CFD Modeling of Aerial Dispersion of Pollutants in Urban Environments

Alec Tauer
Marquette University

Follow this and additional works at: https://epublications.marquette.edu/theses_open

Part of the Engineering Commons

Recommended Citation
https://epublications.marquette.edu/theses_open/658
Population growth and urbanization across the globe is contributing to an increase in air pollution emissions. Because air pollution can negatively impact public health there is a desire to model the aerial dispersion of the pollutants in urban environments. Computational Fluid Dynamics (CFD) is becoming an increasingly common tool used to provide high spatial and temporal resolution of the wind flow and pollutant transport in urban environments. In the present study, CFD is utilized to model the aerial pollutant dispersion in three domains: a flat field, an idealized urban environment, and a real urban environment neighboring the Jones’ Island Water Reclamation Facility with topography.

A new method which utilizes meteorological data with high temporal resolution (one minute) is proposed to improve the lateral dispersion of pollutants in standard CFD studies where hourly-averaged data is used. The proposed and standard methods are tested in the three domains. The idealized cases (flat field and idealized urban environment) are validated using AERMOD, an empirically formulated Gaussian Plume Model, while the real domain is validated using field measurements. The proposed method improves the lateral dispersion in the flat field, but deviates from AERMOD in the idealized urban domain. In the real urban domain, the proposed method shows promise and is able to capture of the qualitative trends in the domain. However, CFD with hourly averaged meteorological data, instead of one minute, appears to provide a slightly better match with the field measurements.
ACKNOWLEDGMENTS

Alec Tauer

First, I would like to thank my thesis advisor, Dr. Roy, for all of his support and guidance throughout my undergraduate and graduate research. In addition, I would like to thank Dr. Borg, Dr. Bowman, Dr. McDonald, and Dr. McNamara for their comments and feedback on this thesis. The financial support through the GHR Seed Grant Opus College of Engineering gave me the opportunity for this research. I would also like to thank Tito Onwuzurike and Nate Hay for their support and contributions and the Milwaukee Metropolitan Sewerage District for providing information to assist with modeling Jones’s Island. Finally, I want to thank my family and friends for their support and encouragement throughout this process.
# TABLE OF CONTENTS

ACKNOWLEDGMENTS ................................................................. i

LIST OF TABLES ..................................................................... v

LIST OF FIGURES ................................................................. vi

1  INTRODUCTION ............................................................... 1

   1.1  Background .............................................................. 1

   1.2  Literature Review .................................................... 3

   1.3  Motivation ............................................................... 5

   1.4  Scope and Objectives ................................................ 6

   1.5  Organization ........................................................... 6

2  ATMOSPHERIC PHYSICS AND NUMERICAL METHODS ....... 8

   2.1  Atmospheric Physics ................................................ 8

       2.1.1  Convective Boundary Layer ................................. 9

       2.1.2  Stable Boundary Layer ................................... 11

   2.2  Numerical Modeling .................................................. 12

       2.2.1  Governing Equations ....................................... 14

       2.2.2  Inlet Profiles .................................................. 16

       2.2.3  Boundary Conditions ....................................... 18

       2.2.4  Solution Algorithm .......................................... 20

   2.3  Mesh Generation ..................................................... 22

   2.4  Directional Variability Method ................................. 29

3  DISPERSION IN IDEALIZED SETTINGS ............................... 33
B.1 makeBuildingFile.sh ........................................... 88

B.2 doMathForBuilding.f95 ......................................... 90
## LIST OF TABLES

2.1 Monin-Obukhov Length Stability Classification ................. 17

2.2 Urban Domain Boundary Conditions ............................. 19

3.1 Meteorological conditions used in the flat field domain .......... 36

3.2 Meteorological conditions used in the idealized urban domain .... 37

3.3 Sensitivity analysis of the DVM to changes in the turbulent Schmidt number ................................................. 53

3.4 Summary of the flat field pollutant dispersion simulations .......... 54

3.5 Summary of the idealized urban environment dispersion simulations .. 55

4.1 Meteorological conditions for the Jones’ Island simulation .......... 58

4.2 Mesh statistics for grid convergence .................................. 60

4.3 Meteorological conditions for the Jones’ Island simulation .......... 63
## LIST OF FIGURES

2.1 The lower level of the troposphere. ................................................. 8
2.2 Typical temperature profile of the CBL. ........................................... 10
2.3 Typical temperature profile of the SBL. ........................................... 11
2.4 Smog in Los Angeles, CA caused by a SBL. Photo from The Weather Network [1] ................................................................. 12
2.5 Transition between the CBL and the SBL over time. Modified from Figure 4.9 of Markowski et al. [2] ......................................................... 13
2.6 The PIMPLE algorithm with SIMPLE and PISO loops highlighted. .... 21
2.7 The building footprints (blue), area of interest (green box), and full domain (brown box). ................................................................. 23
2.8 A single, complex building footprint geometry can be divided into simple, overlapping shapes. ................................................................. 24
2.9 A rotated building component with basis vectors $e_1$ and $e_2$ and side lengths $L_1$ and $L_2$. ................................................................. 25
2.10 Image of Jones’ Island in QGIS used to identify pollutant source geometries. Pollutant sources are marked in pink and purple. .......... 26
2.11 A surface intersects a background cell and is refined to Level 3............ 27
2.12 Removed cells leave a jagged mesh comprised of hexahedral cells. .... 27
2.13 A castellated mesh (left) appears jagged while a snapped mesh (right) appears more realistic. ................................................................. 28
2.14 A snapped mesh (left) gets additional layers (right) near wall boundaries to improve turbulence wall functions. ................................. 29
2.15 The Jones’ Island Domain featuring 4.2 million cells. ....................... 30
2.16 Temporal resolution of wind data (speed or direction) from NWS and ASOS data. ................................................................. 31
2.17 An overview of the Directional Variability Method. 

3.1 The flat field domain and pollutant source (red) 

3.2 The idealized urban environment with buildings (red) and pollutant source (blue) 

3.3 The five sample lines within the flat field domain 

3.4 The five sample lines within the idealized urban environment 

3.5 Pollutant concentrations 40 m downwind in the flat field 

3.6 Pollutant concentrations 80 m downwind in the flat field 

3.7 Pollutant concentrations 40 m downwind in the idealized urban environment in stable conditions 

3.8 Pollutant concentrations 40 m downwind in the idealized urban environment in neutral conditions 

3.9 Pollutant concentrations 80 m downwind in the idealized urban environment in stable conditions 

3.10 Pollutant concentrations 80 m downwind in the idealized urban environment in neutral conditions 

3.11 Pollutant concentration along the domain centerline of the flat field 

3.12 Pollutant concentrations along the domain centerline in the idealized urban environment in a stable atmosphere 

3.13 Pollutant concentrations along the domain centerline in the idealized urban environment in a neutral atmosphere 

3.14 Vertical profile of pollutant concentration 40 m downwind from source in the flat field 

3.15 Vertical profile of pollutant concentration 80 m downwind from source in the flat field 

3.16 Vertical profile of pollutant concentration 40 m downwind from source in the idealized urban environment in a stable atmosphere
3.17 Vertical profile of pollutant concentration 40 m downwind from source in the idealized urban environment in a neutral atmosphere . . . 50

3.18 Vertical profile of pollutant concentration 80 m downwind from source in the idealized urban environment in a stable atmosphere . . . 50

3.19 Vertical profile of pollutant concentration 80 m downwind from source in the idealized urban environment in a neutral atmosphere . . . 51

4.1 Measurement route through the area of interest . . . . . . . . . . . . . . . 57

4.2 Dimensions of the Jones’ Island domain . . . . . . . . . . . . . . . . . . . 59

4.3 Velocity profile of 4 meshes behind a building . . . . . . . . . . . . . . . 60

4.4 Velocity profile of 4 meshes behind a building . . . . . . . . . . . . . . . 61

4.5 Simulated and measured H2S concentrations in the real domain . . . . 62
CHAPTER 1

INTRODUCTION

1.1 Background

Globally, the portion of the population living in urban environments is increasing. In 2018, the UN estimated over 55% of the population lived in urban areas and predicted that the percentage will increase to over 60% by 2030 [3]. This rapid urbanization has potential for an increase in social and cultural interaction, new opportunities, and better access to services [4]. However, large urban areas are associated with high anthropogenic activity and can suffer from high concentrations of aerial pollutants such as particulate matter (PM), carbon monoxide (CO), sulfur dioxide (SO\textsubscript{2}), and nitrogen dioxide (NO\textsubscript{2}). Exposure to these pollutants can lead to health problems including asthma, COPD, and lung cancer [5].

Aerial pollutant concentrations can show spatial and temporal variability [6] which can lead to an increase in uncertainty of field measurements. Modeling aerial pollutant dispersion is one tool that can be used combat the uncertainty by determining how the pollutant concentrations vary in space and time. Models can also give insight to how pollutants interact with the built and natural environment.

Due to the range of length scales and complexity of atmospheric flow, specific models are only effective for specific ranges. The longest range models are Lagrangian Particle Dispersion Models (LPDM) and are used for ranges greater than 50 km. These models use meteorological databases along with Lagrangian mechanics to predict pollutant trajectories. LPDMs are able to compute forward trajectories to determine where pollutants, such as nuclear
fallout from atomic bomb testing in Nevada [7], traveled after leaving the source. Another common application for LPDMs are backward trajectories to determine the source of a pollutant. Backward trajectory modeling has been used for many applications such as determining the source of ragweed pollen in Poland [8] and explaining the variability of tritium concentrations in precipitation over a research site in California [9]. However, LPDMs utilize very coarse meteorological data (> 10 km grid spacing) and cannot account for the complexity of the built environment.

At a smaller scale, Gaussian Plume Models (GPM) are empirically formulated to estimate pollution concentrations less than 50 km from the source. GPMs are often used for regulatory purposes due to their accuracy and low computational cost. The US Environmental Protection Agency developed a GPM, AERMOD, to operate under a variety of meteorological conditions and with complex terrain [10]. In addition to its use in industry for permitting and compliance studies, AERMOD is often used in research studies and has been used to determine emission rates from a swine farm through inverse modeling [11], estimate urban traffic emissions [12], and assess H₂S emissions from sewage treatment plants [13]. While the empirical formulation of GPMs can be useful, GPMs cannot accurately account for complex building geometries or time scales less than one hour.

At the smallest scale, Computational Fluid Dynamics (CFD) offers the highest spatial and temporal resolution for aerial pollutant dispersion modeling. Previously, CFD was reserved for laboratory scale studies due to its high computational cost and need for accurate measurements to validate the results. In recent decades, advancements in computational power and field measurement capabilities have made CFD modeling of atmospheric flow and aerial pollutant dispersion more common. With the increased computational power, researchers
have the tools and resources to study interactions between the built environment, atmosphere, and pollutants.

1.2 Literature Review

Urban Physics is a broad and growing field of research that attempts to understand physical processes in urban environments to improve the health and wellbeing of urban populations [14]. Many different scales of physical processes are studied within urban physics. These scales can range from turbulence within single street canyon [15] to the variation of wind speed across a city [16]. This wide range of scales results in different techniques for validation at different scales. Studies that focus on the scale of individual buildings and streets often use scaled models and wind tunnel tests for validation [17, 18, 19] while studies that focus on collections of buildings will use field measurements [20, 21].

Urban canyons are a common focus for dispersion modeling as most major cities have buildings along the edge of the street creating a canyon-like environment. This street-building arrangement can trap pollutants near the ground and significantly increase pedestrians’ exposure to pollutants. At this scale (∼100 m), structures along the street can have a large effect on the concentration and the pollutant concentrations and dispersion.

Parked cars have been shown to affect pedestrian level pollutant concentrations, but the extent of the effect is dependent on the degree of geometric detail of the car [22]. Oversimplification of the cars’ geometry can overestimate pedestrian exposure to pollutants. Trees can also affect the pollutant concentration and experience similar variation due to geometric simplification [23]. In addition to the trees’ geometry due to species and age, mesh simplification can also impact pedestrian level pollutant concentrations [24].
The contents of an urban canyon are not the only contributors to variation of pollutant concentrations. The buildings themselves can also affect the methods of pollutant dispersion. In deep urban canyons, the presence of balconies can reduce the rate of mass transfer of pollutants generated within the canyon (e.g. combustion products from vehicles, dust, etc.) into the atmosphere above the buildings [25].

While the studies at urban canyon scale provide insight into the effects of minor objects on the aerial dispersion of pollutants, they are limited in scope as they only evaluate wind flow perpendicular to the street canyon and assume an infinitely long street (i.e. symmetry conditions on the lateral domain boundaries). Modeling larger domains allows for more realistic scenarios, e.g. streets with finite lengths, varied wind directions.

