DISPERSION OF POLLUTANTS IN THE ATMOSPHERE: A NUMERICAL STUDY

by

Bamdad Hosseini
B. Sc., Sharif University of Technology 2010

a Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science

in the
Department of Mathematics
Faculty of Science

SIMON FRASER UNIVERSITY
Summer 2013

APPROVAL

Name:	Bamdad Hosseini
Degree:	Master of Science
Title of Thesis:	Dispersion of Pollutants in the Atmosphere: A Numerical Study
Examining Committee:	Dr. Weiran Sun, Assistant Professor, Chair
	Dr. John Stockie, Associate Professor, Senior Supervisor
	Dr. Steven J. Ruuth, Professor, Supervisor
	Dr. Nilima Nigam, Associate Professor, Internal/External Examiner
Date Approved:	August 15, 2013

Partial Copyright Licence

Abstract

In this project we propose a numerical algorithm for solving the atmospheric dispersion problem with point sources and deposition at the ground. The problem is modelled by the 3D advection-diffusion equation with delta distribution source terms, as well as variable advection speed and diffusion coefficients. We construct a finite volume scheme using a splitting approach in which the CLAWPACK software package is used as the advection solver and an implicit solver is proposed for the diffusion terms. Next, we apply the algorithm to an actual industrial scenario involving emissions of airborne zinc from a mining smelter using actual wind measurements, and then compare our results to another set of simulations obtained from an algorithm based on a Gaussian plume solver. Finally, we propose a new and efficient framework for solving the inverse source emission problem that utilizes the solution of the adjoint equations.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Atmospheric Pollution

Atmospheric pollution or *air pollution* refers to the presence of possibly harmful chemicals, particulates or biological matter in the atmosphere. Some well known chemical pollutants include sulfur oxides, nitrogen oxides, carbon monoxide and other organic compounds such as methane. These are chemicals that can cause damage to natural or man made structures, as well as being harmful to humans and other animals. Particulates are tiny airborne solid or liquid particles that are suspended in air. They are categorized based on size and can be as big as 10 micrometers in diameter. Examples include ash emitted from volcanoes, plant pollen and road dust. Another important class of atmospheric pollutants is radioactive material that can pose serious risks even at very low concentration. Examples include nuclear explosions, reactor melt downs (such as the well known Chernobyl disaster), and natural processes such as radioactive decay of radon.

In all of these examples our main topic of interest is tracking concentration of a pollutant as it is dispersed via motion of the atmosphere. *Atmospheric dispersion modelling* refers to the mathematical study and simulation of the processes mentioned above. It is a popular discipline and has been an active topic of research for many years. Atmospheric dispersion models are often used to assess the impact of industrial facilities on the ambient environment, either in risk assessment studies or for emergency response.

In general, atmospheric dispersion modelling is an extremely difficult and complicated problem. The process is clearly linked to the behavior of the atmosphere and hence depends on meteorological data. Chemical reactions are always present in the atmosphere

which result in some pollutants being converted into other chemicals that might behave differently. These are naturally nonlinear processes that are difficult to model. The range of pollution transport can range from a few kilometers up to global scales. Pollution can also be transferred between the various layers of the atmosphere, each of which have different properties.

For all of these reasons, atmospheric dispersion modelling necessitates many simplifying assumptions and is accompanied by uncertainties and large errors (either in the input measurements or the results). In this project we are mainly interested in short range transport of pollutants that is on the order of tens of kilometers. Consequently we will ignore large scale effects such as Coriolis forces or the curvature of the earth. In fact, as we will see in chapter 2, we make some fairly major simplifying assumptions on the meteorological aspect of the problem. Here we focus on pollutants being emitted from *point sources*, which represent stacks or other small-area sources of a given height (Figure 1.1). Contaminants emitted into the atmosphere will be transported by the wind while turbulent behavior of the atmosphere will result in mixing and dispersion of the contaminant and gravity acts to deposit the contaminants on the ground. The final aim of this project is to generate a numerical algorithm that is capable of modelling three dimensional dispersion of pollutants in the atmosphere and their deposition on the ground. We are only interested in transport of a single contaminant and so we will not consider chemical reactions. We will construct our model for dispersion of a generic contaminant which can be used to study a wide class of contaminants. In particular, we will study release of zinc from an industrial facility.

1.2 Review of Previous Works

As mentioned above, atmospheric dispersion modelling is an active and well developed area of research. Zlatev [47] presents a comprehensive review of common methods and challenges arising in atmospheric dispersion modelling on a global scale with many references to existing works and common techniques used in the literature. A more compact review of the same topic can be found in the article by Dimov et al. [9]. It is common that the short range dispersion of pollutants is modelled using the advection-diffusion equation. Ever since the introduction of these models by authors such as Taylor [42], Roberts [37] and Sutton [41], various analytical and approximate solutions for this problem have been introduced. Examples of such solutions can be found in the works of Ermak [11] or Fisher

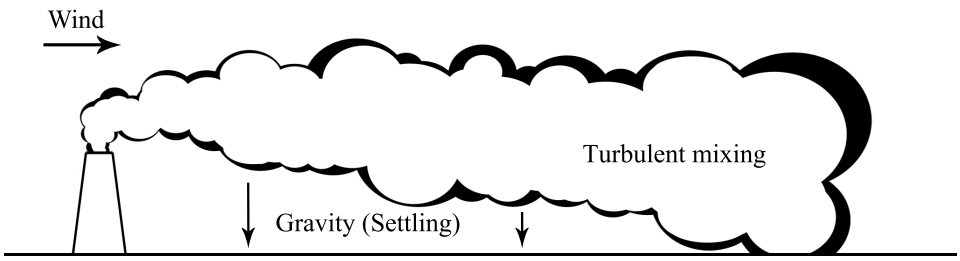


Figure 1.1: Pollution dispersion from a stack due to advection by the wind, diffusion from turbulent mixing and gravitational settling.

and Macqueen [13], most of which fall under the category of Gaussian plume solutions. Seinfeld [39] presents a large collection of Gaussian plume solutions and some of the details behind the derivation of the formulas.

Applications of Gaussian plume models are plentiful in the literature. They have been applied to a large number of industrial studies as well as other applications such as volcanic ash dispersal [43, 10], transport of seeds, pollen and odours [45, 7], or sudden release of biological or radioactive contaminants [20]. As mentioned above there is a large variety of approximate solutions for the dispersion problem. Carrascal et al. [6] investigate the sensitivity of Gaussian plume solutions to parameters in the models and report some concerns regarding sensitivity of the solutions to different atmospheric stability classes. Hamby [16] studies sensitivity of the Gaussian plume solution to wind direction, speed, stability class and other parameters and discovers that the near field solution is very sensitive to meteorological data. As we will see in chapter 2, derivation of Gaussian plume solutions requires a number of simplifying assumptions. See [39] for a collection of Gaussian plume solutions and details of their underlying assumptions. For these reasons one might be interested in a more sophisticated or flexible approach towards solution of the atmospheric dispersion problem. Nevertheless, more complex models are not necessarily more accurate, in the sense that they employ different assumptions and parameterizations whose validity typically depends sensitively on atmospheric conditions, location or other effects.

1.3 Goals and Outline

In this thesis our ultimate goal is to simulate dispersion of pollutants on an industrial site subject to measured wind data in three dimensions. We believe that the main strength of this model is in its ability to handle variable wind and to solve the underlying PDE directly rather than using an approximate solution as is done with Gaussian plume solutions. This work is based on the study of Lushi and Stockie [27] where the inverse source identification problem is considered. Here we develop a three dimensional solver for atmospheric dispersion problems and apply this solver to the same setting they considered. A secondary goal is to compare the solution from this solver to a Gaussian plume solution and also validate results of the inverse problem solved by Lushi and Stockie.

We start by introducing the advection-diffusion PDE as a general model for atmospheric dispersion in chapter 2. We consider generalizations of the model and present as many details as possible while staying in line with the main topic of this thesis. Since the literature on atmospheric dispersion and meteorology is vast we will only briefly touch on this subject. We describe various parameterizations for the velocity fields and diffusion coefficients that are based on both theoretical and experimental results. In chapter 3, we use the models described in chapter 2 and design a numerical scheme based on the CLAWPACK package for solving hyperbolic conservation laws. Afterwards, we apply our numerical scheme to do a simulation based on the real world setting considered in [27]. Finally in chapter 4 we discuss the adjoint equation formulation and its application to the solution of inverse source identification problems. These results are not intimately connected to the first few chapters of the thesis and instead set the stage for future work on this topic.

Chapter 2

Mathematical Models For Atmospheric Dispersion

Here we present some common mathematical models that are used for studying the atmospheric dispersion problem. We start by introducing the advection-diffusion equation which is a partial differential equation (PDE) that represents the physical system. Point and line sources are introduced as examples of stacks or roads that emit pollutants. Afterwards, we consider some analytical and approximate solutions of the PDE in simple circumstances including instantaneous and steady sources. Later, we describe models for approximating the wind velocity profiles and eddy diffusion coefficients based on a combination of experimental and theoretical results, and finish this chapter with a discussion of dry deposition of particulates. All of these concepts are later built into a numerical scheme and their implementation is considered in Chapter 3.

