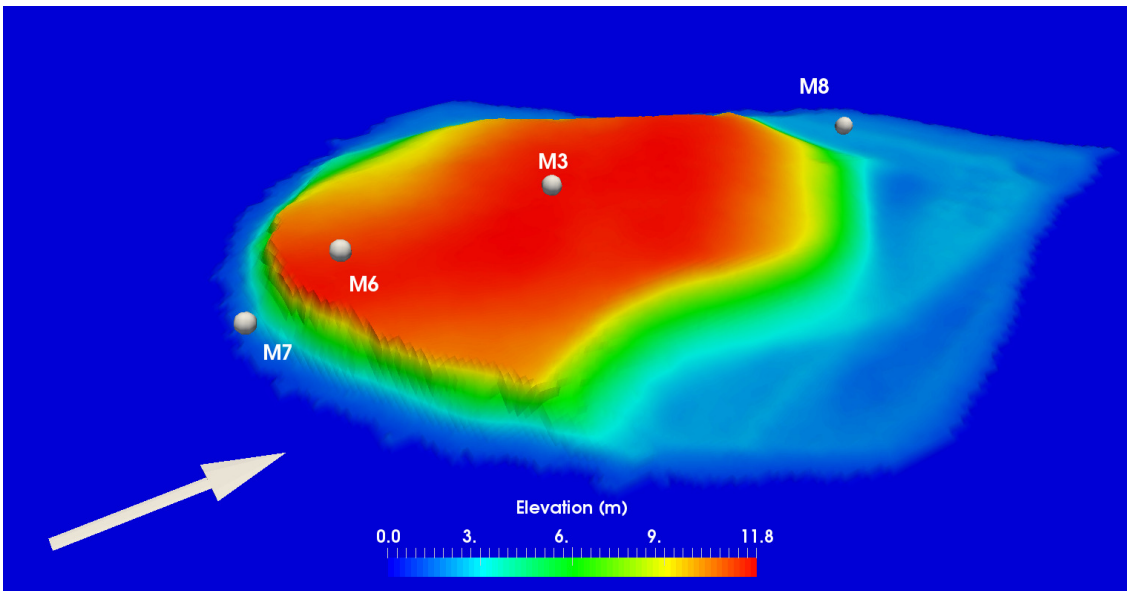
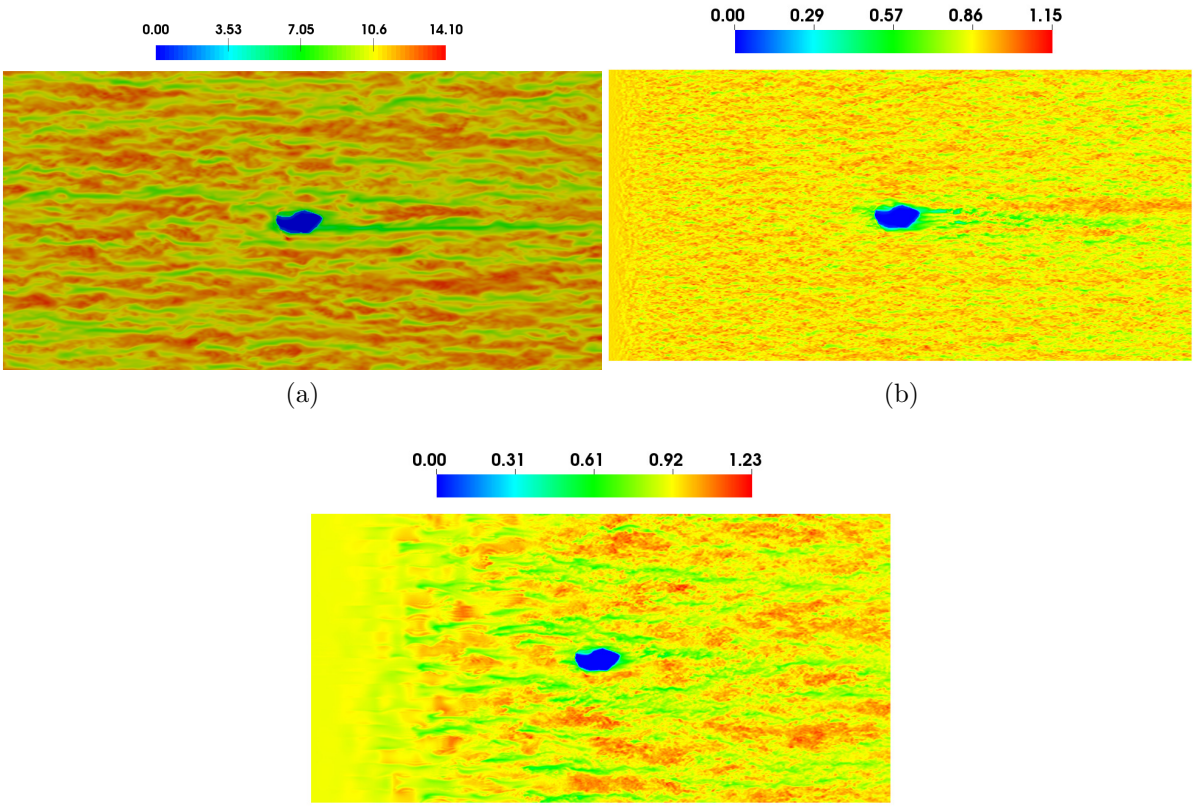


Figure 8.7: Bolund Hill elevation map. The measurement locations, represented by the spheres, are those used in the field measurement campaign.

is eventually developed, as seen in Fig. 8.8c. The smaller perturbation box breaks down rapidly into fine-scale turbulent eddies, visualized in Fig. 8.8b.

Figure 8.9 shows the fractional speedup ratios at the four masts, M7, M6, M3, and M8, shown in Fig. 8.7. Comparison with the reference periodic simulation shows very good collapse among the periodic simulation and the two perturbation box sizes. The notable differences are very close to the surface. A similar trend observed in Askervein is the larger perturbation boxes create a higher acceleration when the wind flows over the hill. Even at M6 that lies directly behind the escarpment that creates a challenging case for LES, there is not significant sensitivity between the two perturbation box sizes. The smaller perturbation box generates a lower wind speed than the larger box at M7 close to the surface. However, this helps with the flow at M8. Overall, the periodic simulation and the turbulent inflow technique agree well with the field data.



(a) (b)

(c)

Figure 8.8: Visualization of the different perturbation box methods used for the Bolund Hill case. Horizontal wind speed given in units of m s^{-1} and is taken at approximately 10 m above the surrounding flat surface. (a) Periodic (b) Case Bol1, $H_{PB} = 2$ (c) Case Bol2, $H_{PB} = 17$ (see Tab. 8.2)

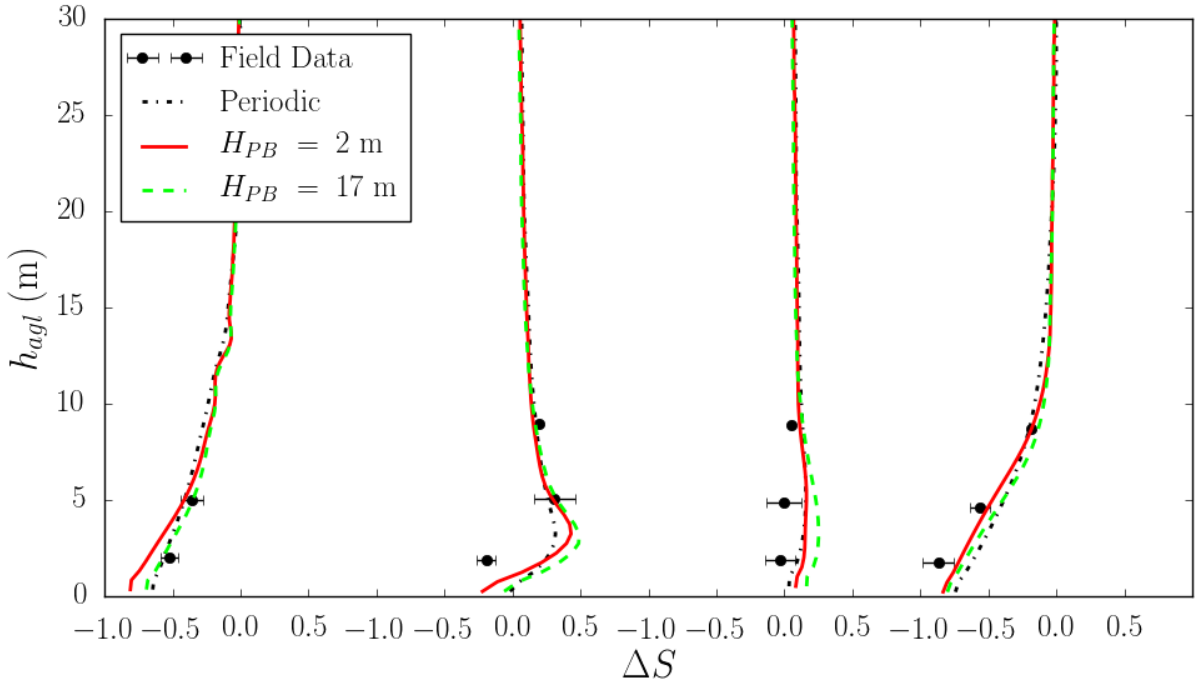


Figure 8.9: Fractional speedup ratio along line B for the 270-degree wind direction in the Bolund Hill field study. Masts M7, M6, M3, and M8 respectively as described in Berg *et al.* (2011).

8.6 Summary and Conclusions

We have presented a perturbation box turbulent inflow generation technique, originally proposed for general engineering flows, applied to complex terrain benchmarks. The perturbation box technique creates colored-noise buoyancy perturbations by introducing stochastic temperature perturbations as a source term in a filtered temperature transport equation. The amplitude of the temperature perturbations is determined by a bulk Richardson number based on the incoming mean velocity and the perturbation box height leading to temperature perturbation amplitudes that vary with distance from the surface. The technique retains the advantage of accounting for time-varying mean inflow directions while requiring minimal knowledge of the turbulent flow correlations at the inflow region.

The perturbation box height was determined in two ways during this study: as a fraction of the Obukhov length and as the height of the inner layer. The smaller perturbation boxes developed into turbulence much more rapidly than their larger counterparts. However, the fine-scale structures reduced the acceleration of wind flowing over a hill. Larger turbulent eddies are required to obtain the necessary acceleration over the hills. Generating only eddies of the finest scales results in low fractional speedup ratios.

However, the larger perturbation boxes that are capable of generating the larger flow motions require a much longer development fetch to develop the stochastic buoyancy perturbations into realistic turbulence. While the vortices are not realistic turbulence, they do not have a significant impact on the acceleration of mean velocity over a hill. Second-order moments would definitely experience a greater impact but such a study would require a turbulence model suitable for flow separation.

CHAPTER 9

Dynamic Rating of Overhead Transmission Lines over Complex Terrain Using a Large-eddy Simulation Paradigm

Phillips, T., DeLeon, R., and Senocak, I. "Dynamic rating of overhead transmission lines over complex terrain using a large-eddy simulation paradigm." *Renewable Energy*, 108: (2017) pp. 380-389.

9.1 Abstract

Dynamic Line Rating (DLR) enables rating of power line conductors using real-time weather conditions. Conductors are typically operated based on a conservative static rating that assumes worst case weather conditions to avoid line sagging to unsafe levels. Static ratings can cause unnecessary congestion on transmission lines. To address this potential issue, a simulation-based dynamic line rating approach is applied to an area with moderately complex terrain. A micro-scale wind solver — accelerated on multiple graphics processing units (GPUs) — is deployed to compute wind speed and direction in the vicinity of powerlines. The wind solver adopts the large-eddy simulation technique and the immersed boundary method with fine spatial resolutions to improve the accuracy of wind field predictions. Statistical analysis of simulated winds compare favorably against wind data collected at multiple weather stations across the testbed area. The simulation data is then used to compute excess transmission capacity that may not be utilized because of a static rating practice. Our results show that the present multi-GPU accelerated simulation-based approach — supported with transient calculation of conductor temperature with high-order schemes — could be used as a non-intrusive smart-grid technology to increase transmission capacity on existing lines.

9.2 Introduction

Investments in renewable energy has been driven by several factors, including energy security and stability, climate change, and economics. Since 2000, wind energy has been the largest source of new renewable generation installed in the United States (win, 2015). However, wind power generation is much more complex than installing wind turbines in windy areas. Grid integration is a major challenge, many of the best locations for wind farms do not have access to the needed transmission capacity (of Energy, 2008). Congestion in existing transmission lines is a growing concern, resulting in inefficiencies for both renewable energy producers, utilities and balancing authorities (Arroyo *et al.*, 2015). At times, transmission service providers (TSPs) may not be able to absorb the power generated, therefore, power production can be curtailed.

Potential sites for wind power generation are usually found in remote open areas that are away from populated cities, where electricity is needed most. Historically, transmission systems have been built together with power production installations in order to meet the electricity demand. For economic reasons they are usually not over-sized, therefore, current transmission networks in many of these sites may not support additional generation. Many wind projects have been able to patch into the existing transmission network, however, these opportunities are shrinking. Further expansion of wind energy may require large investments in transmission networks, creating an obstacle for cost-effective wind deployment (win, 2015; Metevosyan, 2005).

Transmission capacity can be increased in several ways. The obvious way is to reinforce the transmission network with new powerlines. However, this is constrained by the high costs and legal challenges of building new powerlines (Jorge and Hertwich, 2013). Therefore, TSPs have focused on innovative solutions that modifies existing network to increase transmission capacity. Different techniques include prediction of meteorological conditions by means of deterministic (Hall and Deb, 1988) or probabilistic (Reding, 1994) forecasting methods, and

adopting the newest innovations in smart-grid real-time monitoring of temperature, sag, tilt, power, current and weather conditions (Pytlak *et al.*, 2011; Cho *et al.*, 2014; Holyk *et al.*, 2014). In the case of wind energy integration, monitoring meteorological conditions in real-time can be very beneficial for both power generation and transmission purposes. Strong winds needed for wind generation, will also cool down the conductor of local transmission lines, creating additional capacity, which would enable TSPs to "overload" the line when it is needed most (Hosek, 2011; Heckenbergerova and Hosek, 2012).

Transmission conductor capacity is limited by its maximum allowable temperature. The maximum amount of electric current a conductor can transmit before structural damage is known as *ampacity*. Currently, ampacity is generally determined using a static line rating (SLR) methodology. SLR is based on conservative assumptions regarding environmental conditions, such as high ambient temperature and low wind conditions. These assumptions were made to avoid lines sagging to unsafe levels. However, they are overly conservative for areas where wind generation is abundant. Therefore, TSPs are investigating dynamic line rating (DLR) methods to increase ampacity on existing lines. DLR utilizes real-time environmental conditions to better predict the temperature of the conductor. Deployment of DLR has the potential to reduce the estimated $60 billion needed in transmission infrastructure to meet the 20% wind energy by 2030 (of Energy, 2008).