In large scale tests (∼1000 m), features of building facades and smaller objects within the urban canyon are typically oversimplified in the mesh: either neglected if they are small enough (< 1 m) or far enough away from the area of interest or parameterized by increasing surface roughness lengths [14, 26, 27]. Even after neglecting these features, studies have shown good agreement between model and measurement. Toja-Silva et al. [28] used CFD to evaluate CO₂ emissions from an urban thermal power plant in Munich, Germany. The CFD model was able to accurately model the pollutant dispersion when compared field measurements using a spectrometer to measure the atmospheric column of CO₂. However, the pollutant concentrations were only measured along single columns so the study was not able to observe the lateral dispersion of the pollutants.

Another benefit of neglecting minor features, the computational costs related to mesh resolution can be reduced. With the freed computational power new features such as solar radiation models [26], traffic emission models [29], and
Numerical Weather Prediction (NWP) model coupling [27, 30] can be introduced and evaluated.

NWP coupling is a useful technique for large scale studies of real urban domains as it can be used to periodically update the CFD boundary conditions and capture changes in wind speed, wind direction, and temperature. Typically, meteorological data is time averaged over a one hour period. This loss of detail can reduce the accuracy of aerial pollutant dispersion models as wind direction can vary significantly over the averaging period [31]. Without the variability of wind speed and direction lateral dispersion of the plume is difficult to model with CFD.

1.3 Motivation

Methods previously used in the literature that incorporate wind variability, such as NWP coupling, can be very computationally expensive. One method proposed by Joseph, Hargreaves, and Lowndes [32] involved performing several CFD simulations and averaging the results using a Gaussian probability density function. However, this method can be computationally expensive as well because it requires several independent simulations to produce accurate lateral dispersion for a single case and cannot provide temporal resolution less than one hour.

The present study proposes a new method to improve the lateral dispersion of pollutants using Automated Surface Observing System (ASOS) 1-minute wind data. The method will perform one continuous simulation with changing boundary conditions as opposed to numerous independent simulations. The present method aims to improve the lateral dispersion of pollutants and provide results that can be validated in real urban environments.
1.4 Scope and Objectives

The main objective of this work is to accurately model aerial pollutant dispersion in real cities using CFD. The proposed Directional Variability Method (DVM) will be used to improve the simulation results by including variability of the wind speed and direction. To evaluate the effectiveness of the DVM, aerial pollutant dispersion will be simulated in three domains: a flat field, an idealized urban environment, and a real urban domain with topography. The idealized urban domain consists of an array of equally sized buildings and acts as a test case. Meanwhile, the real urban domain features real topography and building layouts consistent with the area surrounding the Jones’ Island Water Reclamation Facility and tests the effectiveness of the CFD modeling of real cities and the DVM.

The flat field and idealized urban domain cases will be validated using AERMOD as the GPM has been tested and validated in simple cases and provides a low cost alternative to wind tunnel testing. The real urban domain will be validated using a combination of AERMOD simulations and field measurements. Success with the DVM could allow future research to evaluate pollutant aging in urban environments.

1.5 Organization

The material in this thesis is divided into 5 chapters. Chapter 2 is subdivided into three sections: an overview the necessary atmospheric physics required to accurately model the Planetary Boundary Layer (PBL); the equations and numerical methods used to model wind flow; and the methods and requirements of creating quality meshes of urban domains. Chapter 3 of this thesis presents results from the idealized domains. Results from the Jones’ Island
domain are presented in Chapters 4. Finally, a summary and discussion of possible future work are presented in Chapter 5.
CHAPTER 2

ATMOSPHERIC PHYSICS AND NUMERICAL METHODS

2.1 Atmospheric Physics

The atmosphere is comprised of several layers. Each layer has its own distinct properties and methods to used to model. The present study focuses exclusively on the troposphere, the atmospheric layer in contact with earth’s surface.

The troposphere is comprised of two major components: the Atmospheric Boundary Layer (ABL) and the Free Atmosphere (FA) as shown in Figure 2.1. The FA is begins at a height of $1 - 2$ km above the surface and governed primarily by horizontal pressure gradients, cold fronts, and the Coriolis effect [33]. Due to the height and the large time and length scales of the FA, the mechanics are not necessary for the present study.

![Figure 2.1: The lower level of the troposphere.](image)

The ABL is comprised of the Atmospheric Surface Layer (ASL) and an upper layer. The ASL the section in contact with the earth’s surface and extends...
to height of 120 – 150 m. Because of the ASL’s close proximity to the surface it is governed primarily by surface heating and roughness. The Coriolis effect within the ASL is negligible as the previously mentioned processes dominate.

Within the upper layer of the ABL, pronounced diurnal effects can be observed. Therefore, discussion on the ABL must be broken into two parts to adequately discuss the major features.

2.1.1 Convective Boundary Layer

The Convective Boundary Layer (CBL), sometimes called the Unstable Boundary Layer, is observed primarily during the day when incoming solar radiation is the highest and increases the available energy in the surface energy budget.

The surface energy budget, Eqn. 2.1, describes how the energy from incoming solar radiation is divided between various heat transfer processes. The incoming radiation can be reflected back into the atmosphere or absorbed by the earth’s surface. The energy absorbed can then be re-emitted back into the atmosphere, heat the soil below the surface, heat the air above the surface, or change the phase of materials, e.g. water, on the surface.

\[
R_n = Q_S + Q_L + H \tag{2.1}
\]

where \( R_n \) is the net radiative heat flux of the surface, \( Q_S \) is the sensible heat flux, \( Q_L \) is the latent heat flux, and \( H \) is the soil heat flux [2]. Of the three terms on the RHS of Eqn. 2.1, only \( Q_S \) contributes significantly to the formation the CBL.

The sensible heat flux raises the temperature of the air near the surface. The CBL forms with the typical temperature profile shown in 2.2, where \( \theta \) is the temperature, \( z \) is the elevation, and \( z_i \) is the height of the boundary layer. The higher temperature near the surface caused by surface heating decreases the
density of the air at the surface. The lower density at lower elevations encourages vertical motion within the boundary layer. As the warm air rises it carries moisture and pollutants to the top of the CBL which is defined by a sharp temperature inversion at $z_i$ in Figure 2.2. The temperature inversion prevents air from rising into the FA. The trapped air and convective mixing cause the CBL to become well mixed after a sufficient amount of time.

The height of the CBL varies throughout the day. Just after sunrise, when the CBL forms, the boundary layer can have a height of as low as 300 m but can grow to a height of 2 km in the late afternoon. The growth and properties of the CBL are affected by land use and soil properties [34]. Therefore, vegetation [35], surface materials, and urban layout [36] affect the height CBL by altering the energy balance of Eqn. 2.1 through the albedo, Bowen ratio, and surface roughness length. Albedo is the fraction of radiation that is reflected by a surface; a surface with a high albedo, e.g. fresh snow, will absorb less thermal radiation and decrease the energy available in Eqn. 2.1. The Bowen ratio is the quotient of the sensible and latent heat fluxes; arid regions have a higher Bowen ratios as there is less water at the surface to evaporate thus leading to an increase in
sensible heat flux and CBL height. The surface roughness length is a length scale used to describe the roughness of a surface and wind profiles along the surface; areas with frequent and tall protrusions, e.g. urban environments, have larger surface roughness lengths.

2.1.2 Stable Boundary Layer

The Stable Boundary Layer (SBL) is characterized by a temperature profile that increases with elevation as shown in Figure 2.3. This increasing temperature profile suppresses vertical motion of air as density decreases with elevation. Without buoyancy effects, pollutants released near the surface are trapped at lower elevations and lead to poor air quality. SBLs can be detrimental to public health in urban areas as large amounts of pollutants are released near the surface from vehicles and industrial sources. Figure 2.4 shows a distinct layer of smog surrounding Los Angeles, California. This sharp transition is caused by suppressed vertical motion in the SBL.

The SBL is most often seen at night and is often referred to as the nocturnal boundary layer. However, the SBL can be observed at any time of day. Increased
cloud and snow coverage can allow the SBL to form by decreasing the net radiative heat flux or increasing the latent heat flux in Eqn. 2.1 [2].

The SBL is much shorter than the CBL, typically reaching only 0.3 km. While the CBL’s height is driven primarily by buoyant forces, the SBL’s height is driven primarily through mechanical forces. Therefore, the SBL grows during periods with higher wind speeds and in areas with larger physical barriers (i.e. larger surface roughness length).

Above the SBL and below the FA is the Residual Layer (RL). Figure 2.5 shows the transition between the CBL and the SBL and RL. The RL is an inactive zone that maintain many properties of the previous CBL. Because of the suppressed vertical motion, pollutants released in the SBL don’t reach the RL. Therefore, the RL can usually be neglected when modeling aerial pollutant dispersion. The methods used for modeling the ABL and generating the domains for CFD modeling will be discussed in sections 2.2 and 2.3, respectively.

2.2 Numerical Modeling

All of the CFD modeling in the present study was performed using OpenFOAM v5 [37]. OpenFOAM utilizes the Finite Volume Method and is
widely used in research studies for CFD simulations as it is open source, scales well across numerous processors, and can be easily modified to implement new models and solvers. The present study uses an adapted \textit{buoyantBoussinesqPimpleFoam} solver with Reynolds Averaged Navier-Stokes (RANS) turbulence modeling. This solver is used for incompressible flow with the Boussinesq approximation to account for buoyancy effects. The adaptation includes an additional passive-scalar transport equation to model pollutant transport.

Turbulence is a crucially important phenomena when modeling atmospheric flow through an urban environment. The large length scales attributed to the buildings and the potential for high wind speeds can lead to Reynolds numbers on the order of $10^7 - 10^9$. Large Eddy Simulations (LES) typically performs better as it more accurately predicts velocity and turbulent kinetic energy in urban canyons [15]. However, LES can be extremely computationally demanding and becomes infeasible with domains larger than idealized urban canyons. Therefore, the majority of large-scale, urban simulations use RANS turbulence modeling [38].
In this section, the governing equations for mass, momentum, turbulence, temperature, and pollutant dispersion are presented. In addition, the inlet profiles, boundary conditions, and algorithms used to solve the equations will be discussed.

2.2.1 Governing Equations

Atmospheric flow is governed by the Navier-Stokes equations in a rotating reference frame. When modeling flow that extends beyond the ASL, it is important to determine if the Coriolis force is applicable to the model. The Rossby number (Ro),

\[ \text{Ro} = \frac{U}{fL} \]  

where \( f = 2\omega \sin(\phi) \), \( U \) is the velocity, \( \phi \) is the latitude, \( L \) is the length scale, and \( \omega (= 7.2921 \times 10^{-5} \text{ rad/s}) \) is the angular velocity of the reference frame, provides context on the importance of the Coriolis force for any numerical model. With wind speeds on the order of 10 ms\(^{-1}\) over length scales of 1000 m in the mid-latitudes (\( \phi \approx 45^\circ \)), \( \text{Ro} \approx 100 \) meaning the Coriolis force is negligible in the present study as it two orders of magnitude less than the inertial forces.

To account for atmospheric stability, temperature gradients and density need to be accounted for in the governing equations. As previously stated, an incompressible solver with the Boussinesq approximation is used. The incompressible, Reynolds-Averaged mass and momentum equations using the Boussinesq approximation can be written as

\[ \frac{\partial \bar{u}_i}{\partial x_i} = 0, \]  

\[ \frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho_0} \frac{\partial}{\partial x_i} (\bar{p} - \rho_0 g_i z) + \frac{\partial}{\partial x_i} \left( 2\nu S_{ij} - \bar{u}_j^T \bar{u}_i \right) - g_i \beta (T - T_0), \]  

where \( \rho_0 \) and \( T_0 \) are the reference density and temperature, respectively, \( \beta \) is the coefficient of thermal expansion, \( g_i \) is the gravitational acceleration vector, \( z \) is the
elevation, $T$ is the temperature, $\overline{p}$ and $\overline{u}$ are the mean pressure and velocity, respectively, from Reynolds decomposition, the subscripts $i$ and $j$ are the index notations referring to vector components, and $S_{ij}$ is the mean stress rate tensor,

$$ S_{ij} = \frac{1}{2} \left( \frac{\partial \overline{p}_i}{\partial x_j} + \frac{\partial \overline{p}_j}{\partial x_i} \right). \tag{2.5} $$

The $u_i' u_j'$ term in Eqn. 2.4 is Reynolds stress tensor and is unclosed. This requires a turbulence closure model to numerically solve the equations. In the present study, the standard $k - \epsilon$ model is used to model the transport of turbulent kinetic energy, $k$, and the turbulent kinetic energy dissipation rate, $\epsilon$.