2.1 Advection-Diffusion Equation

Atmospheric dispersion of pollutants is an example of a mass transport phenomenon where we are generally interested in the advection of contaminants due to the wind, along with dispersion due to turbulent mixing in the atmosphere. It is natural to consider the advection-diffusion equation as a model for this problem. For convenience let us start by deriving the equations for a single contaminant, after which the extension to multiple contaminants is straightforward. For a detailed treatment of advection-diffusion-reaction systems including

examples, see Hundsdorfer [18], Zlatev [46] or Dimov et al. [9]. A detailed derivation and justification of our model can be found in the monograph by Seinfeld [39] but our approach here is similar to that of Stockie [40]. Let $c(\vec{X},t)$ $[kg/m^3]$ denote mass concentration (or density) of a pollutant at time $t\,[s]$ and position $\vec{X}=(x,y,z)\in\mathbb{R}^3\,[m]$. Conservation of the contaminant results in the following equation for concentration,

$$\frac{\partial c}{\partial t}+\nabla\cdot\vec{f}=s, \tag{2.1.1}$$

where $\vec{f}(\vec{X},t)$ $[kg/m^2s]$ is the contaminant mass flux due to advection and diffusion and $s(\vec{X},t)$ $[kg/m^3s]$ is a source term. The diffusive contribution to the flux is due to turbulent mixing and is generally believed to follow Fick's law; that is, the diffusive flux is taken proportional to the concentration gradient $\vec{f_D}:=-\mathbf{K}\nabla c$. The diffusion tensor $\mathbf{K}:=\mathrm{diag}\,(K_x,K_y,K_z)$ $[m^2/s]$ is a diagonal matrix with the eddy diffusion coefficients usually being functions of position. The wind advects the pollutant downstream and results in a simple linear advective flux, $\vec{f_A}:=c\vec{u}$ where $\vec{u}=(u_x,u_y,u_z)\,[m/s]$ is the wind velocity. Adding together the two fluxes and setting $\vec{f}=\vec{f_D}+\vec{f_A}$, we can rewrite (2.1.1) as the familiar advection-diffusion PDE

$$\frac{\partial c}{\partial t}+\nabla\cdot(c\vec{u})=\nabla\cdot(\mathbf{K}\nabla c)+s. \tag{2.1.2}$$

2.1.1 Boundary Conditions

Various boundary conditions can be imposed along with (2.1.2) but we are interested in solving this equation in the half-space $z\geq 0$ where $z=0$ denotes the ground. We also restrict ourselves to short range dispersion of pollutants on the order of tens of kilometers, so that it makes sense to consider the concentration c tending to zero in the far field. This is consistent with the conservation of contaminant mass in the domain. The boundary condition at ground level is more interesting since this is where deposition of pollutants takes place. A simple example would be a non-penetrable ground which corresponds to having zero vertical advective and diffusive fluxes at ground level, or

$$u_z c-K_z\frac{\partial c}{\partial z}=0\quad\text{at } z=0. \tag{2.1.3}$$

In a more general setting, one could consider a settling velocity for the particulates with $u_z=-u_{set}$ and a deposition coefficient W_{dep} that captures the effect of the total flux of

contaminants penetrating the ground. This results in the Robin boundary condition

$$-u_{set}c - K_z\frac{\partial c}{\partial z} = -W_{dep}c. \tag{2.1.4}$$

2.1.2 Source Terms

The approximation of the contaminant sources is of crucial importance in modelling dispersion of pollutants. Source terms represent pollutant sources or sinks and assumptions on their form is essential in the derivation of analytical solutions as well as in numerical studies of (2.1.2). A simple form of the source term arises for a point source representing a stack or exhaust port, which can be modelled as a delta distribution,

$$s(\vec{X},t) = Q\delta(\vec{X}-\vec{X}_s) = Q\delta(x-x_s)\delta(y-y_s)\delta(z-z_s) \tag{2.1.5}$$

in 3D, where $\delta(\cdot)$ is the Dirac delta distribution, $\vec{X}_s = (x_s, y_s, z_s)$ is the position of the source and $Q\,[kg/s]$ is the total output of the source in unit time. For a detailed treatment of distributions and their definition in higher dimensional spaces see Rudin [38]. We come back to the topic of distributions in section 3.4 where we discuss the numerical approximation of point sources.

Line sources are another common type of source term which represent roads or arrays of closely-situated stacks. For example, a line source of length L extending from $(x_s, -L/2, z_s)$ to $(x_s, L/2, z_s)$ can be represented by

$$s(\vec{X},t) = Q_L\delta(x-x_s)\gamma(y)\delta(z-z_s) \tag{2.1.6}$$

where the distribution $\gamma(y)$ is given by

$$\gamma(y) = \begin{cases} 1 & \text{for} \quad |y| \le L/2, \\ 0 & \text{otherwise} \end{cases} \tag{2.1.7}$$

and $Q_L\,[kg/m\,s]$ is the emission rate per unit length of the source in unit time. For detailed treatment of line sources see [33].

We can also consider distributed sources representing piles of debris, stacks that are large compared to the typical advection-diffusion scale, or pollutants spilled over large areas. These area sources have the general form

$$s(\vec{X},t) = \psi(x-x_s, y-y_s, z-z_s) \tag{2.1.8}$$

where $\psi(\vec{X})$ is a distribution with compact support.

2.1.3 Multiple Contaminants

The advection-diffusion equation (2.1.2) is the main focus of this thesis and we are concerned throughout this document with solutions corresponding to a single contaminant. Still, it is worthwhile to mention the extension of this model to multiple contaminants. Assuming there are m contaminants, then we have a PDE for each contaminant c_k for $k = 1, .., m$:

$$\frac{\partial c_k}{\partial t} + \nabla \cdot (c_k \vec{u}) = \nabla \cdot (\mathbf{K} \nabla c_k) + s_k, \tag{2.1.9}$$

note that the diffusion tensor is the same for each contaminant since eddy diffusivity is an attribute of the flow rather than the tracer particle, which is different from molecular diffusion. Contaminant interactions can be modelled via the source terms s_k to generate models that capture chemical reaction of contaminants resulting in coupling of the equations in (2.1.9).

2.2 Some Analytical Solutions

Here we discuss a few analytical solutions of (2.1.2) for various initial and boundary conditions that aid us in building intuition into the behavior of the advection-diffusion equation. Of particular importance is a class of steady state solutions known as *Gaussian plume* solutions that are used extensively in practice and are known to give acceptable approximations to real world measurements, see for example [4, 3, 14, 24]. Gaussian plume solutions are attractive mainly because they are fast to compute and give physically relevant results even in complicated conditions such as time dependent wind, source inversion problems and approximation of dispersion over complicated terrain. For a discussion on the limitations of Gaussian plume solutions see [44]. Details of deriving the Gaussian plume solutions are discussed by Stockie [40] with references to many existing works in the literature. In particular, Seinfeld [39] presents a detailed treatment of Gaussian plume solutions, their derivation and application in various cases, and many references to other sources on this topic. Here we do not go into the details of the derivations, and instead just summarize the results.

2.2.1 Unbounded Domain with Instantaneous Source

Let us first consider a simplified case of the advection-diffusion problem. Consider the homogeneous form of (2.1.2) in the whole of $\mathbb{R}^3$ with non-isotropic diffusion, uni-directional

wind u in the x direction and a point source initial condition at location $\vec{X}_s = (0,0,0)$. To summarize, we will consider the problem

$$\begin{cases} \dfrac{\partial c}{\partial t} + u\dfrac{\partial c}{\partial x} = \nabla \cdot (\mathbf{K}\nabla c), & \text{for} \quad \vec{X} \in \mathbb{R}^3,\ t \geq 0 \\ c(\vec{X}, 0) = Q\delta(x)\delta(y)\delta(z), & \\ c(\vec{X}, t) \to 0 \qquad \text{as} \quad \left|\vec{X}\right| \to \infty, & \end{cases} \tag{2.2.1}$$

where the last far-field boundary condition is prescribed to make sure that the total mass of the contaminant is conserved in the domain. Making the change of variables

$$\tau = t, \quad \xi = \frac{x - ut}{\sqrt{K_x}}, \quad \eta = \frac{y}{\sqrt{K_y}}, \quad \zeta = \frac{z}{\sqrt{K_z}}, \tag{2.2.2}$$

where we recall $\mathbf{K} = \text{diag}(K_x, K_y, K_z)$ and employing the chain rule we have

$$\begin{aligned} &\frac{\partial c}{\partial t} = \frac{\partial c}{\partial \tau} - \frac{u}{\sqrt{K_x}}\frac{\partial c}{\partial \xi}, \quad \frac{\partial c}{\partial x} = \frac{1}{\sqrt{K_x}}\frac{\partial c}{\partial \xi}, \\ &\frac{\partial^2 c}{\partial x^2} = \frac{1}{K_x}\frac{\partial^2 c}{\partial \xi^2}, \quad \frac{\partial^2 c}{\partial y^2} = \frac{1}{K_y}\frac{\partial^2 c}{\partial \eta^2}, \quad \frac{\partial^2 c}{\partial^2 z} = \frac{1}{K_z}\frac{\partial^2 c}{\partial \zeta^2}, \end{aligned}$$

which reduces (2.2.1) to the isotropic heat equation with a singular source term in the new coordinates $\vec{\Theta} = (\xi, \eta, \zeta)$,

$$\begin{cases} \dfrac{\partial c}{\partial \tau} = \nabla \cdot \nabla c, \quad \text{for} \quad \vec{\Theta} = (\xi, \eta, \zeta) \in \mathbb{R}^3,\ \tau \geq 0, \\ c(\vec{\Theta}, 0) = \dfrac{Q}{\sqrt{K_x K_y K_z}}\delta(\xi + ut)\delta(\eta)\delta(\zeta), \\ c(\vec{\Theta}, \tau) \to 0 \qquad \text{as} \quad \left|\vec{\Theta}\right| \to \infty. \end{cases} \tag{2.2.3}$$