Fernandez *et al.* (2016) provide a comprehensive review of real-time DLR technologies that have been developed over the last 30 years, endorsing the potential of DLR for wind power integration. Commercially available DLR technologies include direct line sag, line tension, and conductor temperature measurements (Uski-Joutsenvuo *et al.*, 2012). Wind turbines are increasingly being built in areas of complex terrain, as available sites on flat terrain is diminishing. In complex terrain elevated positions like hill tops are favorable sites due to the increased wind speed. However, complex terrain proves to be challenging for the aforementioned DLR systems. Sag and tension monitoring systems can only inform TSPs of the average sag or tension measurement over large subsectionalized transmission

spans, therefore, only the average temperature of the conductor over large subsections can be known. Direct temperature measurements at a single location may not necessarily represent the critical span, or the hottest subsection along a conductor. Studies have shown that conductor temperature can vary spatially by 10–20°C due to variations in wind speed and direction (EPRI, 2005; Seppa, 1993; Seppa *et al.*, 2010). Therefore, currently adopted DLR systems may not be a good solution for determining the real-time transmission capacity in regions of complex terrain. If implemented, they may potentially lead to severe overestimation of the actual ratings, allowing the conductor to be overloaded and causing degradation of the line. Adding more monitoring devices could be a solution, however these systems are typically expensive for wide deployment that is needed to reduce risks to an acceptable level (Gentle *et al.*, 2012. INL/CON-12-27012). Additionally, implementation of direct DLR systems can prove to be challenging, as transmission lines need to be de-energized during installation and regular maintenance. Therefore, a non-intrusive DLR solution is highly desirable, which also motivates the present study.

In Greenwood *et al.* (2014) two non-intrusive approaches were compared. One approach adopted a CFD-based library approach to extract wind speeds and direction along the path of transmission lines and the other approach used an uncertainty model based on a small number of weather stations. Greenwood et al. suggested that a more sophisticated wind model that can accurately capture the time-dependent nature of winds over complex terrain coupled with uncertainty quantification would be invaluable to expand the DLR concept. Michiorri *et al.* (2010) used actual environmental conditions from a limited number of meteorological stations as input to the steady-state thermal models. An inverse distance interpolation technique and a power law for wind profile were used to estimate the environmental conditions at transmission line. A state-estimation algoritmh based on the Monte-Carlo approach was then used to take into account the uncertainty in data. Michiorri et al identified the source of errors as the physical models used in their approach, and suggested the use of wind flow models based on the computational fluid dynamics (CFD) approach.

With today's improved wind and weather modeling and high performance computing capabilities, the use of computer simulations to forecast wind and determine transmission capacity has emerged as an alternative to intrusive hardware solutions. Short-term wind forecasting can potentially be a valuable tool for TSPs, enabling conductor temperature calculations at dense intervals along transmission lines in complex terrain. Michiorri *et al.* (2015) reviewed current meteorological forecasting technologies for broadening the adoption of DLR and particularly drew attention to the current need to improve low wind speed modeling and turbulence. Michiorri et al. promote the viewpoint of moving from monitoring technologies to an active management technology where wind forecasting for different time horizons becomes critical. To this end, our large-eddy simulation approach directly addresses the need to improve low wind speed modeling in the vicinity of transmission lines.

Meso-scale numerical weather prediction models have long been used to forecast winds and other meteorological variables, however their application to micro-scale atmospheric boundary layer flows over complex terrain with a horizontal spatial resolution ranging from 10 to 100m is still an on-going research and far from realizing the forecasting mode. Mesoscale weather forecasting models typically adopt spatial resolutions on the order of a few kilometers. Results from existing foresting models vary greatly depending on the locations and time period investigated (David Carvalho and Santos, 2014, 2012; Bhaskar *et al.*, 2010; Negnevitsky *et al.*, 2009; Soman *et al.*, 2010). On relatively flat terrain use of mesoscale models may prove effective, but fine-scale forecasting solutions that can resolve complex terrain features with horizontal resolution on the order of 10m are needed. For instance micro-scale complex terrain forecasting models could be used to quantify the stochastic variations in line ratings, which could then be converted to dynamic constraints as described by Banerjee *et al.* (2015).

In what follows, we present the equations for dynamic line rating, followed by our massively parallel micro-scale wind solver to predict wind speed and direction as a function of time. An actual test area with moderately complex terrain is simulated, and predictions are compared against available weather station data at multiple locations. Field and simula-

tion data are then used to compute available ampacity for a dynamic line rating scenario, demonstrating the potential of the current non-intrusive approach to increase transmission capacity.

9.3 IEEE Standard 738-2012 Transmission Capacity Calculation

Transmission line capacity is commonly calculated using procedures described either in the Institute of Electrical and Electronics Engineers (IEEE) 738 Standard (IEEE, 2012) or the CIGRE Standard (CIGRE, 1992). In this study, we follow the IEEE standard and describe the salient features of the calculation procedure for clarity.

Temperature of an overhead electrical conductor is a function of its material properties, weather conditions, and electrical current. The steady-state heat balance is given as

$$q_c + q_r = q_s + q_j, \tag{9.1}$$

where q_c, q_r, q_s, and q_j are the conductor convective heat loss, radiated heat loss, solar heat gain, and Joule heating, respectively.

Joule heating is calculated using the electric current, I, and conductor resistance, $R(T_{ave})$, which is a function of its average temperature, T_{ave}. Joule heating is given as

$$q_j = I^2 \cdot R(T_{ave}). \tag{9.2}$$

The steady-state thermal rating used to calculate conductor capacity is then expressed as

$$I = \sqrt{\frac{q_c + q_r - q_s}{R(T_{ave})}}, \tag{9.3}$$

where resistance is determined at the maximum permissible conductor temperature from lookup tables. It is common practice to use this equation under conservative assumptions for weather conditions, especially for convective heat loss, to rate transmission lines. This

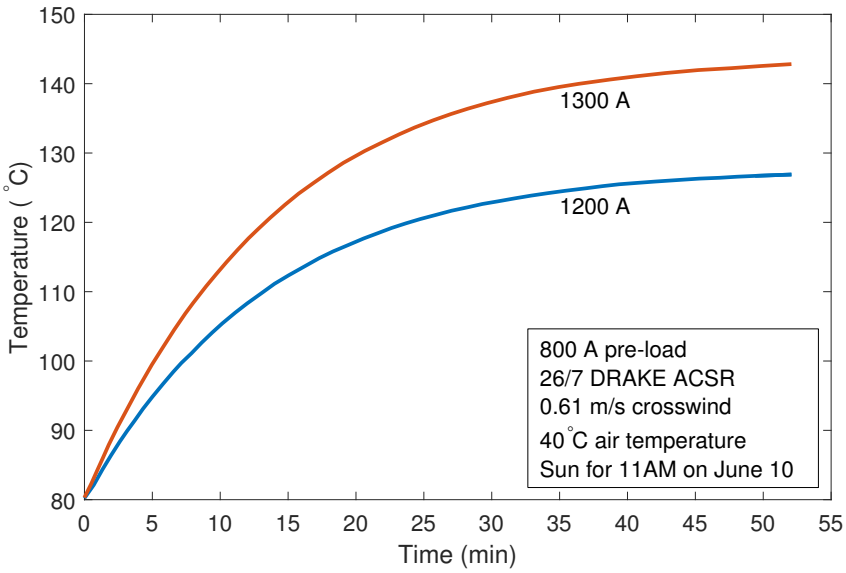


Figure 9.1: Transient temperature response to a step change in current from 800 to 1200/1300 Amps. Graph adapted from IEEE (2012).

practice, known as the *static line rating*, often leads to stringent limits, not enabling the real-time capacity of the line to be utilized.

9.3.1 Dynamic Ratings

The steady-state rating given in Eq. 9.3, is calculated using conservative estimates of weather conditions. CIGRE (CIGRE Working Group B2.12, 2006) recommends that base ratings should be calculated with an effective wind speed of 0.6 m/s, an air temperature near the seasonal maximum (40°C summer) and a solar radiation of 1,000 W/m^2. In reality the electrical current through the conductor and real-time weather conditions exposed to the line are constantly changing. In response to these changes, conductor temperature varies with an associated time scale. Since the temperature of the conductor is what limits its capacity, we want to track its temperature in real-time. The change in temperature from an increase in current from 800 to 1,200 and 1,300 Amps is shown by the digitized data (IEEE, 2012) in Fig. 9.1.

Transient response of a conductor's temperature to changing current and weather condi-

tions can be modeled as a first-order ordinary differential equation (ODE) expressed as

$$\frac{dT_{ave}}{dt} = \frac{1}{mC_p}\left[q_j + q_s - q_c - q_r\right], \tag{9.4}$$

where mC_p is the total heat capacity of the conductor, given as

$$mC_p = \sum m_i C_{pi}, \tag{9.5}$$

where m_i and C_{pi} are the mass per unit length of i^{th} conductor material and the specific heat of i^{th} conductor material, respectively. Therefore, if the electrical current and real-time conditions are known, the ODE can be solved numerically to calculate real-time temperature of the conductor. With the use of a wind forecasting model, conductor temperature can not only be potentially forecast, but it can be done at very dense intervals, which may not be feasible with current hardware solutions. This would give TSPs an unprecedented understanding of the current and future state of the transmission lines, allowing for better efficiency of the transmission and generation network.

The ODE given in Eq. 9.4 represents an initial value problem (IVP). The general form is expressed as

$$\frac{dy}{dt} = f(t, y) \tag{9.6}$$

over a time interval

$$a \leq t \leq b \tag{9.7}$$

subject to an initial condition

$$y(a) = y_0. \tag{9.8}$$

The IEEE Standard 738-2012 does not give a recommended numerical method to solve the ODE given in Eq. 9.4. However, it does supply a sample computer code as a convenience to the user. In that sample code, a first-order accurate forward Euler method is used. In the

standard, it is also pointed out that other numerical methods may well be more appropriate in certain situations. Additionally, it is noted that time step size be kept small to reduce numerical errors.

We believe a forward Euler method is too crude for a critical system such as transmission lines. Therefore we examine the use of a fourth-order accurate Runge-Kutta (RK4) scheme (Chapra and Canale, 2010) for improved accuracy and computation time. The IEEE standard states that there seems to be little advantage in using a time step greater than one second. This may be true when doing a single transient temperature calculation for demonstration purposes, as done in the IEEE standard. However, we are interested in implementing a real-time dynamic rating in practice, which will likely require many thousands of these calculations to be performed along the length of transmission lines. Therefore, computational expense may become an issue when using a forward Euler method with small time steps. An RK4 scheme allow us to assume larger time step sizes while keeping the error low.

An RK scheme can be written as

$$y_{i+1} = y_i + \phi(t_i, y_i, h) \cdot h, \tag{9.9}$$

where $\phi(t_i, y_i, h)$ is called the increment function, which is a representative slope over the interval h. The following 4^{th} order RK scheme (RK4) is used in this study.

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \cdot h, \tag{9.10}$$

where k's represent slope estimates.

As a test case to compare both numerical methods, we have performed the same 800 to 1,200 step increase in current provided in the IEEE 738 Standard, shown in Fig. 9.1. A normalized L2-norm is used to quantify the difference between the two methods. The

Table 9.1: Normalized L2-norm of conductor temperature using a forward Euler and 4^{th} order Runge-Kutta method. The "exact" values are calculated using the RK4 and a time step (dt) of 0.01s. The speedup is based of the Euler calculation with a time step of 1s.

	L2-norm		Speedup	
dt(s)	RK4	Euler	RK4	Euler
1	1.8E-14	1.1E-4	0.3	1
10	9.4E-11	3.4E-3	3.0	10
30	1.3E-8	1.8E-2	9.2	31
60	3.1E-7	5.0E-2	19	61
300	5.1E-4	0.58	93	314
600	1.4E-2	1.8	186	616
1,200	0.52	7.5	367	1,266

normalized L2-norm is given by

$$\| x \| = \frac{1}{N}\sqrt{x_i^2}, \tag{9.11}$$

where N is the number of comparisons between the exact and numerical solution and x_i is the difference between them. There is no analytical solution, therefore, a reference value was used as the exact solution. The exact value was calculated using the RK4 method and a time step of 0.01 seconds.

The results are shown in Fig. 9.2, and tabulated in Table 1. This test case makes it clear that care needs to be taken with the selection of a numerical method, the resulting conductor temperature and computation time can be greatly affected. If a DLR system is put in place it is critically important that temperature computations can be completed in near real-time while keeping numerical errors to an acceptable level. Using the RK4 method allows a time step size of 300s over the Euler of 1s, while keeping numerical errors at the same order of magnitude. This allows calculations to be completed over 90 times faster, potentially reducing computation time from minutes to seconds. This time could prove critical for TSP, giving them additional time to make needed transmission decisions. Therefore, we recommend an RK4 scheme for calculating the temperature of a conductor as it is easy to implement and there is a clear benefit to it.

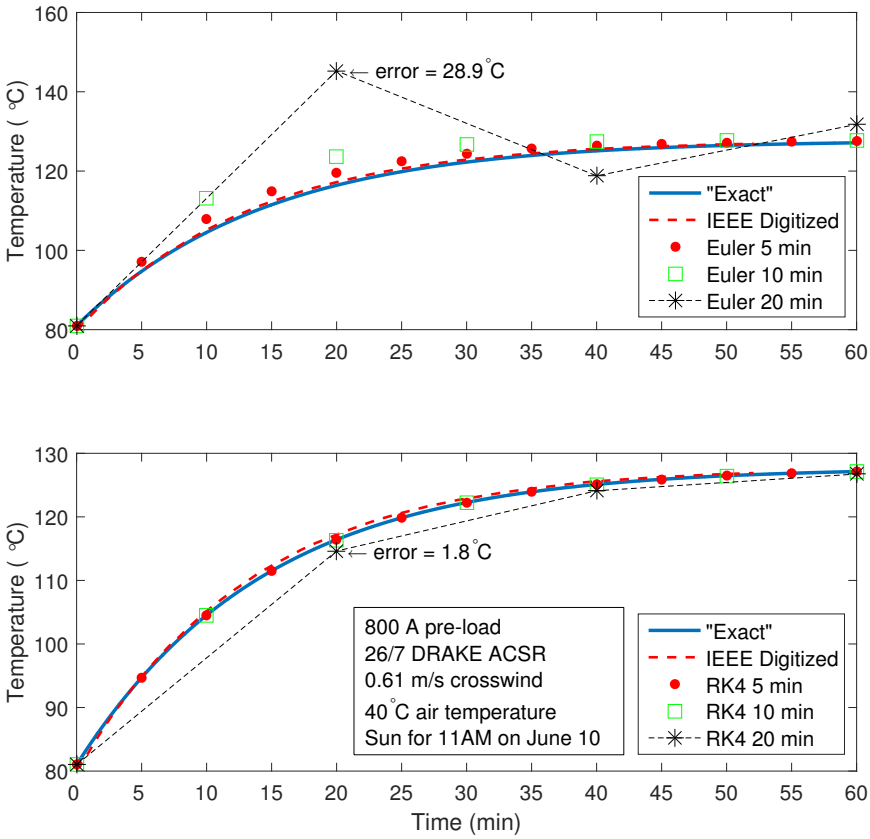


Figure 9.2: Transient conductor temperature solution using a forward Euler method (top) and a 4^{th} order Runge-Kutta method (bottom) with time steps of 5, 10, and 20 minutes. The "Exact" value was calculated using the RK4 and a time step of 0.01s. IEEE standard solution has been digitized.

9.4 Massively Parallel Wind Solver

The need for accurate wind modeling , especially at low speeds and over complex terrain, were mentioned in recent studies (Michiorri *et al.*, 2015; Fernandez *et al.*, 2016). Steady-state CFD solutions based on Reynolds-averaged Navier-Stokes (RANS) equations may not capture the unsteady nature of winds over complex terrain. The large-eddy simulation technique (LES) is inherently unsteady and generally produces better results for separated flows over complex terrain. However, LES is expensive in terms of computational resources, because fine spatial resolutions are needed to resolve energetic eddies. On the other hand, fine resolutions could be beneficial to better monitor the conductor temperature along its path. The unsteady nature of the wind simulations could also help capture the transient response of the conductor to establish a reliable line rating technique. To this end, advances in parallel computing technology can help broaden the adoption of LES technique in practical problems. Graphics processing units offer a relatively economical solution as a small-footprint computing platform because of their massively parallel architecture.