The equations for $k$ and $\epsilon$ can be written as

$$ \frac{\partial k}{\partial t} + u_i \frac{\partial k}{\partial x_i} = \frac{\partial}{\partial x_i} \left[ \left( \nu + \nu_t \sigma_k \right) \frac{\partial k}{\partial x_i} \right] + P_k - \epsilon, \tag{2.6} $$

$$ \frac{\partial \epsilon}{\partial t} + u_i \frac{\partial \epsilon}{\partial x_i} = \frac{\partial}{\partial x_i} \left[ \left( \nu + \nu_t \sigma_\epsilon \right) \frac{\partial \epsilon}{\partial x_i} \right] + \frac{\epsilon}{k} \left( C_1 \epsilon P_k - C_2 \epsilon \right), \tag{2.7} $$

where

$$ \nu_t = C_\mu \frac{k^2}{\epsilon}, \tag{2.8} $$

$$ P_k = -u_i' u_j' \frac{\partial u_i}{\partial x_j}, \tag{2.9} $$

$$ -u_i' u_j' = 2\nu_t S_{ij} - \frac{2}{3} k \delta_{ij} \tag{2.10} $$

where $\delta_{ij}$ is the Kronecker delta and $\sigma_k(= 1.0), \sigma_\epsilon(= 1.3), C_\mu (= 0.09), C_1 (= 1.44), C_2 (= 1.92)$ are constants determined through experimental validation [39].

Temperature is an active scalar that influences the transport of momentum through buoyancy effects. The temperature equation can be written as

$$ \frac{\partial T}{\partial t} + u_i \frac{\partial T}{\partial x_i} = \frac{\partial}{\partial x_i} \left[ \left( \alpha + \frac{\nu_t}{\Pr_t} \right) \frac{\partial T}{\partial x_i} \right], \tag{2.11} $$

where $\alpha$ is the thermal diffusivity and $\Pr_t$ is the turbulent Prandtl number.
In the present study, the pollutant is treated as a passive scalar meaning the pollutant has no effect on the flow field, as commonly practiced in the literature [18, 28, 40, 41, 42]. Treating the pollutant as an active scalar would require a compressible, multi-species solver which can be computationally infeasible in large scale simulations. The passive pollutant equation can be written as

\[
\frac{\partial \overline{C}}{\partial t} + \overline{u_i} \frac{\partial \overline{C}}{\partial x_i} = \frac{\partial}{\partial x_i} \left[ \left( D + \frac{\nu_i}{Sc_t} \right) \frac{\partial \overline{C}}{\partial x_i} \right]
\] (2.12)

where \(Sc_t\) is the turbulent Schmidt number, \(D\) is the mass diffusivity of the pollutant, and \(\overline{C}\) is the mass fraction of the pollutant. The turbulent Schmidt number is a dimensionless quantity that compares the importance of advection and diffusion for mass transport. Unfortunately, there are no methods to determine \(Sc_t\) for a given domain prior to the simulation as the value is dependent on the geometry and flow conditions. However, it is commonly agreed that \(Sc_t\) should fall within the range of \(0.3 - 1.2\) for urban environments so multiple values should be tested [28, 29, 43, 44, 45].

### 2.2.2 Inlet Profiles

The inlet velocity and turbulence profiles in the ABL take the form of

\[
\overline{\pi}(z) = \frac{u_s}{\kappa} \ln \left( \frac{z + z_0}{z_0} \right),
\] (2.13)

\[
k = \frac{u_s^2}{\sqrt{C_\mu}},
\] (2.14)

\[
\epsilon(z) = \frac{u_s^3}{\kappa(z + z_0)},
\] (2.15)

where \(\kappa(= 0.41)\) is the von Karman constant and \(z_0\) is the surface roughness length. The surface friction velocity, \(u_s\), is determined by providing a reference wind speed \((u_{ref})\) at a reference height \((z_{ref})\) and solving Eqn. 2.13.

A temperature profile is also prescribed at the inlet to induce buoyancy effects within the domain. Empirical formulation of the temperature profile
utilize the Monin-Obukhov length, $L$, a dimensioned scalar used to describe the stability of the ABL and formulated as

$$L = -\frac{\rho c_p T_{ref} u_*^3}{\kappa g Q_s} \quad (2.16)$$

where $\rho$ is the density of air, $g$ is the acceleration due to gravity, $c_p$ is the specific heat of air at constant pressure, and $Q_s$ is the sensible heat flux. As previously mentioned, the Monin-Obukhov length is a dimensioned quantity and describes the height at which turbulence is generated more by buoyancy than wind shear; however, it is not necessarily measurable. It is better to think of the quantity as a stability classification as shown in Table 2.1.

<table>
<thead>
<tr>
<th>Stability</th>
<th>$L$ (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stable</td>
<td>$10 \leq L \leq 500$</td>
</tr>
<tr>
<td>Neutral</td>
<td>$</td>
</tr>
<tr>
<td>Unstable</td>
<td>$-500 \leq L \leq -10$</td>
</tr>
</tbody>
</table>

$L$ can be calculated by utilizing Eqn. 2.16. In the present study, $L$ is determined using AERMET, the meteorological preprocessor to AERMOD. Utilizing the AERMET data for the CFD simulations in beneficial to ensure equal comparisons between CFD and AERMOD. The temperature profile can be written using a reference temperature as

$$T(z) = T_{ref} + \frac{T_*}{\kappa} \left( 0.95 \ast \ln \left( \frac{z}{z_{ref}} \right) - \phi \left( \frac{z}{L} \right) + \phi \left( \frac{z_{ref}}{L} \right) \right) + z \lambda_{adia} \quad (2.17)$$

where,

$$T_* = -\frac{Q_s}{\rho c_p u_*}$$

$$\phi \left( \frac{z_i}{L} \right) = 2 \ln \left( 0.5 + 0.5 \left( 1 - 11.6 \frac{z_i}{L} \right)^{0.5} \right)$$
and $T_{ref}$ is the reference temperature taken at the height $z_{ref}$, $\rho$ is the density of air, $c_p$ is the specific heat of air, and $\lambda_{adia}(= -0.009766 \, \text{K/m})$ is the adiabatic lapse rate. The adiabatic lapse rate describes the decrease in temperature as a function of height due to the decrease in pressure at higher altitudes.

### 2.2.3 Boundary Conditions

Proper boundary condition implementation is vital to accurate modeling of the ABL. As stated in the previous section, inlets are given specific profiles to allow for flow development that matches realistic scenarios. Outlets are given outflow or zero gradient conditions and objects such as buildings and bridges are given no-slip conditions.

The boundary conditions applied to the top of the domain are dependent on the meteorological conditions. In the CBL, the top must be able to act as an outlet to allow for the vertical motion caused by buoyancy. In the SBL, several options are available: fixed value, slip, and symmetry. The fixed value condition is often used when the top of the domain is close to the ground. Slip and symmetry are used when the top of the domain is very far away from the area of interest. The present study elected to use a symmetry condition for the top boundary condition when modeling the SBL. A complete list of boundary conditions can be seen in Table 2.2.

As shown in Table 2.2, the turbulent properties ($\epsilon, k, \nu_t$) are all given wall functions. The wall function for each turbulent property is slightly different and it is worth describing the specifics. For $\epsilon$, OpenFOAM’s `epsilonWallFunction` is used to prescribe a Dirichlet condition at the walls. The fixed value is described by Eqn. 2.18 when $y^+ > 5$, where

$$\epsilon = wC_\mu \frac{k^{3/2}}{\nu_l y} \quad (2.18)$$
Table 2.2: Urban Domain Boundary Conditions

<table>
<thead>
<tr>
<th></th>
<th>$\alpha_t$</th>
<th>C</th>
<th>$\epsilon$</th>
<th>$k$</th>
<th>$v_t$</th>
<th>$p$</th>
<th>$p_{rgh}$</th>
<th>$T$</th>
<th>$u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inlet</td>
<td>c</td>
<td>fV</td>
<td>Eq. 2.15</td>
<td>Eq. 2.14</td>
<td>c</td>
<td>tP</td>
<td>fFP</td>
<td>Eq. 2.17</td>
<td>Eq. 2.13</td>
</tr>
<tr>
<td>Outlet</td>
<td>c</td>
<td>zG</td>
<td>zG</td>
<td>zG</td>
<td>c</td>
<td>zG</td>
<td>zG</td>
<td>zG</td>
<td>zG</td>
</tr>
<tr>
<td>Top</td>
<td></td>
<td></td>
<td>Symmetry (SBL) or Outlet (CBL)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ground</td>
<td>wF</td>
<td>zG</td>
<td>wF</td>
<td>wF</td>
<td>zG</td>
<td>fFP</td>
<td>fV</td>
<td>nS</td>
<td></td>
</tr>
<tr>
<td>Structures</td>
<td>wF</td>
<td>zG</td>
<td>wF</td>
<td>wF</td>
<td>zG</td>
<td>fFP</td>
<td>fV</td>
<td>nS</td>
<td></td>
</tr>
<tr>
<td>Sources</td>
<td>c</td>
<td>fV</td>
<td>iV</td>
<td>iV</td>
<td>c</td>
<td>zG</td>
<td>fFP</td>
<td>fV</td>
<td>fV</td>
</tr>
</tbody>
</table>

Nomenclature: $c =$ Calculated, $fV =$ fixedValue, $tP =$ totalPressure, $fFP =$ fixedFluxPressure, $zG =$ zeroGradient, $wF =$ wallFunction, $nS =$ noSlip, $iV =$ inletValue

and $w$ is a weighting factor, $C_\mu(=0.09)$ is a $k - \epsilon$ model constant, $k$ is the turbulent kinetic energy, $v_t$ is the turbulent viscosity, and $y$ is the distance from the wall to the cell center. For $k$, OpenFOAM’s $kqRWallFunction$ is used which applies a zero gradient normal to the wall. Finally, the OpenFOAM condition $nutkWallFunction$ is used for $v_t$ which applies a Dirichlet condition at the wall described by Eqn. 2.19 when $y^+ > 5$, where,

$$v_t = v \left( \frac{y^+ \kappa}{\ln(Ey^+)} - 1 \right), \quad (2.19)$$

and $\kappa$ is the von Karman constant, $E$ is the wall roughness parameter, and $y^+$ is the non-dimensional wall distance estimated by Eqn. 2.20,

$$y^+ = C_0^{0.25} \frac{\sqrt{k}}{v}. \quad (2.20)$$

It can also be seen from Table 2.2 that the turbulent scalars $k$ and $\epsilon$ are given inletValue conditions at the sources. For both $k$ and $\epsilon$, their inletValue conditions act as fixed value conditions. For $k$ the value is calculated using Eqn. 2.21,

$$k_{iV} = 1.5(I|\bar{u}|)^2, \quad (2.21)$$
where $I$ is the intensity and $\bar{u}$ is the velocity at the patch. For $\epsilon$, the value is calculated using Eqn. 2.22,

$$\epsilon_{iV} = \frac{C^0.75\mu k^{1.5}}{L}$$

(2.22)

where, $k$ is the turbulent kinetic energy and $L$ is the mixing length.

The zeroGradient condition applied to the walls (i.e. buildings, structures, ground) for the pollutant, $C$, acts as a zero-flux condition, preventing pollutants from passing through the surface. The fixedFluxPressure condition used with $p_{rgh}$ acts as a zero gradient condition but is used when gravity must be accounted for.

Also, it should be noted that radiative heat transfer is not included in the simulation. The most significant radiative heating within the system is the solar heating of the ground. However, the effects of this heating process are captured through the implementation of the inlet temperature profile.

2.2.4 Solution Algorithm

Due to the pressure-momentum coupling of Eqns. 2.3 & 2.4, specialized numerical techniques are necessary to solve the system of PDEs. The present study uses the PIMPLE algorithm, a combination of the SIMPLE and PISO algorithms.

The SIMPLE algorithm [46] is an iterative algorithm that utilizes a predictor-corrector method to solve the coupled equations. The PISO algorithm [47] is a non-iterative algorithm that uses a similar predictor-corrector method to SIMPLE, but employs multiple corrector steps to satisfy mass conservation. The PIMPLE algorithm combines the two former algorithms by applying SIMPLE’s iterations to the multiple corrections performed in PISO. A flowchart of the algorithm can be seen in Figure 2.6.
The PIMPLE algorithm is considered very stable due to the multiple corrector (PISO) and outer (SIMPLE) loops. The PIMPLE algorithm can even be stable with a Courant number greater than one. However, allowing the Courant number to rise above one is not recommended as information on the transport of turbulence properties, temperature, pollutant concentration can be lost.
2.3 Mesh Generation

Quality mesh generation is a vital aspect to any CFD simulation. A set of best practice guidelines provided by COST [48] ensure CFD studies on the ABL use quality meshes. This section will discuss the guidelines to generate when modeling the ABL and the process used to generate the real urban domain using OpenFOAM’s *snappyHexMesh*. The process to create the idealized urban domain will not be explicitly discussed as it is a simplification of the real urban domain.