This is the fundamental form of the heat equation with an extra factor multiplying the initial condition due to scaling of the coordinates and the emission rate. For a detailed treatment of the fundamental solution of the heat equation, see [12]. The fundamental solution of (2.2.3) is given by

$$c(\vec{\Theta}, \tau) = \frac{Q}{(4\pi\tau)^{3/2}\sqrt{K_x K_y K_z}} \exp\left[-\frac{\left|\vec{\Theta}\right|^2}{4\tau}\right]. \tag{2.2.4}$$

This tells us that the solution is the same as that of the heat equation for an observer that is moving at the advection speed. The fundamental solution in the original coordinates is given by

$$c(\vec{X}, t) = \frac{Q}{(4\pi t)^{3/2}\sqrt{K_x K_y K_z}} \exp\left[-\frac{1}{4t}\left(\frac{(x-ut)^2}{K_x} + \frac{y^2}{K_y} + \frac{z^2}{K_z}\right)\right]. \tag{2.2.5}$$

2.2.2 Non Penetrable Ground with Point Source

A simple case of the Gaussian plume solution is the steady state solution of (2.1.2) for unidirectional wind with non penetrable boundary conditions at ground level for a point source. In order to derive the solution, several assumptions must be made. First, we represent the source term as a delta distribution centered at $\vec{X}_s = (0, 0, H)$ and neglect topographical variations i.e. treat the ground as the $z = 0$ plane. Without loss of generality, we assume that the wind blows in the x direction so that $\vec{u} = (u, 0, 0)$. This solution can be generalized to other wind directions by a simple rotation of coordinates. We also assume the contaminant cannot penetrate the ground, i.e. equation (2.1.3) holds and we assume concentrations tend to zero in the far-field. To summarize, the boundary conditions are

$$\begin{gathered} K_z \frac{\partial c}{\partial z}(x, y, 0) = 0, \\ \lim_{x\to\infty} c(x, y, z) = 0, \quad \lim_{|y|\to\infty} c(x,y, z) = 0, \quad \lim_{z\to\infty} c(x, y, z) = 0, \quad c(0, y, z) = 0. \end{gathered} \tag{2.2.6}$$

Note that the solution for negative values of x is not physically relevant because the wind velocity is positive, so we consider the solution only in the orthant $x, z \geq 0$.

As for the diffusion coefficients, we consider isotropic eddy diffusivities so that $K_x = K_y = K_z = K$. Since the problem is advection-dominated, we can also assume the advective flux in the x direction is much greater than the diffusive flux, and therefore we can take $K_x = 0$.

Putting all of these assumptions together, along with the fact that we are looking for a steady state solution, (2.1.2) simplifies to

$$u\frac{\partial c}{\partial x} = K\frac{\partial^2 c}{\partial y^2} + K\frac{\partial^2 c}{\partial z^2} + Q\delta(x)\delta(y)\delta(z - H). \tag{2.2.7}$$

The solution to this equation is given in [40] as

$$c(r, y, z) = \frac{Q}{4\pi u r} \exp\left(\frac{-y^2}{4r}\right)\left[\exp\left(-\frac{(z - H)^2}{4r}\right) + \exp\left(-\frac{(z + H)^2}{4r}\right)\right], \tag{2.2.8}$$

where the variable r is related to x via

$$r = \frac{1}{u}\int_0^x K(\rho)\, d\rho. \tag{2.2.9}$$

This solution is commonly referred to as the Gaussian plume solution since (2.2.8) is essentially a Gaussian distribution with standard deviation

$$\sigma^2 = \frac{2}{u}\int_0^x K(\rho)\, d\rho = 2r. \tag{2.2.10}$$

The σ coefficients are much easier to measure than the diffusion coefficients, which is one of the major reasons why the Gaussian plume solutions are widely used in practice.

2.2.3 Gaussian Puff Solution

We can also consider the problem in (2.2.1) with a reflecting boundary condition at ground, that is

$$\begin{cases} \dfrac{\partial c}{\partial t} + u\dfrac{\partial c}{\partial x} = \nabla \cdot (\mathbf{K}\nabla c), \qquad \text{for} \quad \vec{X} \in (\mathbb{R}^2 \times \mathbb{R}^+),\ t \geq 0, \\ c(\vec{X}, 0) = Q\delta(x)\delta(y)\delta(z-H), \quad \left.\dfrac{\partial c}{\partial z}\right|_{z=0} = 0, \\ c(\vec{X}, t) \to 0 \qquad \text{as} \quad \left|\vec{X}\right| \to \infty. \end{cases} \tag{2.2.11}$$

Note that this problem is very similar to that of Section 2.2.2 except that the source is not continuous in time but rather consists of a single instantaneous "puff" at $t = 0$. The solution is given by

$$\begin{aligned} c(x,y,z,t) = \frac{Q}{(2\pi t)^{3/2}\sigma_x\sigma_y\sigma_z} \exp\left(-\frac{(x-ut)^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}\right) \\ \times \left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right], \end{aligned} \tag{2.2.12}$$

where

$$\sigma_{x,y,z}^2 = \frac{2}{u}\int_0^x K_{x,y,z}(\rho)\, d\rho. \tag{2.2.13}$$

See Chapter 18 of [39] for more details on derivation of these solutions using Fourier transforms.

2.3 Velocity and Eddy Diffusivity Coefficients

One of the important aspects of modelling atmospheric dispersion is the spatial dependence of parameters in the model. Atmospheric dispersion models often include a large number of parameters that have an inherently large uncertainty or measurement error. This allows a lot of freedom in the parameter space which in turn enables one to fit the solution to almost any measurement. Also, models can be insensitive to some parameters, meaning that one could easily achieve a solution that looks reasonable and yet has all the wrong parameters. Often, it is the case that measurements are provided with large error margins, which means that the simulations will be acceptable for a wider range of parameters. Loosely speaking,

the problem has too many degrees of freedom. For these reasons, when choosing a model for the parameters, one should avoid enlarging the parameter space as much as possible.

Keeping these issues in mind, it is sometimes impossible to avoid introducing new parameters to the model. The most obvious case for our problem is the velocity profile $\vec{u}(x, y, z, t)$. Wind velocity is often measured at a finite number of locations (usually only one) at a certain height. Since we are interested in transport at small scales we have to somehow "guess" what the velocity profile looks like as a function of height. Of course, the velocity profile will vary based on location and topography as well but trying to construct the entire velocity field with only one point measurement is a futile endeavor. Also, in order to model the diffusion coefficients, we need to have an idea of how the eddy diffusivity coefficients vary in space. We often do not have a direct measurement of these coefficients and have to find them implicitly based on atmospheric stability classes and other measurements. In practice one can make measurements of the standard deviation parameters $\sigma_x, \sigma_y, \sigma_z$ which are directly related to the diffusion coefficients.

2.3.1 Velocity Profiles

A lot of effort has been put into studies of velocity profiles $\vec{u}$ and theory of turbulent mixing in the atmospheric boundary layer. Arya [1] has a very detailed treatment of velocity profiles and turbulent diffusivity coefficients for momentum and heat. This is a classic reference in the literature that considers both neutral and thermally stratified surface layers. For the sake of simplicity we assume that the surface layer is neutral. Of course this choice is not strictly correct since we do not have enough data to decide of what type the boundary layer actually is. However, we have decided to apply Occam's razor and stick to the simplest possible model.

There are two main approximations used in the literature: the first is a power law that does not have a sound theoretical basis but tends to fit observations well [1]; and the second is a logarithmic law that can be derived from theory but is not typically as good a fit as the power law. We will not present this theory since it is outside the scope of this thesis and requires extensive background in the theory of atmospheric turbulence. The power law assumes the relationship

$$\frac{u(z)}{u_r} = \left(\frac{z}{z_r}\right)^m, \tag{2.3.1}$$

where $u(z)$ is the velocity at height z. The reference velocity is denoted by u_r and z_r is

the height at which the reference velocity is measured and is commonly taken to be 10 meters. This is a modified version of Prandtl's boundary layer velocity for flow over flat surfaces. The exponent m is known to vary from 0.1 for smooth surfaces to 0.4 for rough urban areas. The exponent also depends on the stability of the atmosphere and generally increases with stability, so that there is more variation in more disturbed surface layers. Hence, the exponent tends to vary from case to case but for generally open and smooth areas $m = 1/7$ is considered a "good compromise" [35]. This is the same value as suggested by Prandtl for laminar boundary layers. Therefore, we will take the velocity profile to be the same as that of laminar flow over a flat surface.

The logarithmic law assumes the following relationship

$$\frac{u(z)}{u_r} = 1 + \frac{\ln(z/z_0)}{\ln(z_r/z_0)}, \tag{2.3.2}$$

where z_0 is known as the roughness parameter or roughness length and varies depending on surface type and stability conditions between 10-100 meters [1]. The two velocity profiles (power law and logarithmic law) can be set to match each other closely but still not much can be said about their parameters unless enough measurements are made in the area of interest over a long period of time. Choosing an appropriate value of m or z_0 is more an exercise in statistics and data fitting (see [34] as an example of such studies). Also note that for the power-law and the logarithmic-law the velocity at $z = 0$ is either zero or undefined, which results in an inconsistency in our simulations; therefore we regularize these velocity profiles by taking the velocity to be constant below a certain height z_{cutoff}. The reason for this inconsistency is that these profiles only make sense on a scale that is larger than that of the turbulent behavior of the atmosphere and they are often accepted at a certain height above the ground.