In this study, we adopt a multi-graphics-processing-unit-accelerated (multi-GPU), parallel wind solver, GIN3D (Thibault and Senocak, 2012; Jacobsen and Senocak, 2013, 2011; DeLeon *et al.*, 2012, 2013), as an improved solution for wind modeling over complex terrain. Depending on the mesh size, GIN3D has the potential compute winds over arbitrarily complex terrain faster than real-time. Computational domain size can range from meters to several kilometers. The computations are accelerated on GPU clusters with a dual-level parallel implementation that interleaves Message Passing Interface (MPI) with NVIDIA's Compute Unified Device Architecture (CUDA). For instance for an area of approximately 6.5km by 5.7km with a spatial resolution of 15m in the horizontal and 8m in the vertical, simulations can be 2.2. times faster than real-time on four Tesla K20 GPUs. In this study, we will execute GIN3D by imposing a wind direction inferred from local measurements to assess potential of a simulation-based DLR approach. Our future goal is to forecast micro-scale

atmospheric flows over complex terrain with a model-chain approach where lateral boundary conditions are informed by a mesoscale weather forecasting model.

The large-eddy simulation (LES) technique is used in GIN3D for subgrid-scale turbulence closure. In LES of atmospheric flows, it is common practice to employ a wall-model due to the complexity and roughness of terrain and the inadequate resolution in the vicinity of the surface. In particular we pursue a hybrid Reynolds-averaged Navier-Stokes (RANS) LES technique. We employ the hybrid eddy viscosity model proposed in Senocak *et al.* (2007) which can be written as follows,

$$\nu_t = \left[\left(\left[1 - \exp\left(-z/h_{RL}\right)\right] C_S\Delta\right)^2 + \left(\exp\left(-z/h_{RL}\right)\kappa z\right)^2\right] |\overline{S}|, \tag{9.12}$$

where z is the surface-normal distance, h_{RL} is the RANS-LES transition height, $C_S\Delta$ representing the sub-grid-scale (SGS) mixing length (C_S being the model coefficient and Δ the LES filter width), and κz representing the RANS mixing length. The SGS mixing length is determined using the Lagrangian dynamic SGS methodology (Meneveau *et al.*, 1996) applied to the Smagorinsky eddy viscosity model. The Lagrangian dynamic model is a localized SGS model that does not require any homogeneous directions in the computational domain. Therefore, it is adequate for arbitrarily complex terrain. The RANS mixing length is that of Prandtl (1925).

We prefer a Cartesian method to solve the governing equations as it maps well to the computer architecture of modern GPUs. The immersed boundary (IB) method is used to impose boundary conditions on the surface using logarithmic reconstructions (Senocak *et al.*, 2004) in conjunction with the above hybrid eddy viscosity model. Note that the goal is to produce the correct Reynolds stresses at the surface. Therefore, it is important that the velocity reconstruction scheme is consistent with eddy viscosity near the surface. A logarithmic reconstruction therefore is suitable because it is consistent with the Prandtl's mixing length model near the surface.

While IB methods eliminate cumbersome meshing and poor mesh quality (e.g. skewed cells), the challenge is to impose the boundary conditions as the immersed surface will most likely not coincide with the Cartesian grid points. We employ the direct-forcing approach proposed by Mohd-Yusof (1997) and later applied by Verzicco *et al.* (2000). This IB method can be classified as a "sharp interface" IB method, as the boundary condition at the surface appears explicitly in the method. The first step of this IB method is to identify the Cartesian grid cells cut by the surface, which can be challenging with arbitrarily complex terrain. The details of the geometric pre-processing can be found in Senocak *et al.* (2015). Once the geometric information is known, the values in near-surface grid cells cut by the immersed surface can be reconstructed each simulation time step by interpolating between the known boundary condition at the immersed surface, e.g. the no-slip condition for velocity, and resolved values from the flow field where the grid cells are not cut by the immersed surface. The logarithmic reconstruction scheme for velocity proposed by Senocak *et al.* (2004) is revised to explicitly enforce the impermeability condition over complex terrain. First, the velocity components are projected onto surface-parallel and surface-normal vectors, $u_{i,t}$ and $u_{i,n}$. The reconstruction scheme for the normal components is a linear interpolation between the flow at a sufficient surface-normal distance, z_2, and the no-slip condition at the immersed surface,

$$u_{i,n}|_{z_1} = u_{i,n}|_{z_2} \frac{z_1}{z_2}, \tag{9.13}$$

where z_1 is the IB node wall-normal distance. The impermeability condition is then explicitly enforced. The tangential reconstruction scheme is based on logarithmic-similarity in the atmospheric surface layer (Stull, 1988) and is given by

$$u_{i,t}|_{z_1} = u_{i,t}|_{z_2} \frac{log(z_1/z_0)}{log(z_2/z_0)}, \tag{9.14}$$

using the same surface-normal distances as in Eq. 9.13, where z_0 is the aerodynamic roughness length.

Table 9.2: Simulation parameters. Target domain is centered in the total domain which includes the extension and tapering regions for the periodic boundary conditions.

Domain size (km)			Grid Points			Resolution (m)		
L_x	L_y	L_z	N_x	N_y	N_z	Δx	Δy	Δz
16.0	23.0	1.94	1025	1025	513	29.3	34.2	3.9

9.4.1 Simulation Setup

The target computational domain is $\sim$368km^2 shown in Fig. 9.3. Periodic boundary conditions were applied in the lateral directions, deemed suitable as the elevation changes relative to the total height of the computational domain are small. As complex terrain may not be the same elevation on all sides of the domain, we extended and tapered the target domain down such that the elevation is constant along the perimeter of the domain. This added approximately 6-7km to each side. The total domain height is $\sim$2km from the lowest elevation. The Cartesian grid consisted of $\sim$539 million points, giving lateral resolution of $\sim$30m and vertical resolution of 4m. Simulation parameters are given in Table 9.2.

The wind flow is driven by a constant 6.0e-05m/s^2 pressure gradient coming from the north-east at an angle of 63.3° using meteorological conventions (i.e. wind coming from north is 0° and clock-wise is positive.). The pressure gradient was adjusted iteratively to approximately match the observed wind speed at a weather station over flat terrain. The top of the domain is set to a free-slip condition. Fluid properties are that of air at standard temperature. Surface roughness, z_0, is set to 0.15m, a value suggested in (Stull, 1988) for rural farmland areas. Following Senocak *et al.* (2007), the RANS-LES interface, h_{RL}, is set to 31.6m, twice the size of the LES filter width, $\Delta = \sqrt[3]{\Delta x \Delta y \Delta z}$. The flow is initialized by superimposing high-amplitude, low-frequency sinusoidal perturbations onto a rough-surface log-law profile. This was a necessary step as the terrain elevation changes were not enough to trip turbulence unassisted, a further indication that periodic boundary conditions are suitable for this case. The flow was allowed to develop for two hours of simulated time

before reaching a stationary state. The wind solver assumes incompressible flow, solving the Poisson equation with geometric multigrid designed for multi-GPUs (Jacobsen and Senocak, 2011) and uses second-order central difference schemes for spatial derivatives and a second-order Adams-Bashforth scheme for time integration.

9.5 DLR Test Area

Idaho Power Company (IPCo) and Idaho National Laboratory joint test bed area for DLR research is located on the Snake River Plain in southern Idaho. The test site lies in an area of high desert with complex terrain, covering an area approximately 1,500km^2 with an elevation range of 754m to 1,1198m.

Seventeen weather stations were mounted by IPCo/INL team at a height of 10m agl in strategic locations along more than 190km of high-voltage transmission lines. Data collection through a cellular network has been underway by IPCo since August of 2010. The measured quantities are wind speed, wind direction, ambient temperature, and solar irradiation. Data from the weather stations is collected every 3 minutes, it is an average of 2s readings over the 3-minute time interval. Weather stations use NRG 40C (NRG, 2015) or the APRS #40R (APR, 2015) three cup anemometers. Both models have similar specifications; wind speed accuracy of 0.1 m/s with a sensor range of 1–96m/s. In Phillips *et al.* (2014) a year-long weather data was analyzed seasonally to demonstrate the limitation of the static rating approach on ampacity.

For the simulations used in this paper, we chose a 16km×23km area with an elevation change of over 330m. Figure 9.3 shows the elevation map and locations of the nine weather stations located in this area.

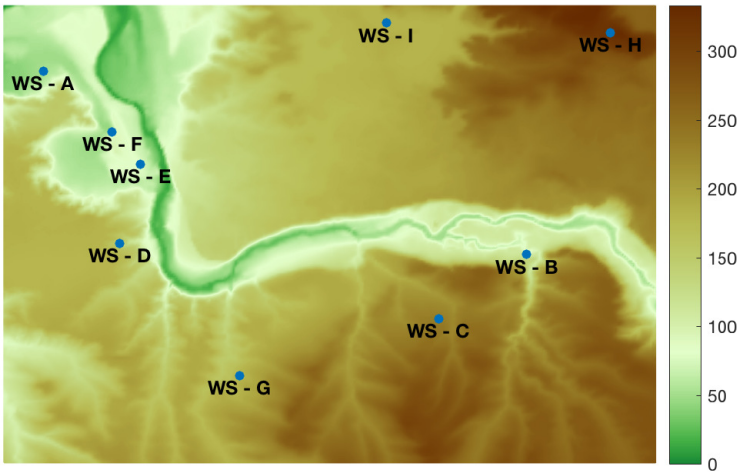


Figure 9.3: Section of INL/IPCo test site for DLR research, colored by terrain height.

9.5.1 Test Area Prevailing Winds

Wind flow patterns emerge from horizontal surface and atmospheric temperature contrasts on all spatial scales, from global to local size (Emeis, 2013). Both local and global systems exhibit large regularity of daily and seasonal wind and weather cycles (Deb, 2000). This regularity can be largely attributed to the local terrain and surface properties. Using year-long data starting July 1, 2012 the prevailing wind direction is illustrated by the wind rose in Fig. 9.4. Two weather stations ∼2km east of the area investigated were selected because they better represent the boundary conditions of the simulation, therefore used as discussed in Section 9.4.1.

Because weather stations operate unattended for a long period and adverse weather conditions can exist during winter months, it was necessary to validate the collected data against a common statistical distribution. The distribution of wind speed is commonly defined using the Weibull probability density function (Brower, 2012). During any time interval the two parameter wind speed probability is given as

$$f(v) = \left(\frac{k}{\lambda}\right)\left(\frac{v}{\lambda}\right)^{k-1} e^{-\left(\frac{v}{\lambda}\right)^{k}} \tag{9.15}$$

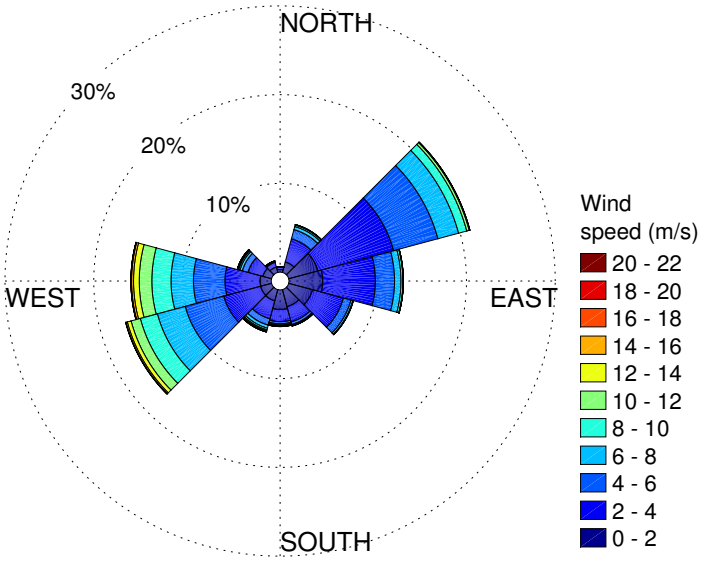


Figure 9.4: Wind rose of year-long wind data starting July 1, 2012 from two weather ∼2km east of the area investigated.

where v is the wind speed, k is the shape parameter, and λ is the scale factor, which is expected to be close to the mean speed. The Weibull probability density function of year-long measured wind data at each of the weather stations is shown in Fig. 9.5. The nondimensional shape parameter for the collected data is in agreement with the commonly observed values (i.e. k ranging from 1.6 to 2.4) (Brower, 2012).

9.6 Results

To demonstrate the feasibility of a simulation-based DLR approach, we first compare our wind solver predictions against field data. A horizontal slice of the eastern region of the target domain in Fig. 9.3 is the focus of Fig. 9.6. Eddy sizes vary visibly over the terrain. Long, streak-like structures with low wind speed are evident in the vicinity of the surface. The location of the canyon can be inferred as the flow in to and out of the canyon breaks down the larger eddies vertically above the canyon into much smaller ones. The wind breaks into smaller eddies as it blows over the canyons. Additionally, acceleration of the flow above

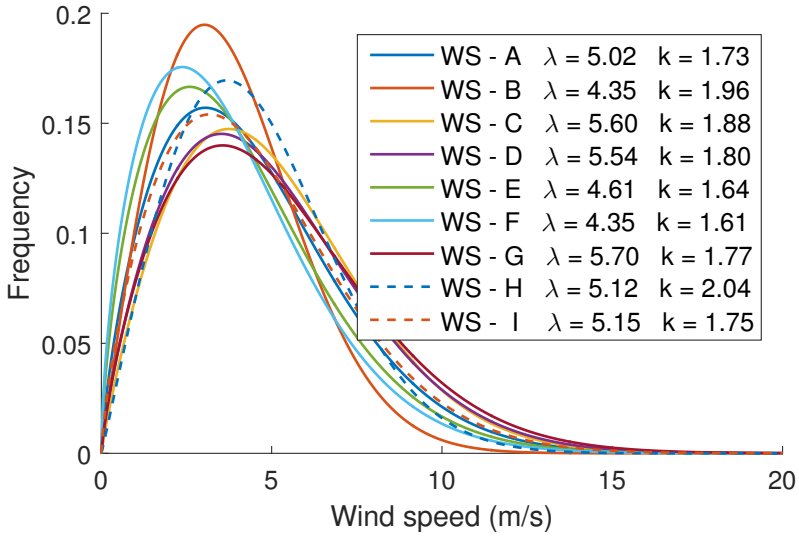


Figure 9.5: Weibull wind distribution using year-long wind data from each weather station.

the canyon can be observed from the color map. We next perform a statistical evaluation of the wind flow simulation.