First, a target region is selected in which field data can be measured and used to validate the CFD results. In the present study, the Jones’ Island Water Reclamation Facility and surrounding area is selected because measurements can be taken near the facility and it is believed to be the only source of H$_2$S in the area.

Next, an area of interest within the target region is selected and the building information (e.g. footprint geometry, location, height, etc.) is recorded. The maximum building height, $H_{\text{max}}$, is used to scale the size of the full 3D domain. For the real urban domain, the height of chimneys and bridges are not considered for $H_{\text{max}}$ as they do not act as bluff bodies in the same fashion as buildings. However, all structures are considered when determining the height of the domain as all structures have the potential to induce artificial acceleration. The area of interest for the real urban domain is the Jones’ Island Water Reclamation Facility and portion of the Third Ward Neighborhood in Milwaukee, WI and can be seen in Figure 2.7.

The inlet of the domain is expected to be placed $5H_{\text{max}}$ ahead of the first building to prevent an artificial increase in pressure near the inlet. The lateral walls of the domain should be located $5H_{\text{max}}$ away from the sides of the area of interest and the top of the domain should extend $5H_{\text{max}}$ above the tallest structure. However, the lateral and vertical extensions may need to be increased
Figure 2.7: The building footprints (blue), area of interest (green box), and full domain (brown box).

to keep the blockage ratio, Eqn. 2.23, under 0.1 and ideally under 0.03,

\[
\text{Blockage Ratio} = \frac{A_{\text{Buildings}}}{A_{\text{Domain}}},
\]  

(2.23)

where \(A_{\text{Buildings}}\) is the projected area of the buildings and \(A_{\text{Domain}}\) is the cross sectional area of the domain. Maintaining a low blockage ratio is crucial in reducing artificial acceleration. The full domain area can be seen in Figure 2.7.

The distance from the area of interest to the outlet of the domain is dependent on the number of buildings within the study and the outlet conditions being used. The distance for a single building should be at least \(15H_{\text{max}}\) to allow the flow to redevelop behind the wake. When the area of interest contains multiple buildings, a shorter distance can be used. The present study uses a distance of \(8H_{\text{max}}\).
Building footprint geometries are determined using QGIS [49] and a building footprint shapefile for Milwaukee County [50]. Building footprints within the area of interest are decomposed into a set of overlapping rectangles as shown in Figure 2.8 and will be used with snappyHexMesh. Rectangles are used because snappyHexMesh can only handle simple geometry (e.g. boxes, spheres, cylinders) without 3D CAD files. The benefits of CAD modeling are negligible for the building geometry in this case as minor features are neglected due to the large size of the domain. The shapes overlap to prevent gaps from forming between the components of the building and snappyHexMesh can merge overlapping structures.

Figure 2.8: A single, complex building footprint geometry can be divided into simple, overlapping shapes.

Building geometry is imported into snappyHexMesh using a searchableSurfaceCollection. Using the decomposed building footprints, each rectangle is evaluated to determine the direction of $e_1$ and the length of $L_1$ and $L_2$ as shown in Figure 2.9. Only the direction of $e_1$ needs to be determined as $e_3$ is set parallel to the z-axis, therefore $e_2$ is set automatically such that $e_1 \times e_2 = e_3$. The length $L_3$ corresponds to the building height which has to be set manually. The heights of the builds are approximated by using the heights of known objects (e.g. doors, cars, signs, etc.) near the building. This approximation is performed using
Figure 2.9: A rotated building component with basis vectors $e_1$ and $e_2$ and side lengths $L_1$ and $L_2$.

A linear relation between the real size of the known object and the size of the known object in a photograph. Approximating lengths using this method lead to some inaccuracies caused by the camera’s focal length and distances between the building and the known object, but databases containing the heights of buildings are not readily available.

All permanent structures within the area of interest are included. Cars, tents, other non-permanent structures, and foliage are excluded from the mesh as they increase computational costs by adding additional cells or their positions cannot be verified during the measurement period.

The topography of the area is generated using 1-meter Digital Elevation Model (DEM) data from the United States Geographical Survey [51]. This data provides high resolution topography of the bare earth meaning buildings and foliage are excluded from the LIDAR measurements. The DEM data can be converted to a 3D STL file using the DEMto3D plugin for QGIS [52]. The STL output is used with snappyHexMesh to implement real topography in the real domain model.
The area of interest for the present study also includes the Daniel W. Hoan Memorial Bridge. This structure must be included as some field measurements are taken on top of the bridge. An STL file [53] is used to implement this feature. OpenFOAM has surface utilities to convert the scale of STL files and to change the position of the surfaces via linear translation.

The pollutant sources are created by overlaying aerial images in QGIS, as shown in Figure 2.10. Rectangles and circles are created to match the location and position of the water treatment beds and their geometries are exported. Within snappyHexMesh, the sources are treated as searchableDisks and searchablePlates placed at ground level.

![Figure 2.10: Image of Jones’ Island in QGIS used to identify pollutant source geometries. Pollutant sources are marked in pink and purple.](image)

The meshing process can proceed once the structure, source, and topography files are created. A background mesh is created using OpenFOAM’s blockMesh with dimensions that align with the best practice guidelines. The hexahedral mesh created by blockMesh is used by snappyHexMesh with the
included STL and *searchable* objects to create the urban domain. The mesh generation process can be broken into three distinct steps: castellation and cell removal, snapping, and layering.

During castellation, intersections between the background mesh and the included surface are located and used to refine cells. Figure 2.11 shows a brief overview of this cell refinement process. It should be noted that Figure 2.11 only shows cells that intersect a surface being refined. However, it is possible, and recommended, to refine cells within a specified distance of an intersection to assist with the snapping procedure and increase mesh resolution in areas with
high gradients. After refinement, excess cells are removed leaving a jagged mesh comprised exclusively of hexahedral cells as shown in Figure 2.12. Proper refinement during the castellation and removal step is crucial as the quality of the snapping and layering steps depend on the castellation resolution.

The snapping performed by \emph{snappyHexMesh} is an iterative process which alters the shape of cells near surface intersections to match the target geometry. Because cells are warped and altered, mesh quality can be greatly effected by the production highly skewed and non-orthogonal cells. Problematic cells are identified as having non-orthogonality greater than 65 and a skewness greater than 4. Unfortunately, there is no procedure available to alter the specific problematic cells, so both the castellation and snapping steps have to be repeated with increased refinement near the surfaces or more iterations to fix the skewed and non-orthogonal cells. The transition from a castellated to a snapped mesh can be seen in Figure 2.13.

Layering is the final step of the mesh refinement procedure and allows for greater refinement near no-slip surfaces in the domain. For standard turbulence wall functions, best practice guidelines recommend the first node to be placed at a distance of $30 \leq y^+ \leq 500$, where $y^+$ is the dimensionless distance normal
to the wall. Castellation can be used to achieve an optimal $y^+$ distance but significantly increases the number of cells within the domain as the process divides cells in all directions. Layering is used to refine near-wall cells in the normal direction only. Figure 2.14 shows the result of the layering step.

![Figure 2.14: A snapped mesh (left) gets additional layers (right) near wall boundaries to improve turbulence wall functions.](image)

With the mesh generation process complete, as shown in Figure 2.15, a mesh refinement study is performed to ensure results are independent of mesh resolution. The refinement study can be performed by altering characteristics of the *snappyHexMesh* process (i.e. castellation level, snapping iterations, layering) or by increasing the resolution of the background mesh. Often a combination of both is necessary to refine specific regions and increase mesh resolution without dramatically increasing the number of cells.

### 2.4 Directional Variability Method

As discussed in Chapter 1, the lateral dispersion predicted by CFD modeling is not necessarily representative of the physical processes of aerial pollutant dispersion. Fluctuations in wind speed and wind direction can alter the concentration of pollutants downwind of a source. The present study proposes the Directional Variability Method (DVM) to implement changes in wind speed and direction using readily available data from the Automated Surface Observing
System (ASOS).

The DVM allows the CFD simulation to periodically update the boundary conditions with new wind speeds and directions to better match the time period which field measurements were taken. Figure 2.16 shows the increase in temporal resolution by using ASOS 1-Minute wind data as opposed to typical National Weather Service (NWS) hourly wind data.

The simulations begin with a flow field that is at rest and is run until the domain reaches a stationary steady state. Once steady state has been achieved, the ASOS wind data is used to update the wind speed and direction of the inlets before running for 60 s of simulation time. The inlet and lateral domain boundaries are each given an atmBoundaryLayer class condition for $u$, $k$, and $\epsilon$. 
This specific class of boundary conditions acts as an inlet-outlet condition; the condition applies a profile using Eqns. 2.13-2.15 for the respective field when the flux is positive (into the domain); the condition applies a zero gradient to boundary when the flux is negative (out of the domain). As the boundary condition works for both inlets and outlets, only the wind speed and direction values need to be changed for the lateral boundaries. The OpenFOAM utility foamDictionary is used to change the speed and direction of the atmBoundaryLayer conditions.

The other fields (C, p, prgh, and T) cannot use the same boundary condition for inlet and outlet. The wind direction is used to determine which lateral boundary will be an inlet and which will be an outlet. No special treatment is needed when a lateral boundary changes from an inlet to an outlet, so the boundary condition can be changed immediately to zeroGradient. However, extra precaution is needed when changing a boundary from an outlet to an inlet. First, the current boundary values are saved from the field dictionary using
foamDictionary. This nonuniform list of values is used to initialize the respective inlet condition. Once all of the boundary conditions have been updated, the simulation is run for another 60 s before the boundary conditions are changed again. This process is continued until the measurement period ends. Once the DVM is complete, the results can be time averaged and compared to AERMOD. A flowchart of the process can be seen in Figure 2.17.

Figure 2.17: An overview of the Directional Variability Method.
CHAPTER 3

DISPERSION IN IDEALIZED SETTINGS

The first two cases to be presented in this work are the idealized settings: a flat field and an idealized urban environment. The purpose of these cases is twofold: test the DVM in a simplified environment and evaluate how DVM changes pollutant dispersion in regards to the standard method (i.e. single wind direction and speed). AERMOD is used as a low cost tool to validate the CFD results in the flat field and idealized urban environment.

For the flat field, AERMOD has been extensively tested and validated in similar, simple domains and is generally accurate within a factor of two [54]. However, less certainty can be placed with AERMOD in the idealized urban environment. The GPM has a building downwash algorithm to estimate the building’s effects on the plume, but the algorithm is empirically formulated and has shown mixed results depending on building configurations and source location [55]. Nonetheless, CFD results in the idealized urban environment are still compared to AERMOD as it provides a low cost alternative to other validation methods.

In the remainder of this chapter the geometry of each domain will be presented. Following will be an overview of the meteorological conditions used for the simulations and AERMOD. Then the results will be presented and discussed. Finally, the chapter will be summarized and key findings will be listed.

3.1 Domain Geometry

The flat field and idealized urban environment have very similar domain geometries. The two domains have the same domain dimensions of 580 m × 500 m × 200 m, and both domains have a single pollutant inlet with a
diameter of 10 m located 130 m from the inlet. Both inlets were given a pollutant mass fraction of $C = 1$ and an inlet velocity normal to the ground of $\bar{u}_z = 1 \text{ m s}^{-1}$.

The domains differ by the addition of buildings in the idealized urban environment. While the flat field contains nothing but a pollutant inlet, the idealized urban environment contains a $5 \times 5$ array of identical buildings. Each building has dimensions of $20 \text{ m} \times 20 \text{ m} \times 20 \text{ m}$. The buildings are arranged in a fashion such that each building is placed 20 m from its nearest neighbors. Thus, giving every urban canyon an Aspect Ration (AR) of 1:1 (i.e. the canyon length is equal to the canyon height). The flat field and idealized urban domain can be seen in Figures 3.1 and 3.2, respectively.

![Figure 3.1: The flat field domain and pollutant source (red)](image)

The concentrations are measured along five lines throughout the domain: two lines at pedestrian height (2 m) spanning the width of the domain to observe the lateral dispersion, one line running the length of the domain at pedestrian height to observe the decrease in pollutant concentration moving away from the
Figure 3.2: The idealized urban environment with buildings (red) and pollutant source (blue)

source, and two vertical lines to observe the vertical pollutant concentration profiles. The lines occur are the same locations in the flat field domain as well as the idealized urban environment; the lines can be seen in Figures 3.3 and 3.4, respectively.

Figure 3.3: The five sample lines within the flat field domain
Figure 3.4: The five sample lines within the idealized urban environment

3.2 Meteorology

The flat field was simulated using stable atmospheric conditions. The meteorological data was gathered from processed AERMET and can be seen in Table 3.1.