2.3.2 Eddy Diffusion Coefficients

The eddy diffusion coefficients (K_x, K_y, K_z) are difficult to measure in practice and for this reason studies of these parameters are rather scarce. A compact treatment of corrections to these coefficients is presented by Seinfeld and Pandis [39] which is also the main reference for this section. The expressions are based on similarity solutions by Monin and Obukhov [32] who were the pioneers in developing the theory of turbulent mixing in the atmosphere. Going back to our derivation of the advection-diffusion equation (2.1.2), we are interested

Surface	$z_0\ (m)$
Very Smooth (ice, mud)	10^{-5}
Snow	10^{-3}
Smooth sea	10^{-3}
Level desert	10^{-3}
Lawn	10^{-2}
Uncut Grass	0.05
Full grown root crops	0.1
Tree Covered	1
Low-density residential	2
Central business district	$5-10$

Table 2.1: Surface roughness parameter z_0 for various terrain types, source [31].

in finding how the diffusion coefficients K_x, K_y, K_z vary with height. Let us start with the vertical diffusivity.

Vertical Diffusivity K_z

Following the Monin-Obukhov similarity theorem and observational results, K_z can be expressed as

$$K_z = \frac{\kappa u_* z}{\phi(z/L)}, \tag{2.3.3}$$

where

$$\phi(z/L) = \begin{cases} 1 + 4.7z/L & \text{Stable (E, F)}, \\ 1 & \text{Neutral (D)}, \\ (1 - 15z/L)^{1/2} & \text{Unstable (A, B, C)}, \end{cases} \tag{2.3.4}$$

where κ is known as the *von Karman* constant and is assumed to be approximately 0.4, u_* is a characteristic velocity also known as the *friction velocity*, and L is a length scale referred to as the *Monin-Obukhov length* [39]. The friction velocity can be written as a function of roughness z_0 and velocity at the reference height

$$u_* = \frac{\kappa u_r}{\ln(h_r/z_0)}, \tag{2.3.5}$$

assuming an adiabatic temperature profile. A complete derivation of the Monin-Obukhov length is presented in Section 16.4.3 of [39] but to simplify its calculation we use the expression proposed by Golder [15]

$$\frac{1}{L} = a + b \log z_0, \tag{2.3.6}$$

where the parameters a and b are determined based on the Pasquill stability class of the atmosphere as shown in Table 2.2.

Pasquill Stability Class	a	b
A (Extremely unstable)	-0.096	0.029
B (Moderately unstable)	-0.037	0.029
C (Slightly unstable)	-0.002	0.018
D (Neutral)	0	0
E (Slightly stable)	0.004	-0.018
F (Moderately stable)	0.035	-0.036

Table 2.2: Monin-Obukhov length parameters for different stability classes, source [39].

Combining equations (2.3.3)–(2.3.6), we have a method for identifying K_z based on the stability of the atmosphere and the wind velocity at a reference height. There are many assumptions and uncertainties underlying these results and therefore they should only be viewed as guidelines for choices of parameters. We could also bypass u_* by assuming that the diffusivity is known at a reference height z_r (usually taken to be $10m$), and then use (2.3.3) to get

$$\frac{K_z(z)}{K_z(z_r)} = \frac{z\,\phi(z_r/L)}{z_r\,\phi(z/L)}, \tag{2.3.7}$$

which is more useful in cases where we have direct access to the diffusion coefficient at some height. Many other approximations of K_z exist in the literature and some of them are mentioned in Section 18.12.2 of [39] but we choose to use these simpler equations. In fact, (2.3.7) is the only expression we build into our code since in reality we cannot say much about behavior of the atmosphere in our real world application. Note that the vertical diffusivity at ground level vanishes in our model; this results in an inconsistency in the boundary conditions of the PDE. In order to avoid this inconsistency we regularize the diffusivity profile by taking $K_z(z)$ to be constant below a certain height z_{cutoff} which we take to be the same cutoff height as that of the velocity profile.

Horizontal Diffusivity K_x and K_y

It is reasonable to assume that $K_x = K_z$ in our application due to symmetry. An expression based on measurements of the σ_x and σ_y parameters for unstable conditions (due to [39]) is

$$K_y \simeq 0.1 z_i^{3/4} (-\kappa L)^{-1/3} u_* \tag{2.3.8}$$

where z_i is height of the mixing layer changing between 100 meters to 3km depending on topography, stability, season and time of year. Note that the value of L is negative only for unstable conditions, and therefore this expression cannot be used directly in stable and neutral atmospheric classes. Here we use (2.3.8) to approximate the horizontal diffusivities for stable and neutral conditions since we do not have another model. Note that the horizontal diffusivities are assumed to be independent of height as opposed to the vertical diffusivity.

2.3.3 Dry Deposition

An accepted assumption in modelling the deposition of particulates is that the deposition flux is proportional to the concentration at ground level [39]. Therefore we can assume the following relationship for the vertical component of the flux at ground level

$$f_z|_{z=0} = -W_{dep} C, \tag{2.3.9}$$

where $W_{dep}\,[m\,s^{-1}]$ is the deposition coefficient (or deposition velocity). An extensive discussion of this parameter is presented by Seinfeld [39]. In this project we are mainly interested in two specific contaminants: zinc (Zn) and strontium (Sr) and we use the deposition velocity given by Lushi and Stockie [27] that is $0.005\,[m/s]$ for both contaminants.

Chapter 3

The Numerical Scheme

In this section we discuss details of the numerical scheme used to obtain the results in this thesis. One of our main objectives is to develop a 3D advection-diffusion solver based on CLAWPACK 4.3. We start by explaining the idea of splitting and justify why the Godunov splitting is preferable for our problem; then we move on to treat the split equations. A short introduction to finite volume schemes and hyperbolic conservation laws is followed by a description of the finite volume formulation of the diffusive term. Next we briefly discuss our discretization of the advective term and explain how CLAWPACK is used to solve the advective term. We also mention our treatment of the point sources and discuss how higher order approximations to delta distributions can be constructed by solving a finite dimensional moment problem. Afterwards we assemble all the pieces of our split scheme and explain the order in which the sub-problems are solved. Finally we present a convergence study with several test cases of varying complexity and show that the expected order of convergence is obtained in all cases.

3.1 Splitting

Here we discuss the splitting approach, which breaks the original equation into smaller sub-problems that can be solved more easily or efficiently. This is a common technique that is often employed in the solution of larger systems of equations or for cases where significant efficiency improvements can be achieved by breaking the problem into smaller parts. Depending on the context and the type of terms that are split, these methods are called *directional splitting*, *source term splitting* or *fractional step* methods. Since we are interested

in eventually splitting various physical behaviours of the system (advection, diffusion and the source term) as well as splitting derivatives in the various coordinate directions, we prefer to use the term *operator splitting* to refer to the most general form of a splitting scheme.

Many well known numerical methods such as Alternating Direction Implicit (ADI) methods and Implicit Explicit (IMEX) methods can be categorized as operator splitting schemes. Hundsdorfer [18] presents a comprehensive review of splitting schemes and provides a general theory of split operators in the last two chapters with many examples focused on advection-diffusion-reaction equations. LeVeque [25] also has an extensive treatment of splitting methods and their error analysis. We start by discussing the simpler first order splitting schemes and consider Godunov splitting since this will be the method of choice in our project. We will briefly mention Strang splitting but will not discuss other techniques since these are not relevant to this project. The reader is advised to look at the books mentioned above as well as the articles [28, 2] for treatment of ADI and IMEX schemes respectively.

3.1.1 First order Splitting Methods

Consider the first order system of equations

$$\omega_t(t) = \mathcal{L}\,\omega(t), \tag{3.1.1}$$

where $\mathcal{L}$ is a general (not necessarily linear) differential operator, either in continuous or discrete form that arises from a PDE with homogeneous or periodic boundary conditions. For simplicity, we assume that $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$ can be split into two operators; in the case of the advection-diffusion equation, $\mathcal{L}_1$ and $\mathcal{L}_2$ may be the advection and diffusion operators respectively. Assuming that $\omega(t)$ is well behaved, we can represent the process of updating the solution from one time step to the next as

$$\omega(t_{n+1}) = e^{\Delta t\,\mathcal{L}}\omega(t_n), \tag{3.1.2}$$

where

$$e^{\Delta t\,\mathcal{L}} = \mathcal{I} + \Delta t\,\mathcal{L} + \frac{1}{2}\Delta t^2\,\mathcal{L}^2 + ... + \frac{1}{n!}\Delta^n\,\mathcal{L}^n + \cdots \tag{3.1.3}$$

is the *exponential expansion* of the operator $\mathcal{L}$ and Δt is the time step. This is a common notation employed in the literature on operator splitting. Straightforward Taylor expansion

of all terms in (3.1.1) about $t = t_n$ tells us where the notation comes from. For simplicity, assume that the operator $\mathcal{L}$ does not depend on time. Then formally differentiating (3.1.1) with respect to t yields

$$\omega_{tt} = \mathcal{L}\,\omega_t = \mathcal{L}^2\omega, \tag{3.1.4}$$

which generalizes to

$$\frac{\partial^n \omega}{\partial t^n} = \mathcal{L}^n\omega. \tag{3.1.5}$$

Substituting this expression into the Taylor expansion of $\omega(t)$ around t_n gives

$$\omega(t_{n+1}) = \left(\mathcal{I} + \Delta t\,\mathcal{L} + \frac{1}{2}\Delta t^2\,\mathcal{L}^2 + ...\right)\omega(t_n) = e^{\Delta t\,\mathcal{L}}\omega(t_n). \tag{3.1.6}$$

Therefore it is natural to think of $e^{\Delta t\,\mathcal{L}}$ as the exact solution map for (3.1.1) or the time stepping operator. Having these results at our disposal, we can design a calculus of solution operators that is useful in terms of error analysis. Of course, one must return to the original Taylor expansion in order to understand how the error behaves in the case of time dependent or nonlinear operators.