9.6.1 Statistical Validation of the Wind Solver

To evaluate the wind solver's performance against anemometer data collected at select locations across the test bed area, we follow an approach similar to the one presented in David Carvalho and Santos (2012) by using five statistical parameters: the mean and standard deviation, the root mean squared error (RMSE), the bias, and the standard deviation of the error (STDE). The mean is given as

$$\overline{v} = \frac{1}{N}\sum_{i=1}^{N} v_i \tag{9.16}$$

where $\overline{v}$ is the mean speed, N is the number of data points, and v_i is the i^{th} wind speed of either the real-time data or simulation results. The standard deviation, S_v, is given as

$$S_v = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i - \overline{v})^2} \tag{9.17}$$

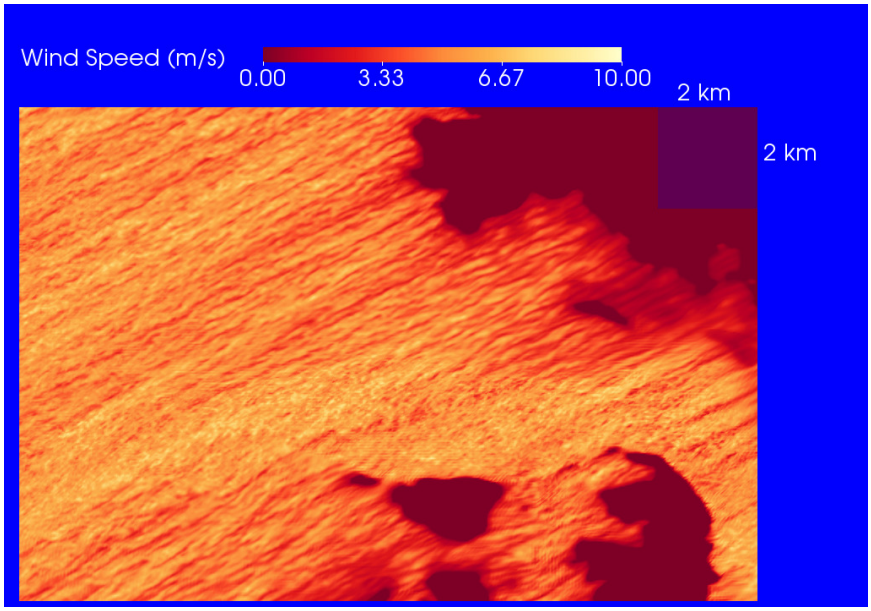


Figure 9.6: Flow visualization. Horizontal slice across domain focusing on eastern part of the canyon in the target domain. Flow is from upper-right moving to lower-left. 2km × 2km box provided to show scale.

and the RMSE is computed as

$$RMSE = \left[\frac{1}{N} \sum_{i=1}^{N} (v_i')^2 \right]^{1/2} \tag{9.18}$$

where N is the total number of deviations, v', between the the simulated wind speed, v^{sim}, and the respective observed wind speed at the weather station, v^{obs}. The deviation is given as

$$v' = v^{obs} - v^{sim} \tag{9.19}$$

The bias is defined as

$$Bias = \frac{1}{N} \sum_{i=1}^{N} v_i' \tag{9.20}$$

and makes possible the evaluation of the data systematic errors. A positive bias means that the simulations overestimate the measured values.

The standard deviation of the error (STDE), helps evaluate the dispersion of the error

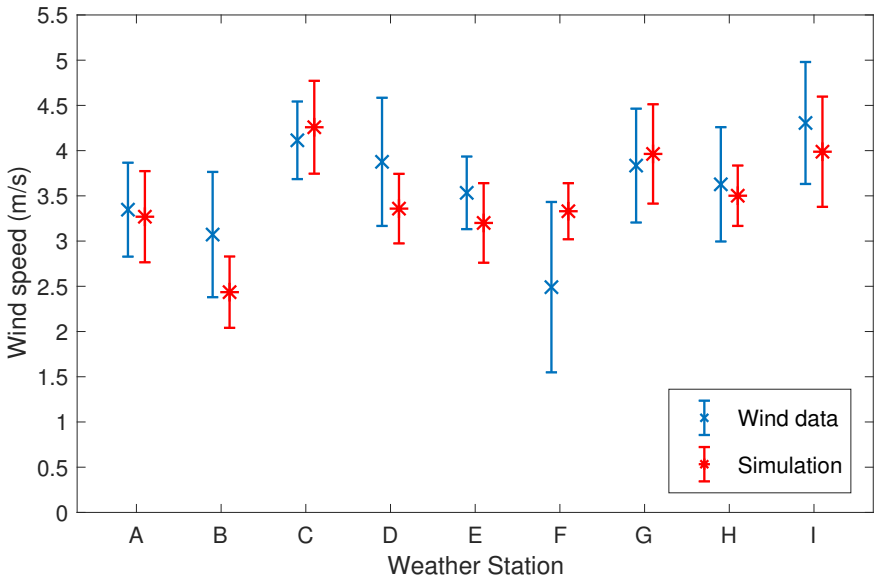


Figure 9.7: Mean and standard deviation of the field data and simulation results for wind speed

and it can be written as

$$STDE = \left[RMSE^2 - Bias^2\right]^{1/2}. \tag{9.21}$$

The STDE removes from the RMSE possible offsets (biases). A low STDE shows if a given error is mainly due to a kind of offset that can more easily be corrected because the underlying physics is correct, whereas a high STDE represents random error and hints unphysical results.

Figure 9.7 shows that the mean and standard deviation of the wind speed between the field data and simulation results. We observe that STDE is larger for weather stations B, D and F than the rest of the weather stations. We attribute this difference to the challenges of collecting seasonal data from weather stations that are unattended for long periods. Another issue is that these weather stations were placed to be close to the powerlines and not necessarily at locations that would capture the dominant wind patterns over the area. It is likely that these weather stations are picking up local details that may not be represented in the simulation.

The comparison between field and simulation data is further quantified in Table 3, with an average RMSE value of 0.863, bias of 0.101, and STDE of 0.767. These values are much

Table 9.3: Statistical comparison between the observed field data and simulated results at each weather station. A negative bias represents a simulated wind speed that is greater than the field data readings.

Weather Station	RMSE (m/s)	Bias (m/s)	STDE (m/s)
A	0.684	0.082	0.679
B	1.047	0.634	0.834
C	0.672	-0.141	0.657
D	0.997	0.526	0.847
E	0.680	0.332	0.591
F	1.295	-0.835	0.990
G	0.855	-0.130	0.846
H	0.762	0.124	0.752
I	0.774	0.318	0.706

lower than the values reported in David Carvalho and Santos (2012). In Michiorri *et al.* (2010) the standard deviation ranged from 0.9 to 1.5, whereas in our approach, it ranges from 0.6 to 1.0. For these reasons, we judge our simulation a reasonable realization of the wind conditions for the assumed global wind direction.

9.6.2 Dynamic Conductor Temperature

We perform the transient calculation of the ODE for temperature to demonstrate the dynamic thermal response of the conductor. Eq. 9.4, using field data from June 10, 2013 at weather station B. Wind speed and ambient temperature values used in the time-marching ODE are updated every three minutes, the rate of field data collection. The initial temperature of the conductor is first solved using the initial wind speed and steady-state equation in the form

$$R(T) = \frac{q_c + q_r - q_s}{I^2} \tag{9.22}$$

After calculating the resistance, the temperature T is extracted from tabulated data of resistance versus temperature using a linear interpolation. For this hypothetical case we picked ACSR 26/7 as the conductor type. Static rating was calculated under the summer

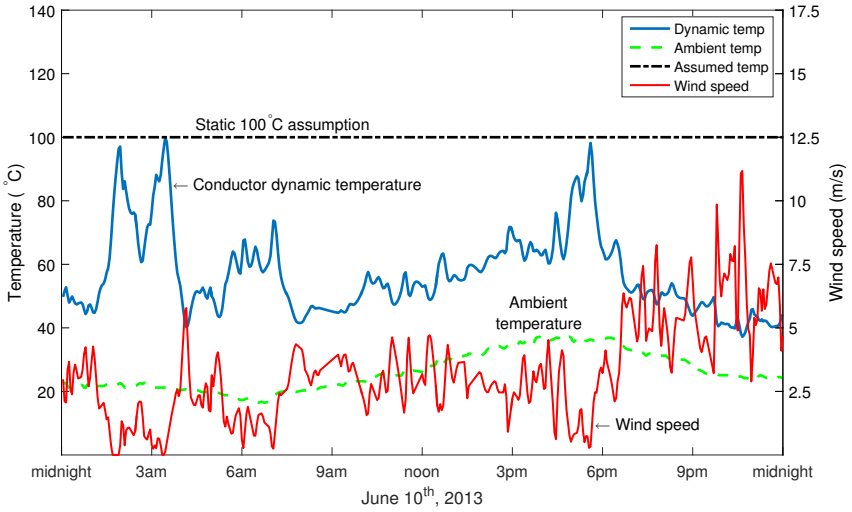


Figure 9.8: Conductor dynamic temperature calculated using wind speed and temperature from field data and compared with the assumed 100 °C static temperature when loaded with 1,025 Amps.

time assumptions of 0.61m/s wind, full sun on June 10 at 11AM (30° latitude, 0m elevation), and an ambient temperature of 40°C. Under these assumptions with an allowable maximum conductor temperature of 100°C the ampacity was calculated with Eq. 9.3, giving 1,025 Amps. We then imposed a current of 1,025A to the conductor and calculated the dynamic temperature using the wind speed and ambient temperature field data. Results, presented in Fig. 9.8, show that conductor temperature—overall—is much lower than the assumed static temperature. Equally important, when adverse conditions persist over long periods of time, TSPs will be informed when conductor temperature is in excess of their limits. Because of these advantages we recommend using a dynamic calculation method over the static rating practice.

As a feasibility test of a simulation-based approach, the conductor temperature was calculated using both field data and simulation results over a four-hour period. Since we are investigating the cooling effect of the wind, we kept other weather conditions constant. The initial conductor temperature used in the ODE calculation was solved using Eq. 9.22. We updated the wind speed every three minutes and solved the dynamic temperature with a RK4 method over the four hours. Figure 9.9 shows the true mean estimate and highlights

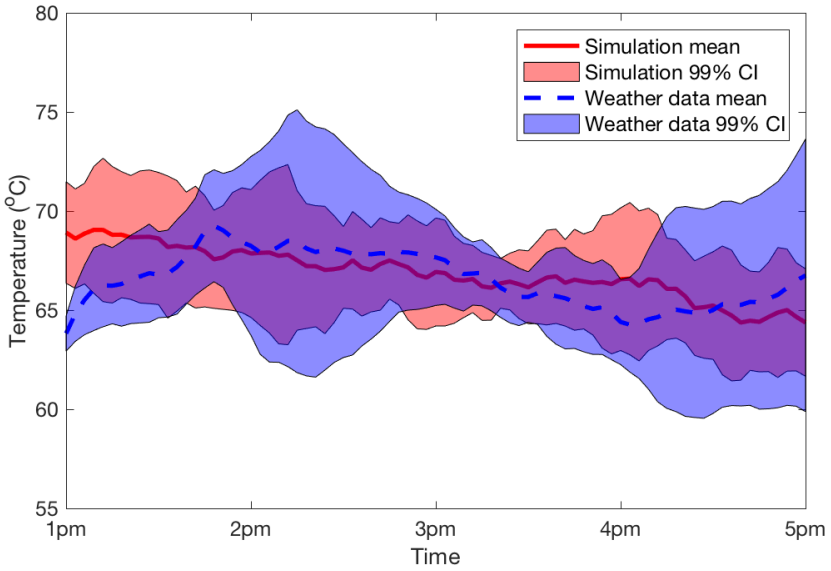


Figure 9.9: Resulting conductor temperature using the field data and simulation results. The highlighted area represents the 99% confidence interval.

the 99% confidence interval (CI). Data from nine weather stations are used to quantify the uncertainty or CI. The true mean estimate, v' is given as

$$v' = \overline{v} \pm CI\ (\%P) \tag{9.23}$$

where CI is the confidence interval at a given probability, P, and is defined as

$$CI = t_{df,P} \cdot S_{\overline{v}} \tag{9.24}$$

here $t_{df,P}$ is the statistical t-value with degrees of freedom, df. The degrees of freedom is the number of data points minus one. The standard error, $S_{\overline{v}}$, is defined as

$$S_{\overline{v}} = \frac{S_v}{\sqrt{N}} \tag{9.25}$$

where S_v is the standard deviation and N is the number of data points.

There are two important conclusions to take away from Fig. 9.9. First, the conductor temperature is much lower than the 100°C imposed by the static rating. Second, conductor

temperature variation relative to its location is significant as evidenced by the confidence interval. Spatial variation of conductor's temperature justifies the need to resolve wind field along the length of the line to identify critical segments.

9.6.3 Dynamic Ampacity

When the conductor is below its maximum allowable temperature, any amount of current can be put on the conductor for a limited amount of time. We therefore calculate the dynamic ampacity using the conductors present temperature and use an iterative method to solve the current that will heat the conductor to 100°C in 15 minutes. To demonstrate this, we show a hypothetical case using ACSR 27/6 conductor with an initial temperature of 60°C, wind speed of 3.5m/s, 40°C ambient temperature, and full sun. The ampacity is calculated to be 1,616A and the heating can be seen in Fig. 9.10. If the steady-state thermal rating, Eq. 9.3, is used under these conditions the ampacity would be at 1,571A and the conductor response would have to be assumed instantaneous. In other words a dynamic ampacity calculation method would enable the operator to see the actual thermal response of the conductor and its ability to ride out sudden drops in wind speed as it takes some time for the conductor to heat up.

The dynamic ampacity across the test area is therefore calculated using the temperature calculated in subsection 9.6.2 and the 15 minute transient temperature response to reach 100°C. The resulting ampacity mean and 99% CI using both field data and simulation results are shown in Fig. 9.11. The results show that there is significant additional capacity available that is not being utilized.

9.7 Conclusions

Dynamic line rating (DLR) holds great promise to alleviate transmission congestion that may hinder integration of new power generation. Using actual weather data from measurements

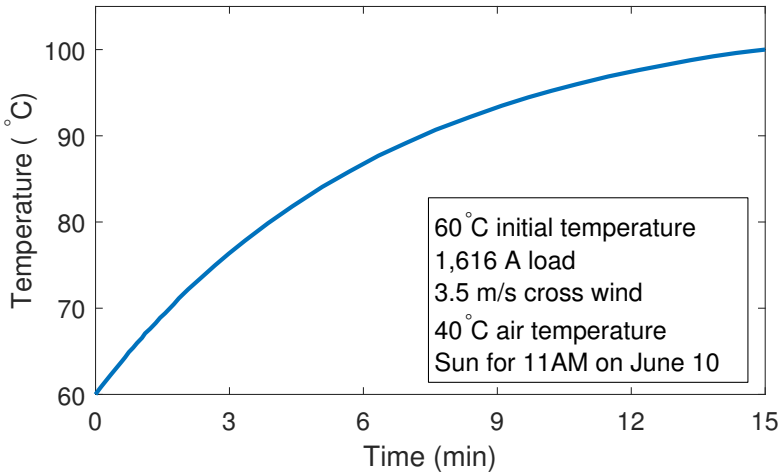


Figure 9.10: Conductor heating from 60 to 100°C in 15 minutes with a current of 1,616 Amps.

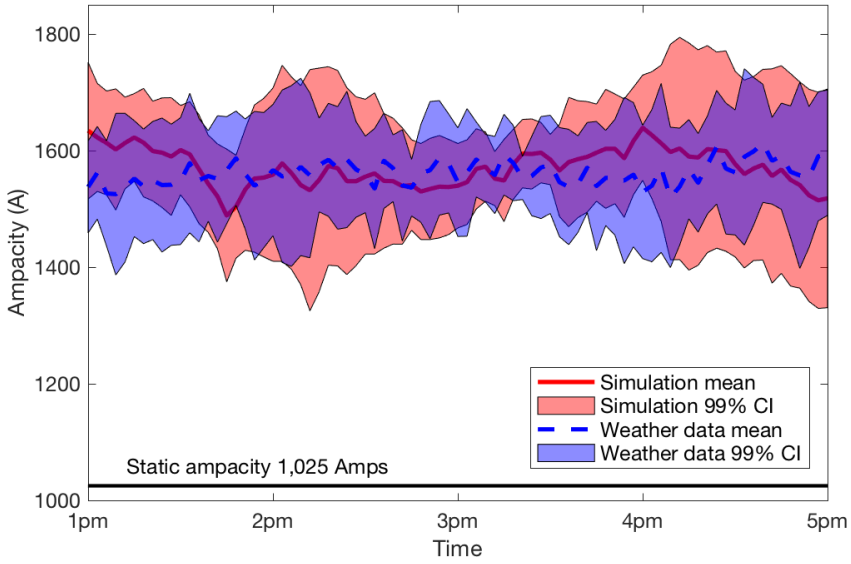


Figure 9.11: Resulting dynamic ampacity using field data and simulation results, i.e. this ampacity will heat the conductor from it's present temperature to 100°C in 15 minutes. The highlighted area represents the 99% confidence interval.

and an LES-based micro-scale wind solver, we demonstrated that ampacity of transmission lines in windy areas with complex terrain can be increased by 40-50% through the DLR concept. Our simulation-based approach is non-intrusive for the powerlines, and it is potentially much-more economical than building new transmission lines.