Table 3.1: Meteorological conditions used in the flat field domain

<table>
<thead>
<tr>
<th></th>
<th>Stable</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U_{ref}$ (m s$^{-1}$)</td>
<td>2.25</td>
</tr>
<tr>
<td>$z_0$ (m)</td>
<td>0.41</td>
</tr>
<tr>
<td>$z_{ref,U}$ (m)</td>
<td>2</td>
</tr>
<tr>
<td>$T_{ref}$ (K)</td>
<td>275.4</td>
</tr>
<tr>
<td>$z_{ref,T}$ (m)</td>
<td>10</td>
</tr>
<tr>
<td>$Q_s$ (W m$^{-2}$ s$^{-1}$)</td>
<td>-26.0</td>
</tr>
<tr>
<td>$L$ (m)</td>
<td>40.5</td>
</tr>
<tr>
<td>Date and Time</td>
<td>12/15/18 1900 CST</td>
</tr>
<tr>
<td>Location</td>
<td>General Mitchell Airport, Milwaukee, WI</td>
</tr>
</tbody>
</table>

The idealized urban environment was simulated in stable and neutral atmospheric conditions. A neutral atmosphere neither promotes nor inhibits
vertical motion. Again, the meteorological data was gathered from processed AERMET data and can be seen in Table 3.2

<table>
<thead>
<tr>
<th></th>
<th>Stable</th>
<th>Neutral</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U_{ref}$ (m s$^{-1}$)</td>
<td>2.25</td>
<td>5.50</td>
</tr>
<tr>
<td>$z_0$ (m)</td>
<td>0.41</td>
<td>0.9</td>
</tr>
<tr>
<td>$z_{ref,U}$ (m)</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>$T_{ref}$ (K)</td>
<td>275.4</td>
<td>298.8</td>
</tr>
<tr>
<td>$z_{ref,T}$ (m)</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$Q_s$ (W m$^{-2}$ s$^{-1}$)</td>
<td>-26.0</td>
<td>39.4</td>
</tr>
<tr>
<td>$L$ (m)</td>
<td>40.5</td>
<td>-1064.5</td>
</tr>
<tr>
<td>Date and Time</td>
<td>12/15/18 1900 CST</td>
<td>8/8/18 1200 CST</td>
</tr>
<tr>
<td>Location</td>
<td>General Mitchell Airport, Milwaukee, WI</td>
<td></td>
</tr>
</tbody>
</table>

It is worth noting that the neutral atmospheric conditions shown in Table 3.2 experiences much higher wind speeds than the stable atmospheric conditions. As the results are presented, the impact and significance of the higher wind speeds will be discussed.

3.3 Results

Both the flat field and idealized urban environment domains were simulated using the standard method and the DVM with a variety of turbulent Schmidt numbers. Because the results from the simulations are being compared to AERMOD, which produces an hour-averaged concentration, DVM simulations are run for 3600 s and then the concentrations are time averaged over the entire hour. Non-DVM simulations are run to a steady state and are do not need to be averaged.
3.3.1 Lateral Dispersion

The first results to be analyzed are the lines spanning the width of the domain; these lines showcase the lateral dispersion of the two CFD methods as compared to AERMOD. By observing the lateral dispersion, the DVM can be evaluated to determine whether it performs to meet its original purpose: to improve the lateral dispersion in CFD simulations.

The pollutant concentrations at 40 m downwind of the source in the flat field can be observed in Figure 3.5. It can be seen that the non-DVM simulations experience less lateral dispersion than AERMOD, as expected. The DVM simulations improve the lateral of pollutants to better match AERMOD, with the DVM simulation with \( Sc_t = 0.3' \) matching AERMOD almost exactly.

Moving downwind to 80 m behind the source in the flat field, pollutant concentrations can be seen in Figure 3.6. Again, it can be seen that the non-DVM simulations fail to capture the lateral dispersion seen by AERMOD whereas the
DVM simulations obtain lateral dispersion much more in line with AERMOD. However, unlike in Figure 3.5, the DVM simulation with $Sc_t = 0.3$ no longer matches AERMOD’s peak concentration; AERMOD is much closer to the DVM simulation with $Sc_t = 1.0$. This shift from matching $Sc_t = 0.3$ at 40 m downwind to matching $Sc_t = 1.0$ at 80 m downwind begins to indicate that the turbulent Schmidt number may not be constant throughout the domain.

Next, the idealized urban environment is observed at 40 m downwind. Pollutant concentrations in the stable atmosphere can be seen in Figure 3.7 while the concentrations in the neutral atmosphere can be seen in Figure 3.8.

From Figures 3.7 and 3.8, a few initial observations can be made. First, the non-DVM trials with large turbulent Schmidt numbers ($Sc_t \geq 1.0$) in the stable conditions and the AERMOD results in both meteorological conditions have shapes not consistent with expectations. Typically, an “M” shape is expected to form behind buildings. Second, significant asymmetry can be seen in the DVM simulations in stable conditions; the DVM simulations in the neutral atmosphere don’t show the same degree of asymmetry.
Figure 3.7: Pollutant concentrations 40 m downwind in the idealized urban environment in stable conditions

Figure 3.8: Pollutant concentrations 40 m downwind in the idealized urban environment in neutral conditions

This pronounced asymmetry seen in the stable conditions can be attributed to lower reference wind speed. Higher wind speeds are correlated to a decrease in wind direction variation [31] meaning lower wind speeds tend to
have higher variation of direction. Therefore the asymmetry of the DVM results in the stable conditions are a result of low wind speeds.

Observing Figures 3.7 and 3.8 to evaluate the effectiveness of the DVM, it can be seen that the non-DVM trial with Sc_t = 0.3 appears to match AERMOD relatively well, albeit with large discrepancies in the maximum concentrations. In the stable conditions, DVM simulation with Sc_t = 0.3 appears to deviate significantly from AERMOD due to the asymmetry (i.e. high directional variation). The DVM simulation with Sc_t = 0.3 in neutral conditions appears to match AERMOD relatively well.

Moving to 80 m downwind of the pollutant source in the idealized urban environment, pollutant concentrations from the stable conditions can be seen in Figure 3.9 and concentrations in the neutral conditions can be seen in Figure 3.10.

Figure 3.9: Pollutant concentrations 80 m downwind in the idealized urban environment in stable conditions

First, observing the stable case in Figure 3.9 shows several things: the non-DVM simulations with Sc_t ≥ 1.0 now have a central “W” shape which does
not align with expectations, the DVM simulation with $Sc_t = 0.3$ does not match AERMOD as well as previously shown at 40 m downwind (Fig. 3.7), and the asymmetry shown by the DVM simulations has increased significantly.

The non-DVM simulation with $Sc_t = 0.3$ moved out of agreement with AERMOD at 80 m downwind, and the non-DVM simulation with $Sc_t = 1.0$ does not entirely agree with AERMOD either. Likely, a turbulent Schmidt number between 0.3 and 1.0 would better match AERMOD. This further indicates that $Sc_t$ is likely not constant throughout the domain. Some work has been done to study the variability of the turbulent Schmidt number, but no agreed upon formulation exists yet [43].

![Figure 3.10: Pollutant concentrations 80 m downwind in the idealized urban environment in neutral conditions](image)

Observing pollutant concentrations in the neutral atmospheric conditions in Figure 3.10 begins to show better agreement between all of the CFD simulations and AERMOD. Both CFD simulations with $Sc_t = 0.3$ show lower pollutant concentrations than AERMOD and slightly wider lateral dispersion.
The DVM simulations with \( \text{Sc}_t = 1.0 \) and \( \text{Sc}_t = 1.2 \) show good agreement with AERMOD in terms of concentrations near the center \((-30 \leq y \leq 30)\). However, the two DVM simulations with large turbulent Schmidt numbers show a wider lateral dispersion than AERMOD.

From the lateral dispersion results shown, the following observations have been made:

- The DVM improves CFD results in flat fields.
- The turbulent Schmidt number may not be constant throughout a domain.
- At low wind speeds, the DVM shows significant asymmetry due to the high variability of wind direction, making the DVM appear inaccurate when compared to AERMOD’s empirical downwash algorithm.
- At high wind speeds, the DVM appears to be accurate to some extent.

### 3.3.2 Centerline of the Domain

Next, the line running the length of the domain is analyzed. This line showcases the decrease of pollutant concentrations when moving downwind from the source. The pollutant concentrations from the flat field domain can be seen in Figure 3.11.

The results shown in Figure 3.11 present a much clearer indication of a variable turbulent Schmidt number. Throughout the domain, AERMOD predicts concentrations between the DVM simulations with \( \text{Sc}_t = 0.3 \) and \( \text{Sc}_t = 1.0 \). However, the spacing between AERMOD and the two DVM simulations does not remain constant.

Moving the the idealized urban environment, the concentrations in the stable atmosphere can be seen in Figure 3.12 and the concentrations in the neutral atmosphere can be seen in Figure 3.13.
The pollutant concentrations along the centerline of the stable atmosphere in the idealized urban environment shows a few things worth noting: the non-DVM simulations with large turbulent Schmidt numbers ($Sc_t \geq 1.0$) show
increasing pollutant concentrations when moving from the leeward (upwind) to the windward (downwind) wall of the urban canyon which is not consistent with expectations [56], AERMOD predicts higher concentrations than all of the CFD simulations throughout most of the domain, and the turbulent Schmidt number appears to have minimal influence on the DVM simulations.

The first two points were worth noting, but the third point is worth a larger discussion for the purpose of understanding the importance the turbulent Schmidt number in the DVM. While not explicitly discussed earlier, Figures 3.7 and 3.9 showed very small differences between the three DVM simulations, meaning the DVM is showing low sensitivity to changes in $Sc_t$ at low wind speeds. The DVM’s low sensitivity to $Sc_t$ at low wind speeds continues to be shown in Figure 3.12.

The understanding the DVM’s sensitivity is useful to better understand the results when modeling real urban environments with field measurements. Because pollutant concentrations can only be measured at a finite number of points in real urban environments, it is beneficial to know if inaccuracies in simulated results are a product of an ineffective model or incorrect parameterizations.

Moving to the neutral atmosphere, pollutant concentrations along the domain centerline can be seen in Figure 3.13. Again, it can be seen that AERMOD concentrations are higher than all of the CFD simulations. However, it is important to use Figures 3.8 and 3.10 as reference as they also showed AERMOD predicted concentrations much higher than the CFD simulations, but the lateral spread predicted by AERMOD was similar to the DVM simulations with $Sc_t = 1.0$ and $Sc_t = 1.2$ at 80 m downwind.

In the neutral atmosphere, with higher wind speeds and lower directional variation, it can be seen that more significant differences exist between the DVM
Figure 3.13: Pollutant concentrations along the domain centerline in the idealized urban environment in a neutral atmosphere simulations. This indicates that the DVM has a higher sensitivity to the turbulent Schmidt number at higher wind speeds. This claim can also be seen through the governing equations. Higher wind speeds correspond to more turbulence and higher values of \( k \), thus a larger value for turbulent viscosity. With larger values of \( \nu_t \), changes in \( \text{Sc}_t \) have a greater effect on the turbulent diffusivity.

From the centerline results shown, the following observations have been made:

- The turbulent Schmidt number is likely not constant throughout the domain.
- The DVM appears to be less sensitive to changes in \( \text{Sc}_t \) at lower wind speeds.

### 3.3.3 Vertical Concentration Profiles

The last sample lines to observe are the vertical pollutant concentration profiles. Reviewing the vertical lines assists with understanding the vertical
extent of the pollution. The vertical pollutant profile 40 m downwind of the
source in the flat field can be seen in Figure 3.14. Observing the vertical pollutant
concentration profile shows that AERMOD predicts the plume to remain near the
surface, albeit with some vertical lift. However, the vertical extent of the
AERMOD plume appears to match the non-DVM simulation with Sc ≤ 1.0,
ending at a height near 20 m.

In general, it appears that DVM simulations have plumes that extend
higher than their non-DVM counterparts (e.g. the plume from the non-DVM
simulation with Sc = 0.3 reaches a height of ≈ 30 m whereas the the plume from
the DVM simulation with Sc = 0.3 reaches a height of ≈ 45 m). This can be
attributed to the variation of wind speed in the DVM simulations.

Higher wind speeds correspond lower pollutant concentrations and
plumes with less vertical and lateral spread. Because the DVM simulations
include wind speeds that are less than the reference wind speed there are

![Figure 3.14: Vertical profile of pollutant concentration 40 m downwind from source in the flat field](image)
intervals of time with much wider plumes, thus the DVM shows a higher vertical extent of the plume.

The vertical pollutant concentration profile 80 m downwind of the source in the flat field can be seen in Figure 3.15. At this distance downwind, the CFD

![Figure 3.15: Vertical profile of pollutant concentration 80 m downwind from source in the flat field](image)

plume’s appear to be much more in line with AERMOD’s. The pollutant concentrations predicted by AERMOD appear to match the non-DVM simulation $Sc_t = 1.2$ and the DVM simulation with $Sc_t = 0.3$. In addition, AERMOD’s vertical extent of the plume matches the non-DVM simulation with $Sc_t = 0.3$ with some accuracy.