Now if we are interested in solving (3.1.2) by operator splitting we could solve the equivalent problem

$$\begin{aligned} \omega^*(t_n) &= e^{\Delta t\,\mathcal{L}_1}\omega(t_n), \\ \omega(t_{n+1}) &= e^{\Delta t\,\mathcal{L}_2}\omega^*(t_n), \end{aligned} \tag{3.1.7}$$

so that the new solution operator is

$$\omega(t_{n+1}) = e^{\Delta t\,\mathcal{L}_2}e^{\Delta t\,\mathcal{L}_1}\omega(t_n). \tag{3.1.8}$$

Expanding both operators will help us identify the truncation error

$$\begin{aligned} e^{\Delta t\,\mathcal{L}} &= \left(\mathcal{I} + \Delta t\,(\mathcal{L}_1 + \mathcal{L}_2) + \frac{1}{2}\Delta t^2\,(\mathcal{L}_1 + \mathcal{L}_2)^2 + ...\right), \\ e^{\Delta t\,\mathcal{L}_2}e^{\Delta t\,\mathcal{L}_1} &= \left(\mathcal{I} + \Delta t\,(\mathcal{L}_1 + \mathcal{L}_2) + \frac{1}{2}\Delta t^2\,(\mathcal{L}_1^2 + 2\mathcal{L}_2\mathcal{L}_1 + \mathcal{L}_2^2) + ...\right). \end{aligned} \tag{3.1.9}$$

Therefore the truncation error is

$$\frac{1}{\Delta t}\left(e^{\Delta t\,\mathcal{L}} - e^{\Delta t\,\mathcal{L}_2}e^{\Delta t\,\mathcal{L}_1}\right)w(t) = \frac{1}{2}\Delta t\,(\mathcal{L}_1\mathcal{L}_2 - \mathcal{L}_2\mathcal{L}_1)\,w(t) + \mathcal{O}(\Delta t^2), \tag{3.1.10}$$

which shows that the splitting scheme introduces a first order error in time unless the operators $\mathcal{L}_1$ and $\mathcal{L}_2$ commute, that is $\mathcal{L}_2\mathcal{L}_1 = \mathcal{L}_1\mathcal{L}_2$. Obviously this is not always true. A

simple example is given by the advection and diffusion operators,

$$\begin{aligned} \mathcal{L}_1 &= \nabla \cdot (\vec{u}), \\ \mathcal{L}_2 &= \nabla \cdot (K \nabla), \end{aligned} \tag{3.1.11}$$

for which

$$\begin{aligned} \mathcal{L}_2\mathcal{L}_1\omega &= \nabla \cdot (K \nabla (\nabla \cdot (\vec{u}\,\omega))), \\ \mathcal{L}_1\mathcal{L}_2\omega &= \nabla \cdot (\vec{u}\,(\nabla \cdot (K \nabla\omega))), \end{aligned} \tag{3.1.12}$$

are not the same operators in the general case where the advection speed u or diffusion coefficients K depend on spatial coordinates. This is clearly the case for our computations since, as we saw in Section 2.4, both wind speed and vertical diffusion coefficients vary with altitude. While formally only first order accurate, the coefficient of the $\mathcal{O}(\Delta t)$ term in Godunov splitting may be much smaller than the coefficients in the second order terms and therefore the first order scheme is not really so bad (Section 17.5 of [25]). Godunov splitting is generally a valuable tool since it ensures first order error for a large class of the solution operators and is ideal for extending existing 1D codes to higher dimensions with minimum programming overhead.

Extension of these concepts to more operators can be achieved naturally via the solution operator notation. For example, given three operators $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$ we have

$$\omega(t_{n+1}) = e^{\Delta t\,\mathcal{L}_3} e^{\Delta t\,\mathcal{L}_2} e^{\Delta t\,\mathcal{L}_1} \omega(t_n). \tag{3.1.13}$$

As expected the truncation error becomes more complicated and commutativity conditions become more entangled;

$$\begin{aligned} &\frac{1}{\Delta t}\left(e^{\Delta t\,\mathcal{L}} - e^{\Delta t\,\mathcal{L}_3} e^{\Delta t\,\mathcal{L}_2} e^{\Delta t\,\mathcal{L}_1}\right) w(t) = \\ &\quad \frac{1}{2}\Delta t\left((\mathcal{L}_1\mathcal{L}_2 - \mathcal{L}_2\mathcal{L}_1) + (\mathcal{L}_1\mathcal{L}_3 - \mathcal{L}_3\mathcal{L}_1) + (\mathcal{L}_2\mathcal{L}_3 - \mathcal{L}_3\mathcal{L}_2)\right) w(t) + \mathcal{O}(\Delta t^2). \end{aligned} \tag{3.1.14}$$

These extensions are key to constructing higher order splitting schemes.

3.1.2 Strang Splitting

Following the idea of constructing a commutative pair of operators, we move on to Strang splitting which is also very popular in practice. Strang splitting avoids the lack of commutativity by applying the split operators in alternating order over two half time steps, that

is

$$\begin{aligned}\omega(t_{n+1}) &= \left(e^{\frac{1}{2}\Delta t\,\mathcal{L}_1}e^{\frac{1}{2}\Delta t\,\mathcal{L}_2}\right)\left(e^{\frac{1}{2}\Delta t\,\mathcal{L}_2}e^{\frac{1}{2}\Delta t\,\mathcal{L}_1}\right)\omega(t_n),\\ &= e^{\frac{1}{2}\Delta t\,\mathcal{L}_1}e^{\Delta t\,\mathcal{L}_2}e^{\frac{1}{2}\Delta t\,\mathcal{L}_1}\omega(t_n).\end{aligned} \tag{3.1.15}$$

It is easy to see that in this case the overall split operators commute and the first order truncation error in (3.1.10) is eliminated. Defining the commutator

$$[A_1, A_2] = A_1A_2 - A_2A_1, \tag{3.1.16}$$

and applying the exponential expansions successively we see that the truncation error is given by

$$\begin{aligned}\frac{1}{\Delta t}\left(e^{\Delta t\,\mathcal{L}} - e^{\frac{1}{2}\Delta t\,\mathcal{L}_1}e^{\Delta t\,\mathcal{L}_2}e^{\frac{1}{2}\Delta t\,\mathcal{L}_1}\right)w(t) =&\\ \frac{1}{24}\Delta t^2\left([\mathcal{L}_1,[\mathcal{L}_1,\mathcal{L}_2]] + 2[\mathcal{L}_2,[\mathcal{L}_1,\mathcal{L}_2]]\right)\omega(t_{n+1/2}) + \mathcal{O}(\Delta t^4),&\end{aligned} \tag{3.1.17}$$

which is second order in Δt.

We will not employ Strang splitting in our applications since it is not supported by CLAWPACK in 3D for dimensional splitting. That means we would not be able to use dimensional splitting for advection, which reduces code performance. A full 3D solve of the hyperbolic operator is more expensive than the dimensional split version and we often need to have a fine mesh. This is not of much concern since the contaminant deposition problem, by nature, is accompanied by large measurement errors for both the pollutant concentration and the model parameters. Therefore, the extra accuracy of a higher order scheme is not necessarily useful.

3.1.3 Boundary Conditions and Split Operators

One of the main difficulties in applying split operators arises in identifying proper boundary conditions for the split steps. In Section 17.9 of [25], LeVeque presents an example of an advection-reaction equation in 1D with Dirichlet boundary conditions:

$$\begin{aligned}&\omega_t + u\,\omega_x = -\beta\omega,\\ &\omega(0,t) = 1,\end{aligned} \tag{3.1.18}$$

where u is the advection speed and β is the reaction rate. A similar example with convergence results is also considered by Hundsdorfer [18].

Suppose that we split the advection and reaction terms by solving the following set of equations over a time step from t_n to t_{n+1}

$$\begin{aligned} \omega_t^* &= -u\,\omega_x^* \quad \text{subject to} \quad \omega^*(0,t) = 1,\ \omega^*(x,t_n) = \omega(x,t_n), \\ \omega_t^{**} &= \beta\omega^{**} \quad \text{subject to} \quad \omega^{**}(0,t) = 1,\ \omega^{**}(x,t_n) = \omega^*(x,t_{n+1}). \end{aligned} \tag{3.1.19}$$

Assume that the problem is discretized in space using a finite volume scheme (see Section 3.1.6). Let us take the Courant Friedrichs Lewy (CFL) number $\nu = u\frac{\Delta t}{\Delta x} = 1$ (see Section 3.4.2), so that the solution operator for the advection equation for the first cell adjacent to the boundary is

$$W_1^* = W_0^n, \tag{3.1.20}$$

where W_0^n is a hypothetical cell which is utilized to impose the boundary condition (see Section 3.3.2 for more details). Then applying the reaction solution operator gives

$$W_1^{n+1} = e^{-\beta\Delta t} W_1^*, \tag{3.1.21}$$

and putting these steps together we have

$$W_1^{n+1} = e^{-\beta\Delta t} W_0^n, \tag{3.1.22}$$

which is the exact solution operator with no splitting error. Since a constant boundary condition is prescribed for the ghost cell, we have $W_0^n = 1$. Then the predicted cell average value is

$$W_1^{n+1} = e^{-\beta\Delta t} \overset{CFL}{=\!=} e^{-\frac{\beta}{u}\Delta x} = 1 - \frac{\beta}{u}\Delta x + \mathcal{O}(\Delta x^2), \tag{3.1.23}$$

while the actual cell average from the exact solution is

$$\frac{1}{\Delta x}\int_{cell} e^{-\frac{\beta}{u}\Delta x} dx = 1 - \frac{\beta}{2u}\Delta x + \mathcal{O}(\Delta x), \tag{3.1.24}$$

where an $\mathcal{O}(\Delta x)$ error is introduced. Assuming that a similar error arises when a diffusive process is added to the equation, then the error would propagate through the whole solution implicitly and our scheme would return to being first order. Also note that the first order error introduced by splitting is only detectable by comparing the numerical solution against the analytic solution. That is, doing a numerical convergence study by comparing the solution over successively finer grids will not uncover this source of error.