The use of a multi-GPU accelerated solver was critical to the success of our study. Instead of using a commercially available general-purpose computational fluid dynamics solver, we carefully selected our numerical methods and parameterizations to develop a fast wind solver, which was a multi-year effort with multiple developers (Thibault and Senocak, 2012; Jacobsen and Senocak, 2013, 2011; DeLeon *et al.*, 2012, 2013). The hardware-oriented design of our numerical solver —combined with the superior computing power of GPUs— enabled us to accommodate spatial and temporal resolutions that are much finer than the current practice for complex terrain wind simulations. Adoption of fine spatial resolutions is important for the resolution of terrain-induced motions, leading to more accurate line ratings. A potential benefit of using a multi-GPU accelerated solver is that simulations can be performed on workstations or clusters that have a much smaller footprint than central processing unit (CPU) based computing platforms.

Statistical analysis of simulation data for wind speed showed a very good agreement with field data. Additionally, we demonstrated that a transient calculation of the conductor temperature offers many advantages over the current practice based on the steady-state response of a conductor. A transient calculation enables us to take advantage of the thermal capacity of a conductor under variable wind conditions when considering a dynamic rating approach. We found that a fourth-order Runge-Kutta scheme performs much better in terms of accuracy and computation time than the forward Euler method suggested in the IEEE-738-2012 standard.

CHAPTER 10

Recommendations for Future Work

As stated in Section 1.2, the research questions that motivated the work in this dissertation were:

1. How reliable is the wall-modeled LES approach of hybrid RANS-LES for turbulent winds over complex terrain?

2. How accurate is the wind flow predicted when using equilibrium attached-flow assumptions to impose boundary conditions created by an immersed complex terrain?

3. How can buoyancy-driven flow physics be used as a turbulence inflow generation technique for LES of wind flow over complex terrain?

The following summarizes the work regarding each research question. Also, recommendations on future avenues of research are made for the different topics.

10.1 Research Question 1

The first research question led to investigation of multiple wall-modeled LES approaches that reduced the log-layer mismatch problem. The observation was made in channel flow that blending eddy-viscosity-type models leads to a lack of resolved Reynolds shear stresses, forcing the velocity gradients to unnatural values to maintain the global momentum balance. The proposal was to split the driving mean pressure gradient by splitting the mass flow rate into different regions based on wall-normal distance. While conserving mass, the effect was to give a missing degree of freedom that allowed the mean velocity to recover the expected logarithmic behavior. Second-order statistics were recovered as well.

The split-forcing approach performed very well on a very coarse mesh. A notable accomplishment is the recovery of first and second-order moments at such a high Reynolds number on a mesh that starts in the logarithmic region of the wall-bounded turbulent flow. However,

an obvious disadvantage is the applicability to general flows. When the flow is over parallel flat-plates, the determination of forcing regions is relatively simple as the cross-sectional area is constant. When the cross-sectional area varies in a domain, the task becomes much more challenging as the mass flow rate may not be known in advance at each streamwise location. Thus, determining the distribution of different forcing regions may not be possible.

One possible idea would be to run multiple cases until the forcing regions converged. First, an initial flow simulation without split-forcing is run to determine the mass flow rates in variable cross-sections. With the mass flow rates now known (or at least estimated), one could determine the forcing regions. Running again could lead to more adjustments to the forcing regions. However, this method is not guaranteed to converge and the process could be very time consuming. Another possible and more desirable idea would to use the split-forcing concept in conjunction with a turbulent inflow technique. Depending on the flow structures generated at the inflow, a forcing term may not need to be exact to provide the LES region with the missing Reynolds stresses that the RANS region cannot provide. This is just a hypothesis that two such methods can be used in conjunction and could be the focus of an in-depth research project.

Regarding the wind flowing over complex terrain studies performed in this dissertation, extending the split-forcing approach to complex terrain may not be possible. However, looking at dimensional profiles of velocity in Fig. 10.1, the difference between the velocity with a single forcing and the DNS benchmark velocity do not disagree much above $y/\delta = 0.2$. The logarithmic scales on the plots presented in Chapters 2 and 3 exaggerate the error. The dimensional plot reveals the mismatch actually taking place near the wall. As the mismatch is so close to the wall and the flow away from the surface agrees well in the dimensional sense, the hybrid RANS-LES approach is acceptable for studying wind flows over complex terrain. However, there is still room for improvement.

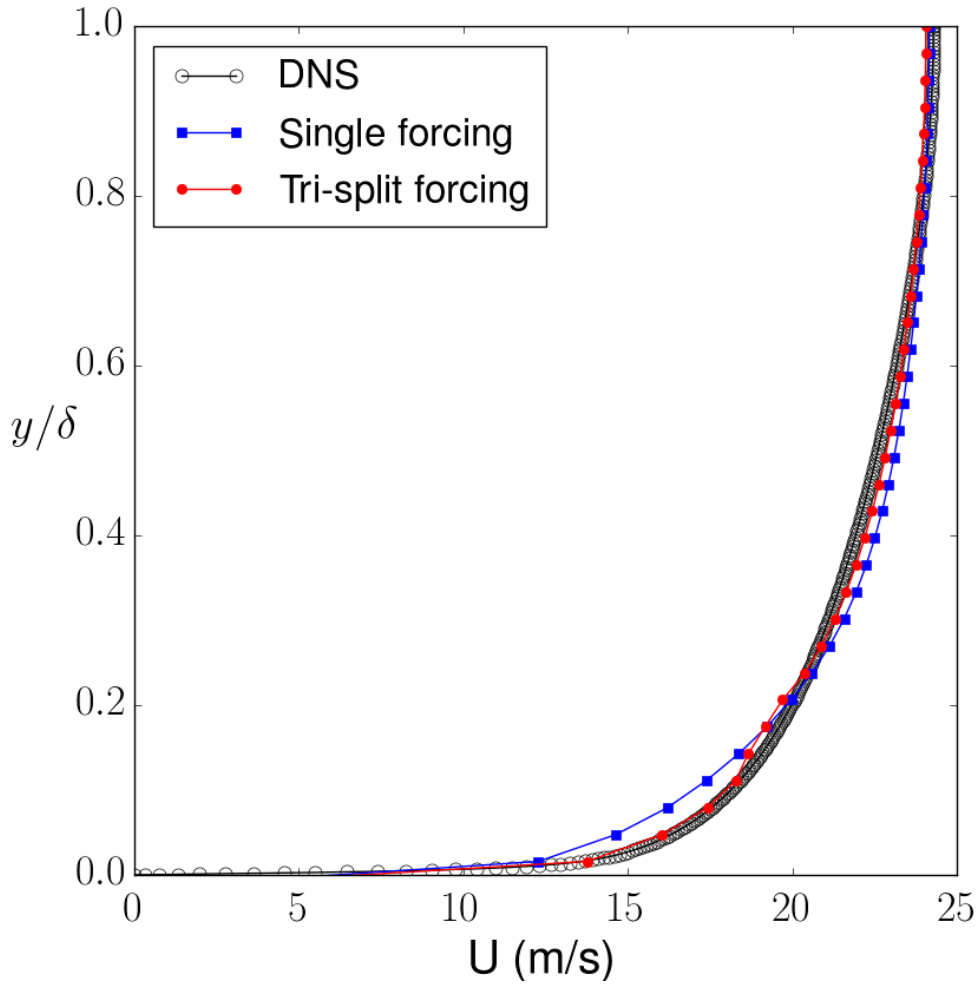


Figure 10.1: Hybrid RANS-LES mean velocity profiles from Section 3 presented in dimensional values. Single forcing refers to single driven mean pressure gradient with no split-forcing. Tri-split forcing refers to results obtained from split-forcing.

10.2 Research Question 2

The research question using IB methods for complex terrain was investigated in multiple facets. First, the geometric preprocessor and issues that arise with complex terrain representations was addressed. Next, the implementation of the immersed boundary method was validated in a DNS of katabatic slope flow. Finally, the use of a logarithmic reconstruction for velocity tangent to the terrain surface was investigated for complex terrain. Normal velocity was reconstructed linearly. The key was performing reconstruction on the eddy viscosity model to avoid the calculation of incorrect velocity gradients at the surface that could create numerically-unstable eddy viscosity values or extremely-large eddy viscosity values that pollute the solution.

Regarding the complex terrain IB investigation, the method gave good agreement in mean flow characteristics when compared to experimental benchmarks. Results showed good agreement and little sensitivity when the flow was attached, provided the mesh was adequate to resolve significant features of the terrain. In regions of separation, the method showed high sensitivity to mesh resolution. One explanation is the reconstruction scheme. Separated flow does not follow an equilibrium logarithmic profile and reconstructing separated flow in that manner is not ideal. A second explanation is the turbulence model. Prandtl's mixing length model assumes the flow is attached meaning the model is not adequate for separated flow. Both explanations contribute to the sensitivity of this IB hybrid RANS-LES approach in separated flows.

Future work should be in the investigation of more accurate reconstruction schemes for separated flow. Working with the primitive variable, e.g. velocity, is highly desirable for its simplicity and the reconstruction schemes should focus on that rather than surface stresses. A reconstruction scheme varying dynamically in operation that is dependent on attached or separated flow regimes would be highly desirable although not straightforward. The dynamic scheme would need to determine if flow separation will occur and when the

flow will reattach. There is also the matter of the appropriate interpolation reconstruction scheme to use in separated flow. Also, investigations into alternate wall models not assuming attached flow should be performed as well.

10.3 Research Question 3

The research question regarding turbulent inflow generation for LES was addressed by investing the perturbation box technique that uses stochastic buoyancy perturbations to create vortices that will develop into realistic turbulence over a fetch. The perturbation box method was applied to engineering-type turbulent flows and turbulent wind flowing over complex terrain. Overall, the perturbation box method showed rapid development of turbulence. The perturbation box method was able to generate accurate turbulence in engineering-type flows within a reasonable development fetch. The size of the box had an integral role in the size of turbulent eddies that developed.

The perturbation box method falls into the turbulent inflow generation category of synthetic methods. The goal of synthetic methods is to reduce the length of the development fetch. A development fetch is required as turbulent flow correlations are typically not known in advance for general flows. Turbulent flow structures generated by the synthetic method will require a distance to develop into realistic turbulence if the prescribed correlations are different than those natural to the flow. The perturbation box method does not require direct prescription of turbulent flow correlations. However, spatial correlations are implicitly set by the size of a perturbation box and the temporal correlations are implicitly set by the perturbation update frequency. As the correlations are implicitly prescribed, optimization is challenging as systematic trial and error is involved. The advantages outweigh the challenge as the perturbation box method can account for time-varying mean flow directions and also does not independently prescribe turbulent fluctuations onto the mean velocity field.

As the correlations are implicitly prescribed, the perturbation box method can still be optimized further than presented in this dissertation. The focus of this work in turbulent inflow

was to propose a simple model with parameter guidance that scaled to different flows and Reynolds numbers. After this investigation, further optimizing the current proposition will most likely not lead to significant reductions in development fetches. However, alterations can be made to the current perturbation box method that may lead to further improvements. One possibility is to vary the size of the boxes with distance from the wall and provide more consistency with wall-bounded turbulence. This may alter the current Richardson number suggestions. Another possibility is to have multiple Richardson number for different regions of the flow, e.g. inner layer vs. outer layer. This approach would be similar to the split-forcing method. A third possibility would be to alter the temporal correlations by having a perturbation update frequency consistent with time scales in wall-bounded turbulence. Again, this may require different Richardson number recommendations than those proposed in this study.

Rather than focusing on the parameters of the perturbation box method, perhaps other approaches could be used to accelerate the development of turbulence and should be the focus of future study of the perturbation box method. For example, artificial perturbation of the mean velocity profile coupled with the stochastic buoyancy perturbations could possibly generate turbulence more rapidly than demonstrated in this study. Another promising approach, perhaps the most promising of the recommendations made so far, is to apply a recycling plane technique, e.g. the approach of Lund *et al.* (1998). Recycling developed turbulence as an inflow condition has been successfully demonstrated and the perturbation box method may benefit from this approach.

10.4 Conclusion

The final recommendations regard the combination of the different research topics into a tool for predicting wind flow over complex terrain. As shown in Chapter 8, the perturbation box method can be used in conjunction with the immersed boundary method to produce turbulent wind flow over complex terrain that produces good agreement with experimen-

tal mean flow characteristics. If a solution to log-layer mismatch in general flows can be found, then that solution can be integrated into the complex terrain simulations. The end result will be a tool capable of simulating neutrally-stable wind flows over complex terrain. True simulation of the ABL will require Coriolis forcing terms to account for the Earth's rotation. The mean inflow conditions can be provided by can be provided by a mesoscale numerical weather prediction model. The addition of these components will allow for true neutral ABL simulations at the microscale. For non-neutral atmospheric conditions, the inflow conditions, immersed boundary method, and turbulence modeling will need to account for stable stratification, moisture effects, surface heat fluxes, and varying-surface-roughness modeling.

References

APRS #40R. APRS World, LLC, Winona, MN (2015).

NRG 40C Anemometer. Renewable NRG Systems, Hinesburg, Vermont (2015).

"Wind vision: A new era for wind power in the United States." Tech. rep., U.S. Department of Energy (2015).

Abe, K. "An improved anisotropy-resolving subgrid-scale model with the aid of a scale-similarity modeling concept." *Int J Heat Fluid Fl*, 39: (2013) pp. 42–52.

Anderson, W. "An immersed boundary method wall model for high-reynolds-number channel flow over complex topography." *Int. J. Numer. Meth. Fl.*, 71 (12): (2013) pp. 1588–1608.

Anderson, W. and Meneveau, C. "A large-eddy simulation model for boundary-layer flow over surfaces with horizontally resolved but vertically unresolved roughness elements." *Bound.-lay. Meteorol.*, 137 (3): (2010) pp. 397–415.

Araya, G., Castillo, L., Meneveau, C., and Jansen, K. "A dynamic multi-scale approach for turbulent inflow boundary conditions in spatially developing flows." *J. Fluid Mech.*, 670: (2011) pp. 581–605.

Arroyo, A., Castro, P., Martinez, R., Manana, M., Madrazo, A., Lecuna, R., and Gonzalez, A. "Comparison between ieee and cigre thermal behaviour standards and measured temperature on a 132-kv overhead power line." *Energies*, 8: (2015) pp. 13,660–13,671.

Baggett, J. S. "On the feasibility of merging LES with RANS for the near-wall region of attached turbulent flows." *Stanford Center for Turbulence Research Annual Research Briefs*, 20.

Balaras, E. and Benocci, C. "Subgrid-scale models in finite-difference simulations of complex wall bounded flows." *AGARD, Neuilly-Sur-Seine, France*.

Banerjee, B., Jayaweera, D., and Islam, S. "Risk constrained short-term scheduling with dynamic line ratings for increased penetration of wind power." *Renewable Energy*, 83: (2015) pp. 1139 – 1146.

Bardina, J., Ferziger, J. H., and Reynolds, W. C. "Improved turbulence models based on large eddy simulation of homogeneous, incompressible, turbulent flows." *Technical Report No. TF-19, Department of Mechanical Engineering, Stanford University, Stanford, CA*.

Bechmann, A., Sørensen, N., Berg, J., Mann, J., and Réhoré, P.-E. "The Bolund experiment, part II: Blind comparison of microscale flow models." *Bound.-Lay. Meteorol.*, 141: (2011) pp. 245–271.