Turning to the idealized urban environment, the vertical pollutant concentration profiles 40 m downwind of the source can be seen in Figures 3.16 and 3.17 for the stable and neutral atmospheres, respectively.

In the stable atmosphere (Fig. 3.16), it can be seen that AERMOD differs in shape from most of the CFD simulations. AERMOD shows a decrease in
Figure 3.16: Vertical profile of pollutant concentration 40 m downwind from source in the idealized urban environment in a stable atmosphere.

Pollutant concentration with height whereas the CFD simulations (DVM and non-DVM) with large turbulent Schmidt numbers ($S_{ct} \geq 1.0$) show a peak at the building height.

The large difference in vertical pollutant concentration profiles can make it difficult qualitatively evaluate and examine the CFD results in relation to AERMOD. Because of this difficulty, it is worth noting again that AERMOD’s building downwash is empirically formulated and may not provide accurate pollutant concentrations in all settings.

In the neutral atmosphere, differences between AERMOD and CFD become even larger. Figure 3.17 show the vertical profile of pollutant concentrations in a neutral atmosphere. A few observations worth noting: AERMOD predicts higher pollutant concentrations than all of the CFD simulations at nearly all heights, AERMOD shows a much taller plume than all of the CFD simulations, and the higher wind speeds changed the shape of the CFD vertical pollutant concentration profiles.
Figure 3.17: Vertical profile of pollutant concentration 40 m downwind from source in the idealized urban environment in a neutral atmosphere

Moving to 80 m downwind in the stable atmosphere, Figure 3.18 shows significant differences between the CFD simulations and AERMOD. AERMOD continues to show a decrease in pollutant concentrations with height, but the
plume predicted by AERMOD is much taller than all of the CFD simulations. However, it is worth noting that the DVM’s sensitivity to the turbulent Schmidt number appears to be more significant at elevations above pedestrian height. By understanding this aspect of the DVM’s sensitivity, the results in the real urban domain can be better understood and interpreted.

Lastly, at 80 m downwind in the neutral atmosphere, Figure 3.19. Again,

Figure 3.19: Vertical profile of pollutant concentration 80 m downwind from source in the idealized urban environment in a neutral atmosphere

AERMOD predicts pollutant concentrations that are much higher than the CFD simulations at all elevations. Because of these vast differences, it is challenging to make any certain claims about the accuracy of the CFD simulations or the usefulness of the DVM. Better validation techniques, such as controlled measurements, could provide more certainty and allow for more useful comparisons with the CFD simulations.
While on the topic of the vertical profiles, it is worth noting that unstable meteorological conditions were not studied in either the flat field or the idealized urban environment and neutral conditions were not studied in the flat field. This is a result of CFD not being able to capture the plume rise predicted by AERMOD. The vertical motion within unstable meteorological conditions is caused by thermals, or updrafts, which large and very turbulent structures [57]. Other works that have evaluated pollutant dispersion used a hybrid RANS-LES approach [58] or utilized LES [59]. By utilizing LES, the turbulent fluctuations caused by buoyancy could be captured and lead to the plume rise. RANS is not able to capture these turbulent fluctuations, only the average motion, thus the plume rise is not captured.

From the vertical pollutant concentration profiles show, the following observations have been made:

- CFD simulations agree to some extent with AERMOD. However, some discrepancies still exist in the flat field.
- AERMOD and CFD produce vastly different vertical pollutant concentration profiles in the idealized urban environment. The quality of AERMOD in dense urban areas is uncertain because of the empirical formulation.
- The DVM is more sensitive to changes in $Sc_t$ with low winds near the top of a structure’s wake.

3.4 Quantitative Assessment of the DVM’s Sensitivity

Throughout the discussion thus far, assessment of the DVM has been primarily qualitative. However, some quantitative assessment of the DVM can be performed in the idealized urban domain. Through this qualitative assessment,
the sensitivity of the DVM to the turbulent Schmidt number can be better understood.

To perform this assessment, the Fractional Bias ($FB$) of each DVM simulation is calculated along the lateral sample lines with respect to the DVM simulation with $Sc_t = 0.3$ as a reference. For this assessment, the $FB$ is calculated using Eqn. 3.1,

$$FB = \frac{\sum_i(C_{i,Sc_t=0.3} - C_{i,Sc_t=j})}{0.5\sum_i(C_{i,Sc_t=0.3} + C_{i,Sc_t=j})}$$  (3.1)

where $C_{i,Sc_t=0.3}$ refers the concentration at a sample point in the DVM simulation with $Sc_t = 0.3$ and $C_{i,Sc_t=j}$ is the concentration at the same sample point in a DVM simulation where $j = 1.0$ or 1.2. However, the value for the $FB$ should not be thought of in the traditional sense, which is discussed further in Section 4.4, and should be viewed as a metric to understand the sensitivity of the DVM to changes in the turbulent Schmidt Number. The values for the $FB$ can be seen in Table 3.3.

Table 3.3: Sensitivity analysis of the DVM to changes in the turbulent Schmidt number

<table>
<thead>
<tr>
<th></th>
<th>Stable</th>
<th>Neutral</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$Sc_t = 1.0$</td>
<td>$Sc_t = 1.2$</td>
</tr>
<tr>
<td>40 m Downwind</td>
<td>$-0.661$</td>
<td>$-0.679$</td>
</tr>
<tr>
<td>80 m Downwind</td>
<td>$-0.357$</td>
<td>$-0.374$</td>
</tr>
</tbody>
</table>

From the values shown in Table 3.3, it can been seen that changes to the turbulent Schmidt number tend to have a greater impact on the lateral dispersion in the neutral atmospheric conditions (i.e. higher wind speed) as they have a more negative value. Thus, the quantitative assessments on the sensitivity of the DVM to the turbulent Schmidt number were correct.
3.5 Summary

Pollutant dispersion was simulated in two idealized domains: a flat field and an idealized urban environment. Both domains were simulated in stable meteorological conditions and the idealized urban environment was also simulated in neutral meteorological conditions. Both domains were validated using AERMOD, albeit with less certainty in with the idealized urban environment domain. An overview of the flat field and idealized urban environment results presented can be seen in Tables 3.4 and 3.5.

Table 3.4: Summary of the flat field pollutant dispersion simulations

<table>
<thead>
<tr>
<th>CFD Simulation</th>
<th>Flat Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-DVM with $Sc_t = 0.3$</td>
<td>Under predicts lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Under predicts concentrations</td>
</tr>
<tr>
<td></td>
<td>Matches vertical extent of plume</td>
</tr>
<tr>
<td>non-DVM with $Sc_t = 1.0$</td>
<td>Under predicts lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Over predicts peak concentrations</td>
</tr>
<tr>
<td></td>
<td>Matches vertical extent of plume</td>
</tr>
<tr>
<td>non-DVM with $Sc_t = 1.2$</td>
<td>Under predicts lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Over predicts peak concentrations</td>
</tr>
<tr>
<td></td>
<td>Matches vertical extent of plume</td>
</tr>
<tr>
<td>DVM with $Sc_t = 0.3$</td>
<td>Accurately predicts lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Accurately predicts concentrations 40 m downwind</td>
</tr>
<tr>
<td></td>
<td>Slightly over predicts vertical extent of plume</td>
</tr>
<tr>
<td>DVM with $Sc_t = 1.0$</td>
<td>Accurately predicts lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Accurately predicts concentrations 80 m downwind</td>
</tr>
<tr>
<td></td>
<td>Slightly over predicts vertical extent of plume</td>
</tr>
<tr>
<td>DVM with $Sc_t = 1.2$</td>
<td>Not performed</td>
</tr>
</tbody>
</table>
Table 3.5: Summary of the idealized urban environment dispersion simulations

<table>
<thead>
<tr>
<th>CFD Simulation</th>
<th>Stable Atmosphere</th>
<th>Idealized Urban Environment Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Stable Atmosphere</td>
<td>Neutral Atmosphere</td>
</tr>
<tr>
<td>non-DVM with $S_C = 0.3$</td>
<td>Agrees with AERMOD</td>
<td>Agrees with AERMOD</td>
</tr>
<tr>
<td></td>
<td>Peak concentration is off</td>
<td>Peak concentration is off</td>
</tr>
<tr>
<td>non-DVM with $S_C = 1.0$</td>
<td>Unexpected lateral dispersion shape</td>
<td>Less lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Unexpected behavior in canyon</td>
<td>Lower peak concentrations</td>
</tr>
<tr>
<td>non-DVM with $S_C = 1.2$</td>
<td>Unexpected lateral dispersion shape</td>
<td>Less lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Unexpected behavior in canyon</td>
<td>Lower peak concentrations</td>
</tr>
<tr>
<td>DVM with $S_C = 0.3$</td>
<td>Asymmetric with significant lateral dispersion</td>
<td>Over predicts lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Significantly lower peak concentrations</td>
<td>Significantly lower peak concentrations</td>
</tr>
<tr>
<td>DVM with $S_C = 1.0$</td>
<td>Asymmetric with significant lateral dispersion</td>
<td>Similar lateral dispersion to AERMOD</td>
</tr>
<tr>
<td></td>
<td>Significantly lower peak concentrations</td>
<td>Slightly lower peak concentrations</td>
</tr>
<tr>
<td>DVM with $S_C = 1.2$</td>
<td>Asymmetric with significant lateral dispersion</td>
<td>Agrees with AERMOD’s lateral dispersion</td>
</tr>
<tr>
<td></td>
<td>Significantly lower peak concentrations</td>
<td>Slightly lower peak concentrations</td>
</tr>
</tbody>
</table>
CHAPTER 4

MODELING HYDROGEN SULFIDE EMISSIONS FROM THE JONES’ ISLAND WATER RECLAMATION FACILITY

The final, and largest, test case of this work is modeling the dispersion of H$_2$S from the Jones’ Island Water Reclamation Facility (WRF) located in Milwaukee, WI. H$_2$S has a distinct rotten-egg smell is produced as a result of micro-organisms breaking down organic matter at the WRF; this gas can escape the water as it sits in the water holding vessels. Jones’ Island is located south of downtown Milwaukee and neighbors the Third Ward and Harbor View neighborhoods, both of which contain a mixture of commercial and residential buildings. The rest of this chapter will present and discuss the measurement techniques, meteorological conditions, geometry and mesh convergence, and simulation results.

4.1 Field Measurements

H$_2$S concentrations were measured using a Sniffer4D mounted to the roof of a moving car by a collaborator in the Department of Civil, Construction, and Environmental Engineering at Marquette University as a part of a larger air quality research project. Measurements taken for this case were taken on November 11$^{th}$, 2020 between 3:37pm and 4:31pm CST (N. Hay, personal communication, March, 2021).

The Sniffer4D has an on-board GPS to record the latitude, longitude, elevation, and time of the measurement. The device has a sampling rate of 1 Hz which provides sufficient data to use for CFD validation.

Figure 4.1 shows the route and measurement locations within the area of interest. The route enters the area of interest through the northwest corner and continues along the north side of Kinnickinnic River before turning around at the
Figure 4.1: Measurement route through the area of interest

edge of lake Michigan. The path continues along the same route before exiting the north side of the area of interest. The route comes back through area of interest, traveling south along the Hoan bridge and finished by traveling north on the Hoan Bridge. A total of 378 samples are located within the area of interest over a period of 11 minutes.

4.2 Meteorological Data

As previously mentioned, the field measurements were taken on November, 9th, 2020. The atmosphere was neutrally stable during the sample period and was experiencing high wind speeds coming from the south. The meteorological data for the time of measurements was acquired through the
National Oceanic and Atmospheric Administration (NOAA)’s Integrated Surface Database [60] and ASOS 1-Minute data [61]. The hourly data is presented in Table 4.1.

Table 4.1: Meteorological conditions for the Jones’ Island simulation

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U_{ref}$ (m s$^{-1}$)</td>
<td>7.91</td>
</tr>
<tr>
<td>$z_0$ (m)</td>
<td>0.9</td>
</tr>
<tr>
<td>$z_{ref,U}$ (m)</td>
<td>2</td>
</tr>
<tr>
<td>$T_{ref}$ (K)</td>
<td>297.0</td>
</tr>
<tr>
<td>$z_{ref,T}$ (m)</td>
<td>10</td>
</tr>
<tr>
<td>$Q_s$ (W m$^{-2}$ s$^{-1}$)</td>
<td>47.9</td>
</tr>
<tr>
<td>$L$ (m)</td>
<td>-2387.1</td>
</tr>
</tbody>
</table>

The meteorological data was recorded at General Mitchell International Airport in Milwaukee, WI. The airport is approximately 8.5 km away from the area of interest. Because this distance is not insignificant, there exists some uncertainty whether the wind conditions recorded at the airport are the same that were experienced at Jones’ Island. However, without meteorological monitoring at the Jones’ Island facility the true wind conditions cannot be verified. Therefore, the data recorded at the Milwaukee airport must be used as it is the closest meteorological measurement location.