3.1.4 Splitting Advection, Diffusion and Sources

Now that the basic ideas surrounding splitting schemes are laid out, we will next present an overview of the splitting method that we will actually employ in this project.

Pollution dispersion in the atmosphere is advection-dominated in the sense that the diffusive effects in (2.1.2) are small compared to the advection. For this reason, it is desirable to use a sophisticated hyperbolic solver to treat the advective term, and we will use the CLAWPACK solver for this purpose. CLAWPACK also uses Strang splitting to handle source terms by default and we develop our own approach for treating the diffusion term. Solving both the advection and diffusion terms in 3D is not cheap even when using splitting and it can take a very long time to compute for very fine grids. Therefore, we employ dimensional splitting for both advection and diffusion, which reduces the linear algebra to 1D solves in each direction. This is very desirable since we end up with an $\mathcal{O}(N)$ algorithm rather than $\mathcal{O}(N^3)$ (where N denotes the number of grid cells in each direction). Of course this type of splitting is highly susceptible to splitting error and since CLAWPACK 4.3 only works with Godunov dimensional splitting in 3D, we should not expect to see better than first order accuracy in time. Choosing a CFL number close to 1 means that the time error will be equivalent to a first order spatial error because Δt will be on the order of Δx. An overview of our algorithm is as follows:

1. Read data, initialize and factor diffusion matrices.
2. Compute source contributions over a half time step.
3. Solve the advection terms using CLAWPACK's wave propagation algorithm, sweeping in x then y then z and also choose Δt if adaptive time stepping is employed.
4. Update and factor the diffusion matrices if adaptive time stepping is employed.
5. Apply the diffusion terms, sweeping in x then y and then z.
6. Compute the source contributions over another half time step.
7. Return to step 2.

3.2 Finite Volume Methods and Conservation Laws

Finite volume schemes are commonly used for the solution of conservation laws in the engineering and mathematical communities. They are also used in many commercial softwares such as FLUENT or open source packages such as OpenFOAM and CLAWPACK. For an extensive treatment of finite volume schemes and an introduction to the CLAWPACK package see the monograph by LeVeque [25] or the CLAWPACK manual [26]. There are a vast number of references on finite volume schemes including the book by Hirsch [17] and the monograph on computational fluid dynamics by Chung [8], both of which consider finite volume methods from an engineering point of view. We will start by introducing the basic concept of a finite volume scheme while heavily depending on [25].

Let us revisit the homogeneous form of equation (2.1.1) in one dimension,

$$c_t + f_x = 0 \tag{3.2.1}$$

with some prescribed boundary conditions. This PDE can be viewed as a conservation law for c where the function $f(c(x,t))$ is the *flux* and $c(x,t)$ is the *conserved quantity*. The theory of conservation laws is very well developed (see chapter 11 of [12] and chapters 2 and 3 of [25]). A finite volume scheme approximates the integral of the solution of (3.2.1) over finite intervals, often called *finite volumes* or *grid cells*. Intuitively, the method attempts to conserve the quantity c over the volume of each cell. Denote the i^{th} cell by the interval $\mathcal{C}_i = [x_{i-1/2}, x_{i+1/2}]$ of length Δx. Integrate (3.2.1) over the grid cell $\mathcal{C}_i$ to obtain the following integral form of the conservation law

$$\frac{d}{dt}\int_{\mathcal{C}_i} c(x,t)dx = f(c(x_{i-1/2},t)) - f(c(x_{i+1/2},t)). \tag{3.2.2}$$

Let the solution in this cell after n time steps be denoted by C_i^n which is defined as the average of the exact solution over the cell $\mathcal{C}_i$,

$$C_i^n \approx \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} c(x,t_n)dx. \tag{3.2.3}$$

Consider integrating both sides of (3.2.2) over a time interval $[t_n, t_{n+1}]$ with $t_{n+1} = t_n + \Delta t$. Then we have

$$\int_{\mathcal{C}_i} c(x,t_{n+1})dx - \int_{\mathcal{C}_i} c(x,t_n)dx = \int_{t_n}^{t_{n+1}} f(c(x_{i-1/2},t))dt - \int_{t_n}^{t_{n+1}} f(c(x_{i+1/2},t))dt. \tag{3.2.4}$$

Divide (3.2.4) by Δx and use (3.2.3) to get

$$C_i^{n+1} - C_i^n = \frac{1}{\Delta x}\left[\int_{t_n}^{t_{n+1}} f(c(x_{i-1/2},t))dt - \int_{t_n}^{t_{n+1}} f(c(x_{i+1/2},t))dt\right], \tag{3.2.5}$$

which is a rule for updating the approximation C_i^n. The next step is to approximate the integrals on right hand side of (3.2.5). Define

$$F_{i-1/2}^{n+1/2} \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f(c(x_{i-1/2},t))dt \tag{3.2.6}$$

as a numerical approximation of the flux integral. Putting equations (3.2.5) and (3.2.6) together we have a formula to update values of the approximate solution in a manner that mimics the conservative form of the original PDE,

$$C_i^{n+1} = C_i^n - \frac{\Delta t}{\Delta x}\left(F_{i+1/2}^{n+1/2} - F_{i-1/2}^{n+1/2}\right). \tag{3.2.7}$$

Figure 3.1 depicts the configuration of the discretized solution in 1D. Since we are looking for a scheme that will update the approximate solution using previous approximations then we need to write the approximate fluxes as a function of the solutions C_i^n. A simple case would be to use the cell average values on either side of the face to write

$$F_{i-1/2}^{n+1/2} = \mathcal{F}(C_{i-1}^n, C_i^n) \tag{3.2.8}$$

where $\mathcal{F}$ is called the *numerical flux*. This definition of the numerical flux guarantees the conservation of C within the domain of solution. The choice of the numerical flux determines the accuracy and order of our finite volume scheme.

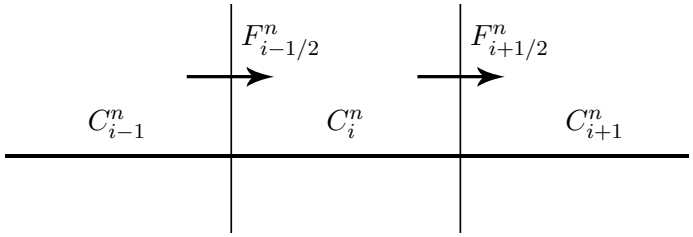


Figure 3.1: Schematic of the finite volume discretization of the conserved quantity c and the fluxes mentioned in (3.2.7).

3.3 Treating Diffusion with an Implicit Solver

In this section we explain how the diffusion term in (2.1.2) is discretized with an implicit finite volume scheme. We consider how the variable diffusion coefficients are treated and discuss the discretization of the boundary conditions.

3.3.1 Diffusion Flux and Variable Diffusion Coefficients

With the basic idea of finite volume methods laid out we can now discretize the diffusion term. Consider the 1D diffusion equation

$$c_t - (K(x)c_x)_x = 0 \tag{3.3.1}$$

with prescribed initial and boundary conditions, which is a parabolic PDE. We assume that the diffusion coefficient $K(x)$ is a function of position but not time. Comparing (3.3.1) and (3.2.1), the flux f for diffusion can be written as

$$f(c(x,t)) = -K(x)c_x(x,t). \tag{3.3.2}$$

To approximate the flux in cell $\mathcal{C}_i$, we use a simple centered difference at the cell interface to obtain the derivative and take $K_{i-1/2} = K(x_{i-1/2})$ which gives a formula for the numerical flux as

$$\mathcal{F}(C_{i-1}, C_i) = -K_{i-1/2}\left(\frac{C_i - C_{i-1}}{\Delta x}\right). \tag{3.3.3}$$

Since we are not interested in using an explicit scheme for the parabolic PDE (3.3.1) due to severe time step restrictions, we define

$$F^n_{i-1/2} = \mathcal{F}(C^{n+1}_{i-1}, C^{n+1}_i) \tag{3.3.4}$$

which we can substitute in (3.2.7) to obtain an implicit finite volume scheme

$$C^{n+1}_i = C^n_i + \frac{\Delta t}{\Delta x^2}\left[K_{i+1/2}(C^{n+1}_{i+1} - C^{n+1}_i) - K_{i-1/2}(C^{n+1}_i - C^{n+1}_{i-1})\right]. \tag{3.3.5}$$

This is the familiar second order centered difference scheme that is commonly used in finite difference approximations. Rearranging the terms in (3.3.5) to keep the implicit terms on the left hand side and the explicit terms on the right, we can see the form of the matrix that will be used in the implicit solve

$$\begin{aligned}\left(-\frac{\Delta t}{\Delta x^2}K_{i+1/2}\right)C^{n+1}_{i+1} + \left(1 + \frac{\Delta t}{\Delta x^2}(K_{i+1/2} + K_{i-1/2})\right)C^{n+1}_i + \\ \left(-\frac{\Delta t}{\Delta x^2}K_{i-1/2}\right)C^{n+1}_{i-1} = C^n_i.\end{aligned} \tag{3.3.6}$$

Therefore, each time step requires inversion of a sparse tridiagonal matrix, which is in general non-symmetric when $K(x)$ varies with position.