Benocci, C. and Pinelli, A. "The role of the forcing term in the large eddy simulation of equilibrium channel flow." In *Engineering Turbulence Modeling and Experiments* (Edited by W. Rodi and E. Ganic). Elsevier, New York, 1990, pp. 287–296.

Berg, J., Mann, J., Bechmann, A., Courtney, M., and Jørgensen, H. "The Bolund experiment, part I: Flow over a steep, three-dimensional hill." *Bound.-Lay. Meteorol.*, 141: (2011) pp. 219–243.

Bhaganagar, K., Kim, J., and Coleman, G. "Effect of roughness on wall-bounded turbulence." *Flow Turbul. Combust.*, 72 (2-4): (2004) pp. 463–492.

Bhaskar, M., Jain, A., and Srinath, N. V. "Wind speed forecasting: Present status." *Power System Technology (POWERCON), 2010 International Conference*: (2010) pp. 1–6.

Bonneville Power Administration. "Bonneville power administration." `www.bpa.gov` (2017).

Brandt, T. "Direct numerical simulation of turbulent channel flow using finite difference schemes." In *VIII Finnish Mechanics Days, Helsinki University of Technology, Laboratory for Mechanics of Materials, Espoo, Finland*. 2003, pp. 639–50.

Bridson, R. *Fluid Simulation for Computer Graphics*. A K Peters, 2008.

Brower, M. (Ed.). *Wind resource assessment: a practical guide to developing a wind project.* John Wiley & Sons, Hoboken, NJ, 2012.

Cabot, W. and Moin, P. "Approximate wall boundary conditions in the large-eddy simulation of high Reynolds number flow." *Flow Turbul Combust*, 63 (1-4): (2000) pp. 269–291.

Chang, P.-H., Liao, C., Hsu, H.-W., Liu, S.-H., and Lin, C.-A. "Simulations of laminar and turbulent flows over periodic hills with immersed boundary method." *Comput. Fluid*, 92: (2014) pp. 233–243.

Chapra, S. C. and Canale, R. P. *Numerical Methods for Engineers.* McGraw Hill, Boston, MA, 2010, 6th ed.

Chester, S., Meneveau, C., and Parlange, M. "Modeling turbulent flow over fractal trees with renormalized numerical simulation." *J. Comput. Phys.*, 225 (1): (2007) pp. 427–448.

Cho, J., Kim, J., Lee, J., Kim, J., Song, I., and Choi, J. "Development and improvement of an intelligent cable monitoring system for underground distribution networks using distributed temperature sensing." *Energies*, 7 (2): (2014) pp. 1076–1094.

Choi, H. and Moin, P. "Grid-point requirements for large eddy simulation: Chapmans estimates revisited." *Phys Fluids*, 24 (1): (2012) p. 011,702.

Choi, J.-I., Oberoi, R. C., Edwards, J. R., and Rosati, J. A. "An immersed boundary method for complex incompressible flows." *Journal of Computational Physics*, 224 (2): (2007) pp. 757 – 784.

Chorin, A. J. "Numerical solution of the Navier-Stokes equations." *Math Comput*, 22 (104): (1968) pp. pp. 745–762.

Chow, F. and Street, R. "Evaluation of turbulence closure models for large-eddy simulation over complex terrain: flow over Askervein hill." *J. Appl. Meteorol. Clim.*, 48 (5): (2009) pp. 1050–1065.

CIGRE. "The thermal behaviour of overhead conductors." CIGRE WG 12.

CIGRE Working Group B2.12. "Guide for selection of weather parameters for bare overhead conductor ratings." Tech. Rep. 299 (2006).

Conan, B., Chaudhari, A., Aubrun, S., van Beeck, J., Hämäläinen, J., and Hellsten, A. "Experimental and numerical modelling of flow over complex terrain: The Bolund hill." *Bound.-Lay. Meteorol.*, 158 (2): (2016) pp. 183–208.

Cristallo, A. and Verzicco, R. "Combined immersed boundary/large-eddy-simulations of incompressible three dimensional complex flows." *Flow. Turbul. Combust.*, 77 (1-4): (2006) pp. 3–26.

Cuxart, J., Jiménez, M. A., and Martinez, D. "Nocturnal meso-beta basin and katabatic flows on a midlatitude island." *Monthly Weather Review*, 135 (3): (2007) pp. 918–932.

D. Eberly. "Distance Between Point and Line, Ray, or Line Segment." `http://www.geometrictools.com/Documentation/DistancePointLine.pdf` (2008).

David Carvalho, Alfred Rocha, M. G. and Santos, C. "A sensitivity study of the wrf model in wind simulation for an area of high wind energy." *Environ Modell Softw*, 33: (2012) pp. 23–34.

David Carvalho, Alfred Rocha, M. G.-G. and Santos, C. "WRF wind simulation and wind energy production estimates forced by different reanalyses: Comparison with observed data for Portugal." *Appl Energ*, 117: (2014) pp. 116–126.

Davidson, L. "The PANS k–ε model in a zonal hybrid RANS–LES formulation." *Int J Heat Fluid Fl*, 46: (2014) pp. 112–126.

Davidson, L. and Billson, M. "Hybrid LES-RANS using synthesized turbulent fluctuations for forcing in the interface region." *Int J Heat Fluid Fl*, 27 (6): (2006) pp. 1028–1042.

Dean, R. "Reynolds number dependence of skin friction and other bulk flow variables in two-dimensional rectangular duct flow." *J Fluid Eng-T ASME,*, 100 (2): (1978) pp. 215–223.

Deardorff, J. W. "A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers." *J Fluid Mech*, 41 (02): (1970) pp. 453–480.

Deb, A. K. *Powerline Ampacity System; Theory, Modeling, and Applications*. CRC Press, Boca Raton London New York Washington, D.C., 2000.

DeLeon, A. R. *GPU-accelerated Modeling of Microscale Atmospheric Flows Over Complex Terrain*. Master's thesis, Boise State University (2012).

DeLeon, R., Felzien, K., and Senocak, I. "Toward a GPU-accelerated immersed boundary method for wind forecasting over complex terrain." In *ASME 2012 Fluids Engineering Division Summer Meeting*, vol. 1. 2012, pp. 8–12.

DeLeon, R., Jacobsen, D., and Senocak, I. "Large-eddy simulations of turbulent incompressible flows on GPU clusters." *Comput. Sci. Eng.*, 15 (1): (2013) pp. 26–33.

DeLeon, R. and Senocak, I. "A novel fix to reduce the log-layer mismatch in wall-modeled large-eddy simulations of turbulent channel flow." In *ASME 2016 Fluids Engineering Division Summer Meeting collocated with the ASME 2016 Heat Transfer Summer Conference and the ASME 2016 14th International Conference on Nanochannels, Microchannels, and Minichannels*. 2016.

DeLeon, R. and Senocak, I. "Turbulent inflow generation for large-eddy simulation of high reynolds number incompressible flows through buoyancy perturbations." In *AIAA AVIATION Forum*. 2017.

di Mare, L., Klein, M., Jones, W., and Janicka, J. "Synthetic turbulence inflow conditions for large-eddy simulation." *Phys. Fluids*, 18 (2): (2006) p. 025,107.

Diebold, M., Higgins, C., Fang, J., Bechmann, A., and Parlange, M. B. "Flow over hills: a large-eddy simulation of the Bolund case." *Bound.-Lay. Meteorol.*, 148 (1): (2013) pp. 177–194.

Edwards, J., Choi, J.-L., Santanu, G., A., G. D., and D., E. J. "An immersed boundary method for general flow applications." In *Proceedings of the ASME Fluids Engineering Summer Meeting*, FEDSM-ICNMM2010-31097. Montreal, Canada, 2010.

Emeis, S. *Wind Energy Meteorology: Atmospheric Physics for Wind Power Generation.* Springer, Verlag Berlin Heidelberg, 2013.

EPRI. *Increased Power Flow Guidebook: Increasing Power Flow in Transmission and Substation Circuits.* Palo Alto, CA. 1010627 (2005).

Fadlun, E., Verzicco, R., Orlandi, P., and Mohd-Yusof, J. "Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations." *J. Comput. Phys.*, 161 (1): (2000) pp. 35 – 60.

Fedorovich, E. and Gibbs, J. "Numerical simulations of laminar and turbulent katabatic flows." (2015). Private communication.

Fedorovich, E. and Shapiro, A. "Structure of numerically simulated katabatic and anabatic flows along steep slopes." *Acta Geophys*, 57 (4): (2009) pp. 981–1010.

Fernandez, E., Albizu, I., Bedialauneta, M., Mazon, A., and Leite, P. "Review of dynamic line rating systems for wind power integration." *Renewable and Sustainable Energy Reviews*, 53: (2016) pp. 80 – 92.

Ferziger, J. H. and Perić, M. *Computational methods for fluid dynamics.* Springer Science & Business Media, 2012.

Foley, A. M., Leahy, P. G., Marvuglia, A., and McKeogh, E. J. "Current methods and advances in forecasting of wind power generation." *Renew. Energ.*, 37 (1): (2012) pp. 1–8.

Fröhlich, J. and von Terzi, D. "Hybrid LES/RANS methods for the simulation of turbulent flows." *Prog Aerosp Sci*, 44 (5): (2008) pp. 349–377.

Gentle, J., Myers, K. S., Baldwin, T., West, I., Hart, K., Savage, B., Ellis, M., and Anderson, P. "Concurrent wind cooling in power transmission lines." In *2012 Western Energy Policy Research Conference*. 2012. INL/CON-12-27012.

Gilmanov, A. and Acharya, S. "A hybrid immersed boundary and material point method for simulatin 3D fluid-structure interaction problems." *Int J Numer Meth Fl*, 56 (12): (2008) pp. 2151–2177.

Gilmanov, A. and Sotiropoulos, F. "A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies." *J. Comput. Phys.*, 207 (2): (2005) pp. 457 – 492.

Gilmanov, A., Sotiropoulos, F., and Balaras, E. "A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids." *J Comput Phys*, 191 (2): (2003) pp. 660 – 669.

Girimaji, S. "Partially-averaged Navier-Stokes model for turbulence: A reynolds-averaged navier-stokes to direct numerical simulation bridging method." *J Appl Mech*, 73 (3): (2006) pp. 413–421.

Golaz, J., Doyle, J., and Wang, S. "One-way nested large-eddy simulation over the Askervein Hill." *Journal of Advances in Modeling Earth Systems*, 1 (3).

Golaz, J.-C., Wang, S., Doyle, J., and Schmidt, J. "COAMPS®-LES: Model evaluation and analysis of second-and third-moment vertical velocity budgets." *Bound.-lay. Meteorol.*, 116 (3): (2005) pp. 487–517.

Greenwood, D. M., Gentle, J. P., Myers, K. S., Davison, P. J., West, I. J., Bush, J. W.,

Ingram, G. L., and Troffaes, M. C. "A comparison of real time thermal rating systems in the U.S. and the U.K." *IEEE Transactions on Power and Delivery*, 29 (4).

Grötzbach, G. "Direct numerical and large eddy simulation of turbulent channel flows." *Encyclopedia of Fluid Mechanics. West Orange NJ*, 13.

Guth, P. L. "MICRODEM." `http://www.usna.edu/Users/oceano/pguth/website/microdem/microdem.htm` (2013).

Hall, J. and Deb, A. "Prediction of overhead transmission line ampacity by stochastic and deterministic models." *IEEE T Power Deliver*, 3 (2): (1988) pp. 789–800.

Hamba, F. "A hybrid RANS/LES simulation of turbulent channel flow." *Theor Comp Fluid Dyn*, 16 (5): (2003) pp. 387–403.

Heckenbergerova, J. and Hosek, J. "Dynamic thermal rating of power transmission lines related to wind energy integration." In *Environment and Electrical Engineering (EEEIC), 2012 11th International Conference on*. 2012, pp. 798–801.

Holcombe, S. "INPOLYHEDRON." MATLAB Central File Exchange (2012).

Holyk, C., Liess, H. D., Grondel, S., Kanbach, H., and Loos, F. "Simulation and measurement of the steady-state temperature in multi-core cables." *Electr Pow Syst Res*, 116: (2014) pp. 54–66.

Hosek, J. "Dynamic thermal rating of power transmission lines and renewable resources." In *ES1002 Workshop, Paris, France*. 2011, pp. 1–3.

Hoyas, S. and Jiménez, J. "Scaling of the velocity fluctuations in turbulent channels up to $\mathrm{Re}_\tau = 2003$." *Phys Fluids*, 18 (1): 011702.

Hu, N. "A rescaling method for correcting log-layer mismatch in detached eddy simulation." *Science China Physics, Mechanics and Astronomy*, 56 (6): (2013) pp. 1165–1175.

Iaccarino, G. and Verzicco, R. "Immersed boundary technique for turbulent flow simulations." *Appl. Mech. Rev.*, 56 (3): (2003) pp. 331–347.

IEEE. "IEEE standard 738-2012: IEEE standard for calculating the current-temperature relationship of bare overhead conductors."

International Electrotechnical Commission. "IEC 61400 - online collection." `https://collections.iec.ch/iec61400?ref=gspromo` (2017).

Jackson, P. and Hunt, J. "Turbulent wind flow over a low hill." *Quarterly Journal of the Royal Meteorological Society*, 101 (430): (1975) pp. 929–955.

Jacobsen, D. A. and Senocak, I. "A full-depth amalgamated parallel 3D geometric multigrid solver for GPU clusters." In *49th AIAA Aerospace Science Meeting*. 2011.

Jacobsen, D. and Senocak, I. "Multi-level parallelism for incompressible flow computations on GPU clusters." *Parallel Comput.*, 39 (1): (2013) pp. 1–20.

Jafari, S., Chokani, N., and Abhari, R. S. "An immersed boundary method for simulation of wind flow over complex terrain." *J. Sol. Energ.-T. ASME*, 134 (1).

Jain, P. *Wind Energy Engineering*. McGraw-Hill, 2011.

Jarrin, N., Benhamadouche, S., Laurence, D., and Prosser, R. "A synthetic-eddy-method for generating inflow conditions for large-eddy simulations." *Int. J. Heat Fluid Fl.*, 27 (4): (2006) pp. 585–593.

Jarrin, N., Prosser, R., Uribe, J.-C., Benhamadouche, S., and Laurence, D. "Reconstruction of turbulent fluctuations for hybrid RANS/LES simulations using a synthetic-eddy method." *Int. J. Heat Fluid Fl.*, 30 (3): (2009) pp. 435–442.

Ji, C., Munjiza, A., and Williams, J. "A novel iterative direct-forcing immersed boundary method and its finite volume applications." *J. Comput. Phys.*, 231 (4): (2012) pp. 1797–1821.

Jimenez, A. *Wind Resource Assessment Report: Mille Lacs Indian Reservation, Minnesota*. National Renewable Energy Laboratory, 2013.

Jorge, R. S. and Hertwich, E. G. "Environmental evaluation of power transmission in norway." *Appl Energ*, 101: (2013) pp. 513–520.

Kang, S., Iaccarino, G., and Ham, F. "DNS of buoyancy-dominated turbulent flows on a bluff body using the immersed boundary method." *J Comput Phys*, 228 (9): (2009) pp. 3189–3208.

Kang, S., Lightbody, A., Hill, C., and Sotiropoulos, F. "High-resolution numerical simulation of turbulence in natural waterways." *Advances in Water Resources*, 34 (1): (2011) pp. 98 – 113.

Kawai, S. and Larsson, J. "Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy." *Phys Fluids*, 24 (1): (2012) p. 015,105.

Keating, A. and Piomelli, U. "A dynamic stochastic forcing method as a wall-layer model for large-eddy simulation." *J Turbul*, 7 (12): (2006) pp. 1–24.

Keating, A., Piomelli, U., Balaras, E., and Kaltenbach, H.-J. "A priori and a posteriori tests of inflow conditions for large-eddy simulation." *Phys. Fluids*, 16 (12): (2004) pp. 4696–4712.