4.3 Geometry and Convergence Study

The constructed domain has dimensions of 1106 m $\times$ 1403 m $\times$ 250 m. $H_{max}$ is set at 35 m while the tallest structure reaches a height of approximately 70 m. Within the domain all of the buildings and structures, the ground, and bodies of water are treated as stationary walls. The full domain with dimensions can be seen in Figure 4.2. The emission rate of the sources were determined using information from the Milwaukee Metropolitan Sewerage District and the EPA’s
compilation of air emission factors [62]. The sources were all given a mass flux rate of $8.19 \times 10^{-4}$ g m$^{-2}$ s$^{-1}$ [63].

To verify the quality of the mesh, a convergence study was performed on the velocity field to ensure results were independent of the mesh resolution. The velocity for the mesh convergence study was set to 10.8 m s$^{-1}$, the highest velocity during the sampling time. Four different mesh sizes were used ranging from 2.07M to 4.29M cells. Velocity profiles were sampled immediately downwind of two of the tallest buildings. The lines were chosen as they are located near the sources and are very turbulent areas. The mesh statistics can be seen in Table 4.2 and the converging velocity profiles can be seen in Figures 4.3 and 4.4.
Table 4.2: Mesh statistics for grid convergence

<table>
<thead>
<tr>
<th></th>
<th>Mesh 1</th>
<th>Mesh 2</th>
<th>Mesh 3</th>
<th>Mesh 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Cells</td>
<td>2.07 M</td>
<td>2.54 M</td>
<td>3.46 M</td>
<td>4.29 M</td>
</tr>
<tr>
<td>Min. Cell Volume (m$^3$)</td>
<td>0.077</td>
<td>0.053</td>
<td>0.021</td>
<td>0.012</td>
</tr>
</tbody>
</table>

Figure 4.3: Velocity profile of 4 meshes behind a building

4.4 Results

The sampling period was simulated using the DVM with two values for the turbulent Schmidt number, 0.3 and 1.2. These two values were chosen to provide an upper and lower limit to the pollutant concentration at each sample location. Furthermore, the simulated data will not be time averaged as the field measurements are taken at specific points in time. Instead the CFD concentrations are sampled at measurement time rounded to the nearest 30 seconds.

In addition, the simulated and measured pollutant concentrations will be analyzed quantitatively. Three commonly used metrics in air quality modeling will be used to evaluate the performance of the simulations. These metrics are the
fractional bias ($FB$), the normalized mean square error ($NMSE$), and the fraction of predictions within a factor of two ($FAC2$) [64]. The equations for these metrics can be seen in Eqns. 4.1-4.3.

\[
FB = \frac{\sum_i (C_{i,Field} - C_{i,CFD})}{0.5 \sum_i (C_{i,Field} + C_{i,CFD})} \quad (4.1)
\]

\[
NMSE = \frac{1}{N} \sum_i \frac{(C_{i,Field} - C_{i,CFD})^2}{C_{i,Field} C_{i,CFD}} \quad (4.2)
\]

\[
FAC2 = \frac{1}{N} \sum_i F_i \quad (4.3)
\]

where

\[
F_i = \begin{cases} 
1, & 0.5 \leq \frac{C_{i,Field}}{C_{i,CFD}} \leq 2 \\
0, & \text{else}
\end{cases}
\]

and $N (= 378)$ is the number of samples, the subscript $i$ is the sample index, $C_{i,Field}$ is the field measurement, and $C_{i,CFD}$ is the CFD prediction.

A perfect model would see $FB = NMSE = 0$ and $FAC2 = 1.0$. However, perfect models do not exist so some margin of error is allowed in these metrics. The literature suggests [64, 65] that good evaluation metrics for a model are:
\[ |FB| \leq 0.3, \text{NMSE} \leq 4, \text{and FAC2} \geq 0.5. \] However, those standards were set with rural and simple domains in mind. For urban environments, it has been proposed that \[ |FB| \leq 0.67, \text{NMSE} \leq 6, \text{and FAC2} \geq 0.3 \] can be acceptable [29, 66].

Of the quality metrics, FAC2 is the most robust as it is not thrown off by outliers. The NMSE provides information on the systematic and random errors that occur in a simulation, but it is heavily influenced by outliers. The FB can provide context on the model’s tendency to under or over predict pollutant concentrations; however, the FB is also sensitive to outliers.

The pollutant concentrations can be seen in Figure 4.5 and the metrics can be reviewed in Table 4.3 Similar to the idealized domains, decreasing the value of \( \text{Sc}_t \) also decreases the pollutant concentration at each sample point.
Table 4.3: Meteorological conditions for the Jones’ Island simulation

<table>
<thead>
<tr>
<th></th>
<th>FB</th>
<th>NMSE</th>
<th>FAC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-DVM with $S_{Ct} = 0.3$</td>
<td>0.38</td>
<td>15.15</td>
<td>0.58</td>
</tr>
<tr>
<td>non-DVM with $S_{Ct} = 1.2$</td>
<td>-0.381</td>
<td>32.13</td>
<td>0.26</td>
</tr>
<tr>
<td>DVM with $S_{Ct} = 0.3$</td>
<td>0.14</td>
<td>72.34</td>
<td>0.49</td>
</tr>
<tr>
<td>DVM with $S_{Ct} = 1.2$</td>
<td>-0.74</td>
<td>579711.28</td>
<td>0.17</td>
</tr>
</tbody>
</table>

Observing the results present in Figure 4.5, it can be seen that the measured concentrations fluctuate significantly less than the CFD results, ranging from $244 - 261 \ \mu g \ m^{-3}$. While a large portion of the simulated and measured concentrations do not match, there are a few trends worth noting. First, a dip in pollutant concentration is seen in all of the simulations and the field measurements at the sample points ranging from 100 – 200. This range of sample points corresponds to the turn around point in the Third Ward. The dip occurs because the vehicle leaves the bulk of plume (decrease from points 100-150) before turning around and going back into the plume (increase from points 150-200).

A second trend that is similar to all simulations and the field data is the decrease in concentration from the point 225 – 275 which corresponds to the portion of the route driving west along the Kinnickinnic River. The decrease is cause by the vehicle exiting the plume as the wind was blowing towards the north and northeast. Although, it should be noted that the magnitude of these trends differs significantly between the simulations and the field measurements.

It can be seen in Table 4.3 that the non-DVM simulation with $S_{Ct} = 0.3$ performed the best of the CFD simulations with the highest $FAC2$ and lowest $NMSE$ while the DVM simulation with $S_{Ct} = 1.2$ performed the worst with less than 20% of samples points falling within a factor of two of the measured concentration.
However, there are two large discrepancies between the CFD simulations and the measurements seen in the results: the CFD simulations experience large fluctuations whereas the measurements show a near constant value across all of the sample points and CFD sees areas without any pollutant concentrations.

The points with near-zero values in the CFD simulations can be attributed to the wind direction. During the time period the measurements were taken, the wind was not blowing towards the northwest. Therefore, points to the north and northeast received more of the pollution. However, this does not explain the near constant concentrations seen in the measurements. The near constant values may be caused by a background concentration of H$_2$S.

Originally, it was assumed that the Jones’ Island WRF was the only major source of H$_2$S in the area as other industrial source of H$_2$S (i.e. livestock farms, natural gas and oil refineries, etc.) are not located in the region [67]. However, the comparing the measured results to the CFD indicates that this original assumption may not be correct. Other potential sources could include the surrounding lakes and rivers. One potential solution to this problem could be implementing a background concentration of H$_2$S in the domain. It is believed that background concentrations of H$_2$S can range from 0.11 – 0.33 ppb ($\approx 100 – 300 \ \mu g \ m^{-3}$) [68]. This range of background concentrations could account H$_2$S for the concentrations measured. As for the performance of the DVM, it can be seen that the proposed method was able to meet two of the three metric requirements for air quality modeling in urban environments. Although, the DVM underperforms when compared to the standard method in this real urban domain. There are a few changes that could be made to improve the performance of the DVM.

First, a longer sampling period should be utilized. While the wind speed and direction does fluctuate over a period of time, the sampling period in this
study only lasted for 11 minutes. Because of the high wind speeds, the wind
direction did not fluctuate significantly. Thus, the potential benefits of accounting
for the varied wind direction were not fully utilized.

Second, time averaged stationary measurement data should be utilized. In
the present study, measurements were taken atop a moving vehicle. While this is
a novel approach to air quality monitoring, it is not consistent with the standard
approach of measuring data at a single location over a period of time. The
potential effects of taking measurements on top of a moving vehicle are not well
studied which introduces significant uncertainty to the field measurements.
Using a stationary measurement device could remove any potential measurement
error or uncertainty induced by the moving vehicle.

Third, the emission rates have some degree of uncertainty to them. The
value utilized for the emission rate is considered to be the average for the entire
facility. However, each source could emit H\textsubscript{2}S at different rates which would
impact the simulation. Thus, better estimations, or measurements, of the emission
rates could provide better results and increase the certainty in the simulation.

Lastly, the DVM should start well before the sampling period. In the
present study the DVM ran for 60 s before the sampling period began. Running
the DVM for a longer period before the sampling period would include historical
effects of the pollutant dispersion. These historical effects would include the
transport of lingering pollutants previously trapped at a location caused by the
wind direction.

4.5 Summary

The dispersion of H\textsubscript{2}S from the Jones’ Island Water Reclamation Facility
was modeled using CFD. Real topography and building layouts were
implemented to assess the pollutant dispersion. The CFD simulations were
performed using the standard method and the DVM. The standard method outperformed the DVM with a $FAC_2$ of 0.58. However, measured pollutant concentrations remained nearly constant throughout the area of interest, indicating that unknown sources likely lead to the measured results shown in Figure 4.5.
CHAPTER 5

CONCLUSION

Aerial pollutant dispersion in an urban environments was simulated using CFD. The DVM was developed to improve the lateral dispersion in CFD simulations, evaluated in two idealized domains, and tested in a real urban environment with topography. The idealized domains consisted of a flat field and an idealized urban environment, and CFD simulations were validated using AERMOD, an empirically formulated Gaussian Plume Model developed by the Environmental Protection Agency. The real domain consisted of the Jones’ Island Water Reclamation Facility and the surrounding area, and the CFD simulations were validated using field measurements taken atop a moving vehicle.

The DVM was qualitatively assessed in the idealized domains by comparing the results to AERMOD. In the flat field, the DVM improved the lateral dispersion of pollutant concentrations. AERMOD and the CFD results also indicated that the turbulent Schmidt number may not be constant throughout the domain.

In the idealized urban environment the standard method for simulating pollutant transport with CFD appeared to outperform the DVM as the DVM showed substantial asymmetry caused by the wind variation. However, the performance of the two methods was assessed using AERMOD’s empirically formulated downwash algorithms; therefore, the accuracy of the results used for validation is uncertain. Although, CFD results from the idealized urban environment could be used to determine the sensitivity of the DVM and it was seen that the DVM was less sensitive to changes in the turbulent Schmidt number at low wind speeds.
The real urban domain with topography included field measurements which provided the highest level of certainty for validation. CFD simulations were performed using the standard method and the DVM to model the pollutant dispersion. Based off quantitative metrics, the standard method with $\text{Sc}_t = 0.3$ performed the best with nearly 60% of simulated sample points falling within a factor of two of the measured pollutant concentrations. The DVM was able to produce metric values within the ideal range for pollutant dispersion models in urban environments, but did not meet the metric values set by the standard method.

5.1 Future Work

CFD modeling of aerial pollutant dispersion has the potential to provide high spacial and temporal resolution of pollutant concentrations in urban environments. However, validating CFD results can be a challenge as empirically formulated models may not be accurate in all settings and field measurements include pollutant concentrations attributed to the source of interest and background pollutant concentrations. The background concentration can be difficult to determine as there is no method to be certain of the number of natural and anthropogenic sources in the surrounding areas and the emission rates. In addition, CFD simulations can only be performed in a fraction of meteorological conditions as RANS simulations with passive pollutant transport cannot capture the predicted plume rise caused by thermal instability.

Together, the problems encountered provide a number of potential areas of future research. First, AERMOD’s building downwash algorithm could be improved upon through a combination of scaled testing, field measurements, and numerical simulations. AERMOD is an open source software and some research studies have developed modifications for AERMOD [69]. By performing
additional numerical and experimental studies on building downwash in dense urban areas, the AERMOD algorithms can be improved and lead to less uncertainty and higher accuracy in the results.