3.3.2 Boundary Conditions

So far we have only approximated the solution for interior points that are not adjacent to the boundary. Extra work is required to implement the boundary conditions with the same order of accuracy as the interior equations. We will describe the discretization of three types of boundary conditions (Dirichlet, Neumann and Robin) although we are more interested in the Robin condition which will be used to model the ground deposition.

Dirichlet BC

Let us focus our attention on the boundary cells C_1^{n+1} and C_N^{n+1}. The Dirichlet boundary condition for the PDE (3.3.1) over a domain $\Omega = [a, b]$ dictates

$$\begin{cases} c(a) = l(t), \\ c(b) = r(t), \end{cases} \tag{3.3.7}$$

where the left and right boundary conditions $l(t)$ and $r(t)$ are given functions of time. To satisfy the Dirichlet conditions in discrete form we make use of *ghost* cells. Consider a hypothetical cell C_0^{n+1} to the left of C_1^{n+1} as depicted in Figure 3.2. Now the boundary of the domain is the left interface of $\mathcal{C}_1$ and this is where we need to satisfy the boundary conditions. We approximate the solution at this point by the average of the cell centered values and set it equal to the Dirichlet boundary condition,

$$C_{1/2}^{n+1} = \frac{C_0^{n+1} + C_1^{n+1}}{2} = l(t_{n+1}). \tag{3.3.8}$$

Similarly for the right boundary we have

$$C_{N+1/2}^{n+1} = \frac{C_N^{n+1} + C_{N+1}^{n+1}}{2} = r(t_{n+1}), \tag{3.3.9}$$

where C_{N+1}^{n+1} is the hypothetical value. Solving for the ghost values in equations (3.3.8) and (3.3.9) and then substituting back in (3.3.6), we have a closed system of equations. This process gives a second order approximation to the Dirichlet boundary condition according to LeVeque [25].

Neumann BC

The next well known form of boundary conditions is the Neumann condition given by

$$\begin{cases} -K(a)\dfrac{\partial c}{\partial x}(a) = l(t), \\ K(b)\dfrac{\partial c}{\partial x}(b) = r(t), \end{cases} \tag{3.3.10}$$

in the 1D case. Again, we will derive the boundary condition for the cell on the far left of the domain and the same argument can be used for the cell adjacent to the right boundary. Consider the hypothetical value C_0^n, but this time we make use of this value to approximate the derivative of the solution at the left interface of $\mathcal{C}_1$. We discretize the first condition in (3.3.10) using centered differences to get

$$\frac{C_1^{n+1} - C_0^{n+1}}{\Delta x} = -\frac{1}{K(a)} l(t_{n+1}), \tag{3.3.11}$$

and similarly for the right boundary

$$\frac{C_{N+1}^{n+1} - C_N^{n+1}}{\Delta x} = -\frac{1}{K(b)} r(t_{n+1}). \tag{3.3.12}$$

Putting equations (3.3.11), (3.3.12) and (3.3.6) together we have a closed system of equations for the solution at the cell centers.

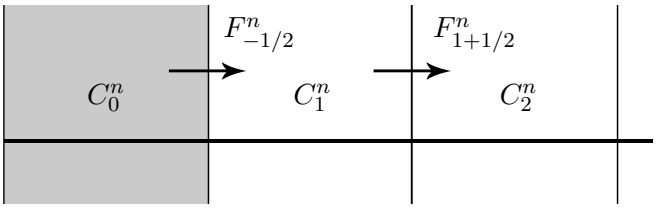


Figure 3.2: Schematic of a ghost cell (shaded) for implementation of the boundary conditions at the left boundary.

Robin

With the Dirichlet and Neumann boundary conditions laid out, we can extend our implementation of these conditions to the mixed Robin boundary condition,

$$\begin{cases} \alpha c(a) - K(a)\dfrac{\partial c}{\partial x}(a) = l(t), \\ \beta c(b) + K(b)\dfrac{\partial c}{\partial x}(b) = r(t), \end{cases} \tag{3.3.13}$$

where α and β are constants. Again we employ the idea of ghost cells to discretize this boundary condition. We use centered differences to discretize the derivative and write out a fully implicit formula that can be substituted in (3.3.6) to give the value in the ghost cell based on interior points. For the boundary condition on the left side of the interval we have

$$\alpha\frac{C_1^{n+1}+C_0^{n+1}}{2}-K(a)\left(\frac{C_1^{n+1}-C_0^{n+1}}{\Delta x}\right)=l(t_{n+1}), \tag{3.3.14}$$

and similarly for the right boundary,

$$\beta\frac{C_{N+1}^{n+1}+C_N^{n+1}}{2}-K(b)\left(\frac{C_{N+1}^{n+1}-C_N^{n+1}}{\Delta x}\right)=r(t_{n+1}). \tag{3.3.15}$$

Similar to the previous cases, equations (3.3.14) and (3.3.15) will close the system given by (3.3.6). In particular we are interested in homogeneous Robin conditions with $l(t)=r(t)=0$ with the freedom to choose the coefficients α,β.

3.4 Advection Solver

Now we consider the discretization of the advective term which involves solving a problem of the form

$$c_t+\nabla\cdot(\vec{u}c)=0, \tag{3.4.1}$$

with prescribed boundary conditions and a given velocity field $\vec{u}(x)$. Going back to the form of (3.2.1), it is clear that this is a conservation law with flux

$$f(c)=\vec{u}c. \tag{3.4.2}$$

Note that this form of the flux is indeed more general in the sense that it also incorporates variable velocity fields that depend on time or space. For now we consider a constant velocity field so that we can provide some basic definitions. Later we will generalize our arguments to the case of variable velocity field. The main reference for the material in this section is the monograph by LeVeque [25] which has a detailed treatment of hyperbolic conservation laws.

3.4.1 Characteristics and Riemann Problems

Since we are eventually interested in dimensional splitting, let us consider the 1D version of (3.4.1) with given initial condition over the whole of $\mathbb{R}$ and positive advection speed $u>0$

$$c_t+uc_x=0,\qquad c(x,0)=g(x). \tag{3.4.3}$$

We can make the simple change of variable $\xi = x - ut$ to obtain

$$\frac{dc}{d\xi} = -\frac{1}{u}c_t + c_x = 0, \tag{3.4.4}$$

which means that the solution is constant along the *characteristic curve* $x = ut + C$ where C is a constant. This suggests that the solution to the advection equation is simply the initial condition shifted along the characteristics, that is

$$c(x,t) = g(x - ut). \tag{3.4.5}$$

A special case of the hyperbolic problem (3.4.3) with a jump initial condition

$$g(x) = \begin{cases} c_l & \text{for } x < 0, \\ c_r & \text{for } x > 0, \end{cases} \tag{3.4.6}$$

is called the *Riemann problem* corresponding to (3.4.3), where c_l and c_r are given constant values. This is solved by simply propagating the jump discontinuity along characteristics.

3.4.2 Finite Volume Formulation

Following the same conventions as Section 3.2.1, we can discretize the hyperbolic equation in (3.4.3). Many choices can be made for the numerical flux which affect the accuracy and stability of the numerical scheme. The simplest choice is the first order *upwind flux* which uses a first order approximation of the spatial derivative of c to give

$$F^n_{i-1/2} = uC^n_{i-1}, \tag{3.4.7}$$

that allows us to rewrite (3.2.6) as

$$C^{n+1}_i = C^n_i - \frac{u\Delta t}{\Delta x}\left(C^n_i - C^n_{i-1}\right). \tag{3.4.8}$$

This scheme is stable when $\nu = \frac{u\Delta t}{\Delta x} \leq 1$, which is called the *CFL condition* named after Courant-Friedrichs-Lewy [25]. Intuitively, the CFL condition dictates that flow of information in the discretized hyperbolic system should mimic the flow of information in the actual PDE. In other words, the discrete domain of dependence of the solution must contain the analytical domain of dependence [25].

To understand how the upwind method makes use of the solution to a Riemann problem, take the solution at the current time step C^n_i to be exact in all cells. We want to solve the

PDE with a piecewise constant initial condition as pictured in Figure 3.3a. Based on the initial data there is a Riemann problem at each cell interface with a jump $\Delta C^n_{i+1/2} = C^n_{i+1} - C^n_i$ across each cell face. Updating the solution over a time increment Δt means that the discontinuities must move along characteristics a distance $u\Delta t$ into the next cell (Figure 3.3b). Assuming the CFL condition is satisfied, the discontinuity can only move as far as the next cell interface otherwise characteristics from neighboring cells cross and solution will not be unique anymore. So far the solution is updated exactly but since the numerical scheme can only see the solution as averages over the cells, we have to take the average of the new solution over each cell which gives the updated solution as shown in Figure 3.3c. The variation in the average cell value after this step is exactly the second term on the right hand side of (3.4.8). This image helps to understand why we prefer to keep the CFL number close to one in order to reduce the smearing of discontinuities.

A large collection of higher order variants of this method is available such as the *Lax-Wendroff* method,

$$C^{n+1}_i = C^n_i - \frac{u\Delta t}{2\Delta x}(C^n_{i+1} - C^n_{i-1}) + \frac{1}{2}\left(\frac{u\Delta t}{\Delta x}\right)(C^n_{i-1} - 2C^n_i + C^n_{i+1}). \tag{3.4.9}$$

These high resolution schemes tend to give oscillatory solutions near discontinuities. Hence they are effective when the solution is smooth. In chapter 6 of [25] a detailed treatment of flux limiters and high resolution schemes is presented. We will not discuss these methods since they constitute a vast topic deserving of an extensive amount of time and background.