Klein, M., Sadiki, A., and Janicka, J. "A digital filter based generation of inflow data for spatially developing direct numerical or large eddy simulations." *J. Comput. Phys.*, 186 (2): (2003) pp. 652–665.

Larsson, J., Lien, F., and Yee, E. "The artificial buffer layer and the effects of forcing in hybrid LES-RANS." *Int J Heat Fluid Fl*, 28 (6): (2007) pp. 1443–1459.

Le, H., Moin, P., and Kim, J. "Direct numerical simulation of turbulent flow over a backward-facing step." *J. Fluid Mech.*, 330 (1): (1997) pp. 349–374.

Lee, J., Cho, M., and Choi, H. "Large eddy simulations of turbulent channel and boundary layer flows at high Reynolds number with mean wall shear stress boundary condition." *Phys Fluids*, 25 (11): (2013) p. 110,808.

Lee, L. and LeVeque, R. J. "An immersed interface method for incompressible Navier–Stokes equations." *SIAM J. Sci. Comput.*, 25 (3): (2003) pp. 832–856.

Lee, M. and Moser, R. "Direct numerical simulation of turbulent channel flow up to $\mathrm{Re}_\tau \approx 5200$." *J Fluid Mech*, 774: (2015) pp. 395–415.

Leveque, R. and Li, Z. "The immersed interface method for elliptic equations with discontinuous coefficients and singular sources." *SIAM J. Numer. Anal.*, 31 (4): (1994) pp. 1019–1044.

Lopes, A. S., Palma, J., and Castro, F. "Simulation of the Askervein flow. part 2: Large-eddy simulations." *Bound.-Lay. Meteorol.*, 125 (1): (2007) pp. 85–108.

Lund, T. S., Wu, X., and Squires, K. D. "Generation of inflow data for spatially-developing boundary layer simulations." *J. Comput. Phys.*, 140 (2): (1998) pp. 233–258.

Lundquist, K., Chow, F., and Lundquist, J. "An immersed boundary method for the weather research and forecasting model." *Monthly Weather Review*, 138: (2010) pp. 796–817.

Lundquist, K. A., Chow, F. K., and Lundquist, J. K. "An immersed boundary method enabling large-eddy simulations of flow over complex terrain in the WRF model." *Monthly Weather Review*, 140 (12): (2012) pp. 3936–3955.

Mahrt, L. "Stably stratified atmospheric boundary layers." *Annual Review of Fluid Mechanics*, 46 (1): (2014) pp. 23–45.

Mahrt, L. "Stably stratified flow in a shallow valley." *Boundary-Layer Meteorology*: (2016) pp. 1–20.

Mann, J. "Wind field simulation." *Probabilist. Eng. Mech.*, 13 (4): (1998) pp. 269–282.

Marusic, I., Kunkel, G. J., and Porté-Agel, F. "Experimental study of wall boundary conditions for large-eddy simulation." *J. Fluid Mech.*, 446: (2001) pp. 309–320.

Mayor, S., Spalart, P., and Tripoli, G. "Application of a perturbation recycling method in the large-eddy simulation of a mesoscale convective internal boundary layer." *J. Atmos. Sci.*, 59 (15): (2002) pp. 2385–2395.

McDonald, B. "SURF2STL." MATLAB Central File Exchange (2004).

Mehta, D., Van Zuijlen, A., Koren, B., Holierhoek, J., and Bijl, H. "Large eddy simulation of wind farm aerodynamics: a review." *J. Wind Eng. Ind. Aerod.*, 133: (2014) pp. 1–17.

Meneveau, C., Lund, T., and Cabot, W. "A Lagrangian dynamic subgrid-scale model of turbulence." *J. Fluid. Mech.*, 319: (1996) pp. 353 – 85.

Metevosyan, J. "Wind power in areas with limited transmission capacity." In *Wind power in power systems* (Edited by T. Ackermann), chap. 20. John Wiley and Sons, 2005, pp. 433–501.

Michiorri, A., Nguyen, H.-M., Alessandrini, S., Bremnes, J. B., Dierer, S., Ferrero, E., Nygaard, B.-E., Pinson, P., Thomaidis, N., and Uski, S. "Forecasting for dynamic line rating." *Renewable and Sustainable Energy Reviews*, 52: (2015) pp. 1713 – 1730.

Michiorri, A., Taylor, P. C., and Jupe, S. C. E. "Overhead line real-time rating estimation algorithm: Description and validation." *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, 224 (3): (2010) pp. 293–304.

Mickle, R., Cook, N., Hoff, A., Jensen, N., Salmon, J., Taylor, P., Tetzlaff, G., and Teunissen, H. "The Askervein Hill project: Vertical profiles of wind and turbulence." *Bound.-Lay. Meteorol.*, 43 (1-2): (1988) pp. 143–169.

Mittal, R., Dong, H., Bozkurttas, M., Najjar, F., Vargas, A., and von Loebbeck, A. "A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries." *Journal of Computational Physics*, 227: (2008) pp. 4825–4852.

Mittal, R. and Iaccarino, G. "Immersed boundary methods." *Annu. Rev. Fluid Mech.*, 37: (2005) pp. 239–261.

Moeng, C. "A large-eddy-simulation model for the study of planetary boundary-layer turbulence." *J Atmos Sci*, 41 (13): (1984) pp. 2052–2062.

Mohd-Yusof, J. "Combined immersed boundary/B-spline methods for simulations of flow in complex geometries." Annual Research Briefs, Center for Turbulence Research, NASA-Ames/Stanford University (1997).

Moser, R. D., Kim, J., and Mansour, N. N. "Direct numerical simulation of turbulent channel flow up to $\mathrm{Re}_\tau = 590$." *Phys. Fluids*, 11 (4): (1999) pp. 943–945.

Muñoz-Esparza, D., Kosović, B., Mirocha, J., and van Beeck, J. "Bridging the transition from mesoscale to microscale turbulence in numerical weather prediction models." *Bound.-Lay. Meteorol.*, 153 (3): (2014) pp. 409–440.

Muñoz-Esparza, D., Kosović, B., van Beeck, J., and Mirocha, J. "A stochastic perturbation method to generate inflow turbulence in large-eddy simulation models: Application to neutrally stratified atmospheric boundary layers." *Phys. Fluids*, 27 (3): (2015) p. 035,102.

Munters, W., Meneveau, C., and Meyers, J. "Shifted periodic boundary conditions for simulations of wall-bounded turbulent flows." *Phys. Fluids*, 28 (2): (2016a) p. 025,112.

Munters, W., Meneveau, C., and Meyers, J. "Turbulent inflow precursor method with time-varying direction for large-eddy simulations and applications to wind farms." *Bound.-Lay. Meteorol.*, 159 (2): (2016b) pp. 305–328.

Nagib, H. M. and Chauhan, K. A. "Variations of von Kármán coefficient in canonical flows." *Phys Fluids*, 20 (10): (2008) p. 1518.

Negnevitsky, M., Mandal, P., and Srivastava, A. "An overview of forecasting problems and techniques in power systems." *Power Energy Society General Meeting, 2009. PES 2009. IEEE*: (2009) pp. 1–4.

Nicoud, F., Baggett, J., Moin, P., and Cabot, W. "Large eddy simulation wall-modeling based on suboptimal control theory and linear stochastic estimation." *Phys Fluids*, 13 (10): (2001) pp. 2968–2984.

Nikitin, N., Nicoud, F., Wasistho, B., Squires, K., and Spalart, P. "An approach to wall modeling in large-eddy simulations." *Phys Fluids*, 12 (7): (2000) pp. 1629–1632.

of Energy, D. *20% Wind Energy by 2030: Increasing Wind Energy's Contribution to U.S. Electricity Supply*. Technical report DOE/GO-102008-2567, 2008.

O'Rourke, J. *Computational Geometry in C*. Cambridge University Press, 1994.

Osher, S. and Fedkiw, R. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2003.

Pacheco-Vega, A., Pacheco, J. R., and Rodić, T. "A general scheme for the boundary conditions in convective and diffusive heat transfer with immersed boundary methods." *J Heat Transf*, 129 (11): (2007) pp. 1506–1516.

Pamiès, M., Weiss, P.-É., Garnier, E., Deck, S., and Sagaut, P. "Generation of synthetic turbulent inflow data for large eddy simulation of spatially evolving wall-bounded flows." *Phys. Fluids*, 21 (4): (2009) p. 045,103.

Papadopoulos, K. H. and Helmis, C. G. "Evening and morning transition of katabatic flows." *Boundary-Layer Meteorology*, 92 (2): (1999) pp. 195–227.

Parish, T. and Waight III, K. "The forcing of Antarctic katabatic winds." *Mon Weather Rev*, 115 (10): (1987) pp. 2214–2226.

Park, J., Basu, S., and Manuel, L. "Large-eddy simulation of stable boundary layer turbulence and estimation of associated wind turbine loads." *Wind Energy*, 17 (3): (2014) pp. 359–384.

Peskin, C. "Flow patterns around heart valves: A numerical method." *Journal of Computational Physics*, 10: (1972) pp. 252–271.

Peskin, C. S. "Numerical analysis of blood flow in the heart." *J. Comput. Phys.*, 25 (3): (1977) pp. 220 – 252.

Phillips, T., DeLeon, R., and Senocak, I. "Dynamic rating of overhead transmission lines over complex terrain using a large-eddy simulation paradigm." *Renewable Energy*, 108: (2017) pp. 380–389.

Phillips, T., Senocak, I., Gentle, J., Myers, K., and Anderson, P. "Investigation of a dynamic power line rating concept for improved wind energy integration over complex terrain." In *Proceedings of the ASME 2014 4th Joint US-European Fluids Engineering Division Summer Meeting*, FEDSM2014-21377. 2014.

Piomelli, U. "Wall-layer models for large-eddy simulations." *Prog Aerosp Sci*, 44 (6): (2008) pp. 437 – 446.

Piomelli, U., Balaras, E., Pasinato, H., Squires, K. D., and Spalart, P. R. "The inner–outer layer interface in large-eddy simulations with wall-layer models." *Int J Heat Fluid Fl*, 24 (4): (2003) pp. 538–550.

Poletto, R., Craft, T., and Revell, A. "A new divergence free synthetic eddy method for the reproduction of inlet flow conditions for LES." *Flow Turbul. Combust.*, 91 (3): (2013) pp. 519–539.

Pope, S. B. *Turbulent Flows*. Cambridge University Press, 2000.

Porté-Agel, F., Meneveau, C., and Parlange, M. B. "A scale-dependent dynamic model for large-eddy simulation: application to a neutral atmospheric boundary layer." *J. Fluid Mech.*, 415: (2000) pp. 261–284.

Porté-Agel, F., Wu, Y.-T., and Chen, C.-H. "A numerical study of the effects of wind direction on turbine wakes and power losses in a large wind farm." *Energies*, 6 (10): (2013) pp. 5297–5313.

Prandtl, L. "Bericht über die Entsteung der Turbulenz." *X. Angew. Math. Mech.*, 5: (1925) pp. 136 – 139.

Prandtl, L. *Führer durch die Strömungslehre*. Vieweg und Sohn, 1942.

Pu, J., Zheng, B., Leader, J., Wang, X., and Gur, D. "An automated CT based lung nodule detection scheme using geometric analysis of signed distance field." *Medical Physics*, 35: (2008) pp. 3453–3461.

Pytlak, P., Musilek, P., Lozowski, E., and Toth, J. "Modelling precipitation cooling of overhead conductors." *Electr Pow Syst Res*, 81 (12): (2011) pp. 2147–2154.

Radhakrishnan, S., Piomelli, U., Keating, A., and Lopes, A. S. "Reynolds-averaged and large-eddy simulations of turbulent non-equilibrium flows." *J Turbul*, 7 (63): (2006) pp. 1–30.

Raithby, G., Stubley, G., and Taylor, P. "The Askervein Hill project: a finite control volume prediction of three-dimensional flows over the hill." *Bound.-Lay. Meteorol.*, 39 (3): (1987) pp. 247–267.

Reding, J. "A method for determining probability based allowable current ratings for BPA's transmission lines." *IEEE T Power Deliver*, 9 (1): (1994) pp. 153–161.

Roman, F., Armenio, V., and Fröhlich, J. "A simple wall-layer model for large eddy simulation with immersed boundary method." *Phys. Fluids*, 21 (10): (2009) p. 101,701.

Sagaut, P. *Large Eddy Simulation for Incompressible Flows: An Introduction*. Springer-Verlag Berlin Heidelberg New York, 2002, 2 ed.

Salmon, J., Bowen, A., Hoff, A., Johnson, R., Mickle, R., Taylor, P., Tetzlaff, G., and Walmsley, J. "The Askervein Hill project: mean wind variations at fixed heights above ground." *Bound.-Lay. Meteorol.*, 43 (3): (1988) pp. 247–271.

Schumann, U. "Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli." *J. Comput. Phys.*, 18 (4): (1975) pp. 376–404.

Senocak, I., Ackerman, A., Kirkpatrick, M., Stevens, D., and Mansour, N. "Study of near-surface models for large-eddy simulations of a neutrally stratified atmospheric boundary layer." *Bound.-Lay. Meteorol.*, 124: (2007) pp. 405–424.

Senocak, I., Ackerman, A., Stevens, D., and Mansour, N. "Topography modeling in atmospheric flows using the immersed boundary method." Annual Research Briefs, Center for Turbulence Research, NASA-Ames/Stanford University (2004).

Senocak, I., Sandusky, M., DeLeon, R., Wade, D., Felzien, K., and Budnikova, M. "An immersed boundary geometric preprocessor for arbitrarily complex terrain and geometry." *J. Atmos. Ocean. Tech.*, 32 (11): (2015) pp. 2075–2087.

Seppa, T. O. "A practical approach for increasing the thermal capabilities of transmission lines." *IEEE T Power Deliver*, 8 (3): (1993) pp. 1536–1550.

Seppa, T. O., Mohr, R. D., and Stovall, J. "Error sources of real-time ratings based on conductor temperature measurements." *Report to CIGRE WG B2.36*: (2010) pp. 20–21.

Sethian, J. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.

Shapiro, A. and Fedorovich, E. "Unsteady convectively driven flow along a vertical plate immersed in a stably stratified fluid." *J Fluid Mech*, 498: (2004) pp. 333–352.

Shapiro, A. and Fedorovich, E. "Katabatic flow along a differentially cooled sloping surface." *J Fluid Mech*, 571: (2007) pp. 149–175.

Shur, M. L., Spalart, P. R., Strelets, M. K., and Travin, A. K. "A hybrid RANS-LES approach with delayed-DES and wall-modelled LES capabilities." *Int. J. Heat Fluid Fl.*, 29 (6): (2008) pp. 1638–1649.

Skamarock, W., Klemp, J., Dudhia, J., Gill, D., Barker, D., Duda, M., Huang, X., W.Wang, and Powers, J. "A description of the advanced research WRF version 3." NCAR Tech. Note NCAR/TN-475+STR (2008).

Skyllingstad, E. D. "Large-eddy simulation of katabatic flows." *Boundary-Layer Meteorology*, 106 (2): (2003) pp. 217–243.

Smagorinksy, J. "General circulation experiments with the primitive equations." *Mon Weather Rev*, 91 (3): (1963) pp. 99–164.

Smolarkiewicz, P. K., Sharman, R., Weil, J., Perry, S. G., Heist, D., and Bowker, G. "Building resolving large-eddy simulations and comparison with wind tunnel experiments." *Journal of Computational Physics*, 227 (1): (2007) pp. 633 – 653.

Soman, S., Zareipour, H., Malik, O., and Mandal, P. "A review of wind power and wind speed forcasting methods with different time horizons." *North American Power Symposium (NAPS)*: (2010) pp. 1–8.