Second, research into the variability of the turbulent Schmidt number should be investigated. It was shown that $\text{Sc}_t$ is not constant throughout a domain and can vary in simple geometries. Utilizing a constant turbulent Schmidt number requires multiple independent simulations to be performed to find the optimal value for $\text{Sc}_t$. Investigating methods for a variable turbulent Schmidt number could result in more accurate CFD simulations without requiring multiple simulations.

Third, controlled measurements in urban, or urban-like, environments can be performed to provide more certain field measurements. Many pollutants that have an anthropogenic source are either emitted by numerous sources and cannot be tracked to a single location (i.e. $\text{CO}_2$ from vehicle exhaust) or emitted from natural sources as well. The large number of potential sources can lead to an inherent background concentration. Utilizing a tracer gas without natural or anthropogenic sources could improve the certainty in the field measurements for validation.

Fourth, investigations into the effects of rivers and lakes on pollutant transport should be performed. In the Jones’ Island domain, two rivers and Lake Michigan were included in the modeling domain but were treated as stationary walls instead of moving bodies of water. Understanding how these bodies of water effect the transport and deposition of pollutants can help increase the certainty of the simulations.

Lastly, methods to model the plume rise in convective meteorological conditions should be investigated. As previously mentioned, it is believed that LES would be able to capture the turbulent fluctuations that lead to the formation
of thermals. However, LES will make CFD simulations more computationally expensive. Therefore, it is worth investigating other less costly methods to capture the phenomena. Utilizing multi-species flow solver or parameterizing the passive scalar transport equation are two routes worth investigating.
REFERENCES


APPENDIX A

OPENFOAM CODE FOR MODIFIED SOLVER

The buoyantBoussinesqPimpleFoam solver in OpenFOAM v5 was modified to include a passive scalar transport equation. The buoyantBoussinesqPimpleFoam.C, readTransportProperties.H, and createField.H files were modified. Included are the modified files.

A.1 buoyantBoussinesqPimpleWithTurbulentScalarTransportFoam.C

\*-------------------------------------------------------*/

ACKNOWLEDGEMENT

OPENFOAM(R) is a registered trade mark of OpenCFD Limited, producer and distributor of the OpenFOAM software via www.openfoam.com.

DISCLAIMER

This source code is not approved or endorsed by OpenCFD Limited, producer and distributor of the OpenFOAM software via www.openfoam.com, and owner of the OPENFOAM(R) and OpenCFD(R) trade marks.

Application

buoyantBoussinesqPimpleWithTurbulentScalarTransportFoam

Description

Modified buoyantBoussinesqPimpleFoam solver to include a passive scalar transport equation with turbulent Schmidt number.

\*-------------------------------------------------------*/

#include "fvCFD.H"
#include "singlePhaseTransportModel.H"
#include "turbulentTransportModel.H"
#include "radiationModel.H"
#include "fvOptions.H"
#include "pimpleControl.H"

// * * * * * * * * * * * * * * * * * * * * * * * * * * //

int main ( int argc , char * argv [] )
{
  #include "postProcess.H"
  #include "setRootCase.H"
  #include "createTime.H"
  #include "createMesh.H"
  #include "createControl.H"
  #include "createFields.H"
  #include "createFvOptions.H"
  #include "createTimeControls.H"
  #include "CourantNo.H"
  #include "setInitialDeltaT.H"
  #include "initContinuityErrs.H"

  turbulence -> validate ();

  // * * * * * * * * * * * * * * * * * * * * * * * * * * //

  Info << "\nStarting time loop\n" << endl;

  while ( runTime.run () )
  {
    #include "readTimeControls.H"
    #include "CourantNo.H"
# include "setDeltaT.H"

runTime++;

Info<< "Time = " << runTime.timeName() << nl << endl;

// --- Pressure-velocity PIMPLE corrector loop
while (pimple.loop())
{
    # include "UEqn.H"
    # include "TEqn.H"

    // --- Pressure corrector loop
    while (pimple.correct())
    {
        # include "pEqn.H"
    }

    if (pimple.turbCorr())
    {
        laminarTransport.correct();
        turbulence->correct();
    }
}

// *** Passive Scalar Transport ***
// Create a scalar field with an effective mass diffusivity
volScalarField DTT("DTT", DT + turbulence->nut()/Sct);

// Define scalar transport equation
fvScalarMatrix CEqn
( 
  fvm::ddt(C) 
  + fvm::div(phi, C) 
  - fvm::laplacian(DTT, C) 
); 
CEqn.solve();
// *** End Passive Scalar Transport ***
runTime.write();

Info<< "ExecutionTime = " << runTime.elapsedCpuTime() << " s" 
<< " ClockTime = " << runTime.elapsedClockTime() << " s" 
<< nl << endl;
}

Info<< "End\n" << endl;

return 0;
}

// ********************************************************** //
A.2 readTransportProperties.H

singlePhaseTransportModel laminarTransport(U, phi);

// Thermal expansion coefficient [1/K]
dimensionedScalar beta
    ("beta",
dimless/dimTemperature,
laminarTransport);

// Reference temperature [K]
dimensionedScalar TRef("TRef", dimTemperature, laminarTransport);

// Laminar Prandtl number
dimensionedScalar Pr("Pr", dimless, laminarTransport);

// Turbulent Prandtl number
dimensionedScalar Prt("Prt", dimless, laminarTransport);

// Mass Diffusivity
dimensionedScalar DT("DT", dimLength*dimLength/dimTime,
    laminarTransport);

// *** Read Turbulent Schmidt Number ***
dimensionedScalar Sct("Sct", dimless, laminarTransport);
// *** End Turbulent Schmidt Number ***
A.3 createFields.H

Info << "Reading thermophysical properties\n" << endl;

Info << "Reading field T\n" << endl;
volScalarField T
(
    IOobject
    ("T",
     runTime.timeName(),
     mesh,
     IOobject::MUST_READ,
     IOobject::AUTO_WRITE
    ),
    mesh
);

// *** Create passive scalar field ***
Info << "Reading field C\n" << endl;
volScalarField C
(
    IOobject
    ("C",
     runTime.timeName(),
     mesh,
     IOobject::MUST_READ,
     IOobject::AUTO_WRITE
    ),
    mesh
);
// *** End passive scalar field ***

Info << "Reading field p_rgh\n" << endl;
volScalarField p_rgh
(
    IOobject
    (
        "p_rgh",
        runTime.timeName(),
        mesh,
        IOobject::MUST_READ,
        IOobject::AUTO_WRITE
    ),
    mesh
);

Info << "Reading field U\n" << endl;
volVectorField U
(
    IOobject
    (
        "U",
        runTime.timeName(),
        mesh,
        IOobject::MUST_READ,
        IOobject::AUTO_WRITE
    ),
    mesh
);

#include "createPhi.H"
#include "readTransportProperties.H"

Info<< "Creating turbulence model\n" << endl;
autoPtr<incompressible::turbulenceModel> turbulence
(
    incompressible::turbulenceModel::New(U, phi, laminarTransport)
);

// Kinematic density for buoyancy force
volScalarField rhok
(
    IOobject
    (
        "rhok",
        runTime.timeName(),
        mesh,
        IOobject::MUST_READ,
        IOobject::AUTO_WRITE
    ),
    1.0 - beta*(T - TRef)
);

// Kinematic turbulent thermal conductivity m2/s
Info<< "Reading field alphat\n" << endl;
volScalarField alphat
(
    IOobject
    (
        "alphat",
        runTime.timeName(),
        mesh,
        IOobject::MUST_READ,
        IOobject::AUTO_WRITE
    ),
mesh
);

#include "readGravitationalAcceleration.H"
#include "readhRef.H"
#include "gh.H"

volScalarField p
(
    IOobject
    (
        "p",
        runTime.timeName(),
        mesh,
        IOobject::NO_READ,
        IOobject::AUTO_WRITE
    ),
    p_rgh + rhok*gh
);

label pRefCell = 0;
scalar pRefValue = 0.0;
setRefCell
(
    p,
    p_rgh,
    pimple.dict(),
    pRefCell,
    pRefValue
);
if (p_rgh.needReference())
{
    p += dimensionedScalar
    ("p",
     p.dimensions(),
     pRefValue - getRefCellValue(p, pRefCell))
}

mesh.setFluxRequired(p_rgh.name());

#include "createMRF.H"
#include "createIncompressibleRadiationModel.H"
APPENDIX B

CREATING BUILDINGS IN SNAPPYHEXMESH

Creating the searchable object files for snappyHexMesh is a semi-automated process. Within QGIS, a virtual layer is created to draw a rectangle matching a portion of the Milwaukee building outline shapefile. The virtual layer is exported as a GeoJSON file with UTM coordinates. With all building components exported, the makeBuildingFile.sh script can executed. Included is the associated code.

B.1 makeBuildingFile.sh

#!/bin/bash

# Read the GeoJSON files within the directory
# GeoJSON files should only have 4 points in them
# Points written in WGS 84 / UTM Zone 16N (units of meters)

# copyThis.txt contains a list of files needed to be included in
# snappyHexMeshDict
rm copyThis.txt
touch copyThis.txt

for geoJSONFile in *.geojson
do
    # Print off the file being worked on
    echo $geoJSONFile

    # Extract the building ID from the file name
    buildingNumber=${geoJSONFile//".geojson"/}

    # extract the 4 points from the geoJSON file
# format the 4 points correctly
awk -F '[][]' '{print $2}' $geoJSONFile > temp.comma
sed 's/,/\t/g' temp.comma > tempExtra.tab
tail -4 tempExtra.tab > temp.tab

# Call the FORTRAN file to do the math
# Building points are assumed to go clockwise
# First vertex is assumed to be the northern most point
./doMathForBuilding

# Extract data from the 3 FORTRAN output files
origin=$(head origin.data)
scale=$(head scale.data)
orientation=$(head orientation.data)

# Create searchableObject file for snappyHexMesh
echo "building_"$buildingNumber > $buildingNumber
echo "{" >> $buildingNumber
echo "surface useThis;" >> $buildingNumber
echo "scale ("$scale");" >> $buildingNumber
echo "transform" >> $buildingNumber
echo "{" >> $buildingNumber
echo "coordinateSystem" >> $buildingNumber
echo "{" >> $buildingNumber
echo "type cartesian;" >> $buildingNumber
echo "origin ("$origin");" >> $buildingNumber
echo "coordinateRotation" >> $buildingNumber
echo "{" >> $buildingNumber
echo "type axesRotation;" >> $buildingNumber
echo "e1 ("$orientation");" >> $buildingNumber
echo "e3 (0 0 1);" >> $buildingNumber
echo "})" >> $buildingNumber
echo "}" >> $buildingNumber

# remove temporary files
rm temp*
rm *.data

echo "# include \"buildingData/$buildingNumber\"" >> copyThis.txt
done

B.2 doMathForBuilding.f95

program doMath4Building
implicit none

! variables

double precision, dimension(2,4) :: points

double precision :: theta, e1_1, e1_2

double precision :: xLength, yLength, xCenter, yCenter

integer :: i, j

double precision, dimension(2) :: pointA, pointB, pointC, pointD

integer :: locA, locB, locC, locD

! Code
! Open file and extract points from the file

open(unit=12, file="temp.comma")

read(12,*) points

close(12)
!!! Determine the position of points A, B, C, and D
pointA = points(:,1)
pointB = points(:,2)
pointC = points(:,3)
pointD = points(:,4)

!!! Find length of sides
xLength = sqrt( (pointA(1)-pointB(1))^2 + &
(pointA(2)-pointB(2))^2 )

yLength = sqrt( (pointB(1)-pointC(1))^2 + &
(pointB(2)-pointC(2))^2 )

!!! Find Center of rectangle
xCenter = 0.25 * (pointA(1) + pointB(1) + pointC(1) + pointD(1))
yCenter = 0.25 * (pointA(2) + pointB(2) + pointC(2) + pointD(2))

!!! Translate the center from UTM 16 to OpenFOAM origin
! Southwest corner of OpenFOAM domain has UTM coordinates
! 426185.0 meters E, 4763284.0 meters N
xCenter = xCenter - 426185.0
yCenter = yCenter - 4763284.0

!!! Find orientation
theta = atan( (pointA(2)-pointB(2)) / &
(pointA(1)-pointB(1)) )

! Calculate direction of e_1
e1_1 = cos(theta)
e1_2 = sin(theta)
! Open temporary files for bash script to read
open(unit=13,file="scale.data")
open(unit=14,file="origin.data")
open(unit=15,file="orientation.data")

! Write to temporary files
! Note: 3.5 is an arbitrary building height
! Note: 2 is an arbitrary building z-origin
write(13,*) xLength, yLength, 3.5
write(14,*) xCenter, yCenter, 2
write(15,*) e1_1, e1_2, 0

close(13)
close(14)
close(15)

end program