Sotiropoulos, F. and Yang, X. "Immersed boundary methods for simulating fluid–structure interaction." *Prog. Aerosp. Sci.*, 65: (2014) pp. 1–21.

Spalart, P. "Direct simulation of a turbulent boundary layer up to r θ= 1410." *J. Fluid Mech.*, 187: (1988) pp. 61–98.

Spalart, P. R. "Detached-eddy simulation." *Annu. Rev. Fluid Mech.*, 41: (2009) pp. 181–202.

Spalart, P. R., Deck, S., Shur, M., Squires, K., Strelets, M. K., and Travin, A. "A new version of detached-eddy simulation, resistant to ambiguous grid densities." *Theor. Comput. Fluid Dyn.*, 20 (3): (2006) pp. 181–195.

Spalart, P., Jou, W., Strelets, M., and Allmaras, S. "Comments on the feasibility of LES for wings, and on a hybrid RANS/LES approach." *Advances in DNS/LES*, 1: (1997) pp. 4–8.

Stevens, R., Graham, J., and Meneveau, C. "A concurrent precursor inflow method for large eddy simulations and applications to finite length wind farms." *Renew Energ*, 68: (2014) pp. 46–50.

Stevens, R. and Meneveau, C. "Flow structure and turbulence in wind farms." *Annu. Rev. Fluid Mech.*, 49: (2017) pp. 311–339.

Stull, R. B. *An Introduction to Boundary Layer Meteorology*. Kluwer Academic Publishers, 1988.

Sunday, D. "Intersection of a Ray/Segment with a Triangle." `http://geomalgorithms.com/a06-_intersect-2.html#Segment-Triangle` (2001a).

Sunday, D. "Lines and Distance of a Point to a Line." `http://geomalgorithms.com/a02-_lines.html` (2001b).

Tabor, G. and Baba-Ahmadi, M. "Inlet conditions for large eddy simulation: a review." *Comput. Fluids*, 39 (4): (2010) pp. 553–567.

Tannehill, J. C., Anderson, D., and Pletcher, R. H. *Computational Fluid Mechanics and Heat Transfer, Second Edition*. Series in Computational and Physical Processes in Mechanics and Thermal Sciences. Taylor & Francis, 1997.

Taylor, P. and Teunissen, H. "The Askervein Hill Project: Report on the September/October 1983, main field experiment." (1985).

Taylor, P. and Teunissen, H. "The Askervein Hill project: overview and background data." *Bound.-Lay. Meteorol.*, 39 (1-2): (1987) pp. 15–39.

Tessicini, F., Iaccarino, G., Fatica, M., Wang, M., and Verzicco, R. "Wall modeling for large-eddy simulation using an immersed boundary method." *Annual Research Briefs, Stanford University Center for Turbulence Research, Stanford, CA*: (2002) pp. 181–187.

Thibault, J. C. and Senocak, I. "Accelerating incompressible flow computations with a Pthreads-CUDA implementation on small-footprint multi-GPU platforms." *J Supercomput*, 59: (2012) pp. 693–719.

Tseng, Y.-H. and Ferziger, J. H. "A ghost-cell immersed boundary method for flow in complex geometry." *J. Comput. Phys.*, 192 (2): (2003) pp. 593 – 623.

Tuerke, F. and Jiménez, J. "Simulations of turbulent channels with prescribed velocity profiles." *J Fluid Mech*, 723: (2013) pp. 587–603.

Umphrey, C. *A Stratified Turbulence Formulation and a Turbulent Inflow Boundary Condition for Large-Eddy Simulation of Complex Terrain Winds.* Master's thesis, Boise State University, Boise, Idaho, USA (2015).

Umphrey, C., DeLeon, R., and Senocak, I. "A Cartesian immersed boundary method to simulate stably stratified turbulent flows." In *54th AIAA Aerospace Science Meeting.* 2016.

Umphrey, C., DeLeon, R., and Senocak, I. "Direct numerical simulation of turbulent katabatic slope flows with an immersed-boundary method." in-press for Bound.-Lay. Meteorol. (2017).

Umphrey, C. and Senocak, I. "Turbulent inflow generation for the large-eddy simulation technique through globally neutral buoyancy perturbations." In *54th AIAA Aerospace Sciences Meeting.* 2016.

Uski-Joutsenvuo, S., Pasonen, R., and Rissanen, S. "Maximising power line transmission capability by employing dynamic line ratings: Technical survey and applicability in Finland."

van Heerwaarden, C., van Stratum, B., Heus, T., Gibbs, J., and Fedorovich, E. "Micro HH." `http://microhh.org` (2016).

Verzicco, R., Mohd-Yusof, J., Orlandi, P., and Haworth, D. "Large eddy simulation in complex geometric configurations using boundary body forces." *AIAA J.*, 38: (2000) pp. 427–433.

Vuorinen, V., Chaudhari, A., and Keskinen, J.-P. "Large-eddy simulation in a complex hill terrain enabled by a compact fractional step OpenFOAM® solver." *Advances in Engineering Software*, 79: (2015) pp. 70–80.

Walmsley, J. and Taylor, P. "Boundary-layer flow over topography: impacts of the Askervein study." In *Boundary-Layer Meteorology 25th Anniversary Volume, 1970–1995*. Springer, 1996, pp. 291–320.

Wu, X. "Inflow turbulence generation methods." *Annu. Rev. Fluid Mech.*, 49: (2017) pp. 23–49.

Xiao, H. and Jenny, P. "A consistent dual-mesh framework for hybrid LES/RANS modeling." *J Comput Phys*, 231 (4): (2012) pp. 1848–1865.

Xie, Z.-T. and Castro, I. P. "Large-eddy simulation for flow and dispersion in urban streets." *Atmos. Environ.*, 43 (13): (2009) pp. 2174–2185.

Xue, M., Wang, D., Gao, J., Brewster, K., and Droegemeier, K. "The Advanced Regional Prediction System (ARPS), storm-scale numerical weather prediction and data assimilation." *Meteorol. Atmos. Phys.*, 82 (1): (2003) pp. 139–170.

Ye, T., Mittal, R., Udaykumar, H., and Shyy, W. "An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries." *J. Comp. Phys.*, 156 (2): (1999) pp. 209–240.

Yeow, T., Cuerva-Tejero, A., and Perez-Alvarez, J. "Reproducing the bolund experiment in wind tunnel." *Wind Energy*, 18 (1): (2015) pp. 153–169.

You, J., Choi, H., and Yoo, J. Y. "A modified fractional step method of keeping a constant mass flow rate in fully developed channel and pipe flows." *KSME Int J*, 14 (5): (2000) pp. 547–552.

Yuan, J. and Piomelli, U. "Estimation and prediction of the roughness function on realistic surfaces." *J. Turbul.*, 15 (6): (2014a) pp. 350–365.

Yuan, J. and Piomelli, U. "Numerical simulations of sink-flow boundary layers over rough surfaces." *Phys. Fluid*, 26 (1): (2014b) p. 015,113.

Zhang, N., Zheng, Z., and Eckels, S. "Study of heat-transfer on the surface of a circular cylinder in flow using an immersed-boundary method." *Int J Heat Fluid Fl*, 29 (6): (2008) pp. 1558–1566.

Zhao, H. "A fast sweeping method for eikonal equations." *Mathematics of Computation*, 74: (2004) pp. 603–627.

Appendix A: Copyright Permissions

Copyright Information for *American Society of Mechanical Engineering*

The following is an electronic letter from American Society of Mechanical Engineering (ASME) granting permission for using the ASME conference paper, "A Novel Fix to Reduce the Log-Layer Mismatch in Wall-Modeled Large-Eddy Simulations of Turbulent Channel Flow,' reproduced in its entirety. The material appears on pages 16- 35 in this dissertation.

> Dear Prof. DeLeon, It is our pleasure to grant you permission **to use all or any part of** the ASME paper "A Novel Fix to Reduce the Log-Layer Mismatch in Wall-Modeled Large-Eddy Simulations of Turbulent Channel Flow," by Rey DeLeon and Inanc Senocak, Paper No. FEDSM2016-7698, cited in your letter for inclusion in a doctoral dissertation entitled Large-eddy Simulations of Microscale Turbulent Winds over Arbitrarily Complex Terrain to be published by University of Idaho.
>
> Permission is granted for the specific use as stated herein and does not permit further use of the materials without proper authorization. Proper attribution must be made to the author(s) of the materials. **Please note**: if any or all of the figures and/or Tables are of another source, permission should be granted from that outside source or include the reference of the original source. ASME does not grant permission for outside source material that may be referenced in the ASME works.
>
> As is customary, we request that you ensure full acknowledgment of this material, the author(s), source and ASME as original publisher. Acknowledgment must be retained on all pages where figure is printed and distributed.
>
> Many thanks for your interest in ASME publications.

Sincerely,

Beth Darchi // Publishing Administrator // ASME // 2 Park Avenue // New York, NY 10016-5990 // Tel 1.212.591.7700 // darchib@asme.org

Copyright Information for *Renewable Energy*

The following is a blanket statement regarding the use of the preprint version of the research article "Dynamic Rating of Overhead Transmission Lines over Complex Terrain Using a Large-eddy Simulation Paradigm" appearing in *Renewable Energy*, reproduced on pages 172- 198 in this dissertation.

Copyright Information for *Journal of Atmospheric and Oceanic Technology*

The following is an electronic letter from the American Meteorology Society (AMS) granting permission to use the manuscript version of "An Immersed Boundary Geometric Preprocessor for Arbitrarily Complex Terrain and Geometry" appearing in *Journal of Atmospheric and Oceanic Technologies*, reproduced on pages 60- 82 in this dissertation.

Dear Dr. DeLeon-

My name is Jinny Nathans and I'm the Permissions Officer at AMS. Your question was referred to me. This signed message constitutes permission to use the material requested in your email below.

You may use your article in your as part of your thesis with the following conditions:

+ please include the complete bibliographic citation of the original source, and

+ please include the following statement with that citation: ©American Meteorological Society. Used with permission.

Thanks very much for your request and if you need any further information, please get in touch with me. My contact information is below.

Regards,

Jinny Nathans

Permissions Officer

American Meteorological Society

jnathans@ametsoc.org

617 226-3905

Copyright Information for *Boundary-layer Meteorology*

The follow is an email from the permissions office at Springer granting permission to use the manuscript version of the published research article. The pertinent material is on pages 83-104 in this dissertation. The message is reproduced here with no alteration.

> Dear Anthony,
>
> Thank you for your request. As an author, you are welcome to reuse this content in other works created by yourself, provided that you clearly acknowledge the original publication. It has to be your own version of the article, not the published version.
>
> Use only for the purpose of defending your thesis and with a maximum of 100 extra copies in paper. **If the thesis is going to be published, permission needs to be reobtained.**
>
> - includes use in an electronic form, provided it is an author-created version of the thesis on his/her own website and his/her university's repository, including UMI (according to the definition on the Sherpa website: `http://www.sherpa.ac.uk/romeo/`);
>
> - is subject to courtesy information to the co-author or corresponding author.
>
> Kind regards,
>
> Oda Siqveland
>
> Rights Assistant
>
> SpringerNature
>
> The Campus, 4 Crinan Street, London N1 9XW, United Kingdom
>
> T +44 (0) 207 014 6851

Appendix B: Split-forcing Supplemental Material

Total Shear Stress and Local Friction Velocity

The theoretical total shear profile, $\tau(y)$, in a turbulent channel flow is given by

$$\tau(y) = \tau_w \left(1 - \frac{y}{\delta}\right) \tag{10.1}$$

where y is the wall-normal distance ranging from 0 to 2δ, δ is the channel half-height, and τ_w is the wall shear stress at $y = 0$ or $y = 2\delta$. The wall shear stress can also be represented by the friction velocity, u_τ, and density, ρ, as

$$\tau_w = \frac{u_\tau^2}{\rho}. \tag{10.2}$$

where friction velocity is a constant value for the entire channel flow. For incompressible flows, rho is typically set to unity an vanishes, which is applied here. Local friction velocity is the idea that friction is no longer a constant value but varies with wall-normal distance and is defined by

$$u_\tau(y) = \sqrt{\frac{\tau(y)}{1 - y/\delta}}. \tag{10.3}$$

Regarding using local friction velocity profile as normalization for total shear stress shown in Fig. 3.4a, we obtain the result shown in Fig. 10.2. The normalized local friction velocity is found in Fig. 3.4b.

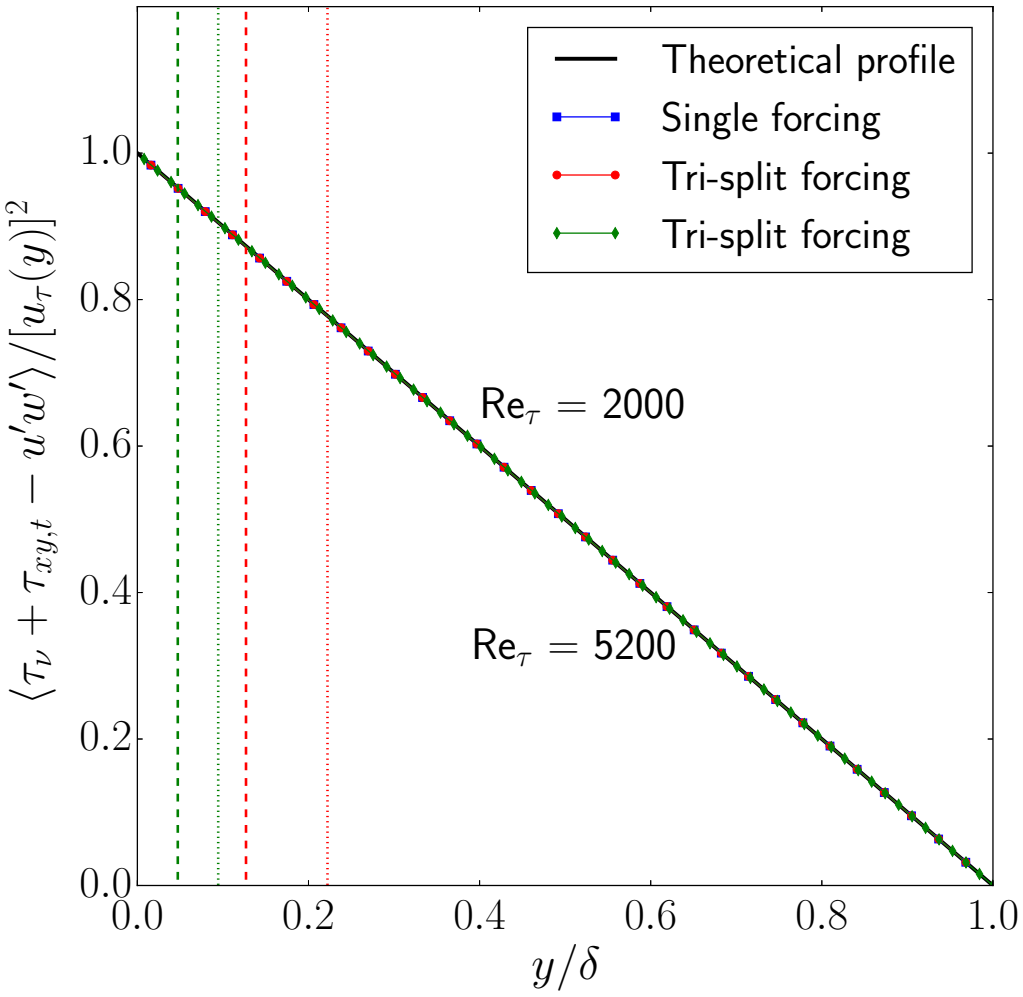


Figure 10.2: The total shear stress obtained from split-forcing normalized by local friction velocity.