3.4.3 Variable Coefficients

So far we have only considered a constant coefficient hyperbolic system, whereas the pollutant advection problem with velocity field correction and time dependent wind is a variable coefficient problem following our discussions in Chapter 2. The case of a time dependent wind $u(t)$ is rather easy to deal with, since (3.4.1) becomes

$$c_t + (u(t)c)_x = c_t + u(t)c_x = 0. \tag{3.4.10}$$

The only difference in our numerical scheme is that at each time step we have to update the advection speed. Hence, the upwind scheme reads

$$C^{n+1}_i = C^n_i - \frac{u(t_n)\Delta t}{\Delta x}\left(C^n_i - C^n_{i-1}\right). \tag{3.4.11}$$

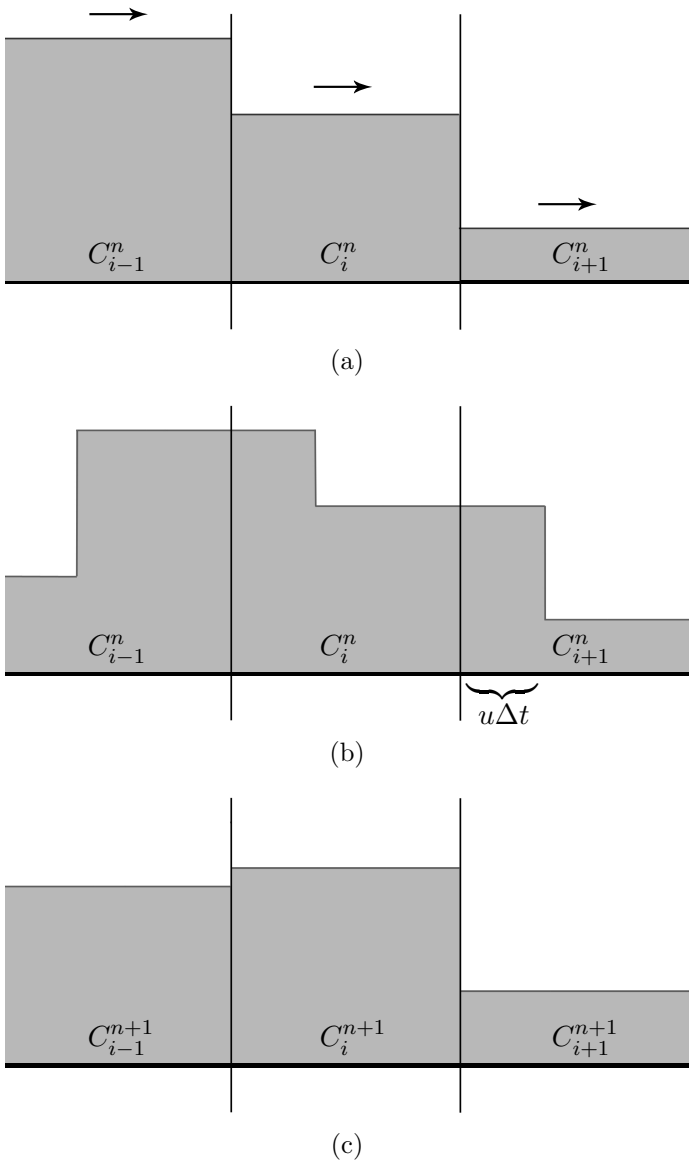


Figure 3.3: Schematic depiction of the solution from the upwind scheme. (a) The solution at time t_n. (b) Taking the solution at t_n as the initial condition for a Riemann problem at each cell interface we get the exact solution at t_{n+1} by shifting the initial condition a distance $u\Delta t$. (c) Taking the average of the new solution at each cell yields the numerical solution at the next time step.

It is important to note that in this case the Courant number ($\nu = u\Delta t/\Delta x$) changes as the velocity varies in time. This means that we must either use adaptive time stepping or else choose a time step for the largest velocity value (which could generate excessive smearing when the velocity is small).

In the case of a spatially dependent advection speed we have the PDE,

$$c_t + (u(x,t)c)_x = 0. \tag{3.4.12}$$

Using (3.2.7) and defining the flux as

$$F_{i-1/2}^{n+1/2} = u(x_{i-1}, t_n)C_{i-1}^n, \tag{3.4.13}$$

we can write the upwind scheme as

$$C_i^{n+1} = C_i^n - \frac{\Delta t}{\Delta x}\left(u(x_i,t_n)C_i^n - u(x_{i-1},t_n)C_{i-1}^n\right). \tag{3.4.14}$$

3.5 Point Sources

In this section we consider the numerical treatment of point sources since these are the most common type of sources used in atmospheric dispersion applications. The physical dimension of pollution sources is typically much smaller than the discrete grid sizes used and therefore they can be considered as point sources or delta distributions.

3.5.1 Delta Distribution and Action on Test Functions

The reader is encouraged to consult the monograph by Rudin [38] for a detailed definition and treatment of the theory of distributions and test functions. Here we will only present a very brief introduction to these subjects.

We define a *test function* in a domain $\Omega \subseteq \mathbb{R}$ as an infinitely differentiable and compactly supported function $\phi(x) \in \mathbb{C}_c^\infty(\Omega)$ such that $\partial^\alpha \phi$ vanishes outside a compact set K for all multi-indices α and supp $\phi \subseteq K$ for any set $K \Subset \Omega$. Let $\mathcal{D}(\mathbb{R})$ denote the space of test functions over the whole real line, and denote by $\mathcal{D}^*(\mathbb{R})$ the space of distributions. Now consider a sequence of test functions $\{\phi_j\} \in \mathcal{D}(\mathbb{R})$ such that

$$\partial^\alpha \phi_j \to 0 \quad \text{uniformly on } K \text{ as } j \to \infty,$$

for all multi-indices α. Then a linear functional T belongs to $\mathcal{D}^*(\mathbb{R})$ and hence is a distribution if and only if

$$T(\phi_j) \to 0 \quad \text{as } j \to \infty.$$

In other words distributions are elements that act like bounded linear functionals over test functions. Note that distributions, in general, are very different from conventional functions. For example, they do not need to be single valued or defined pointwise. Therefore we should keep in mind that they are only defined via their action on test functions. Of particular interest to our work here is the delta distribution defined as

$$\langle \delta, \phi \rangle := \delta(\phi) = \int_{\mathbb{R}} \delta(x)\phi(x)\, dx = \phi(0), \tag{3.5.1}$$

which selects a point value of a smooth function. Note that we have employed the Riesz representation theorem [38] in our notation in order to write the action of the delta distribution as an integral. Having this definition in mind, it is obvious that we cannot discretize this distribution directly and so we need to approximate the delta distribution with another distribution that can be represented as a function. Once we have such an approximation, then it can be discretized and built into a numerical scheme.

3.5.2 Moment Conditions

Consider approximating the delta distribution with a sequence of distributions $\delta_h(x)$ that converges to the delta distribution as $h \to 0$ in the sense of (3.5.1). That is, find $\delta_h \in \mathcal{D}^*(\Omega)$ such that $\forall \phi \in \mathcal{D}(\Omega)$

$$\lim_{h\to 0} \int_{\mathbb{R}} \delta_h \phi\, dx \to \int_{\mathbb{R}} \delta\phi\, dx = \phi(0), \tag{3.5.2}$$

so that action of the approximation δ_h is similar to that of the delta distribution in the limit of small h. Now consider expanding $\phi(x) \in \mathcal{D}(\Omega)$ about $x = 0$ using Taylor's theorem. Since ϕ is infinitely differentiable we have

$$\begin{aligned} \langle \delta_h, \phi \rangle &= \int_{\mathbb{R}} \delta_h \left(\phi(0) + x\phi_x(0) + \frac{x^2}{2}\phi_{xx}(0) + \cdots \right) dx \\ &= \int_{\mathbb{R}} \delta_h \phi(0)\, dx + \int_{\mathbb{R}} \delta_h x \phi_x(0)\, dx + \int_{\mathbb{R}} \delta_h \frac{x^2}{2}\phi_{xx}(0)\, dx + \cdots \end{aligned} \tag{3.5.3}$$

Of particular interest are forms of δ_h that have compact support on an interval $[-h/2, h/2]$, in which case (3.5.3) takes the form

$$\langle \delta_h, \phi \rangle = \int_{-h/2}^{h/2} \delta_h \phi(0)\, dx + \int_{-h/2}^{h/2} \delta_h x \phi_x(0)\, dx + \int_{-h/2}^{h/2} \delta_h \frac{x^2}{2}\phi_{xx}(0)\, dx + \cdots \tag{3.5.4}$$

Now if we look at the error between the action of the approximate distribution and the original delta distribution we see that

$$\begin{aligned}\langle \delta_h, \phi \rangle - \langle \delta, \phi \rangle =& \phi(0) \left(\int_{-h/2}^{h/2} \delta_h \, dx - 1 \right) + \phi_x(0) \left(\int_{-h/2}^{h/2} \delta_h x \, dx \right) \\ &+ \frac{\phi_{xx}(0)}{2} \left(\int_{-h/2}^{h/2} \delta_h x^2 \, dx \right) + \cdots \end{aligned} \tag{3.5.5}$$

This way of expressing the residual makes clear what conditions should be satisfied in order to minimize the error. Roughly speaking each term in (3.5.5) involves integration over higher powers of x, which represent the different orders of convergence. That is, for each integral in (3.5.5) we have

$$\left| \int_{-h/2}^{h/2} \delta_h x^m \, dx \right| \leq \int_{-h/2}^{h/2} |\delta_h| \, |x^m| \, \, dx \leq h^m \int_{-h/2}^{h/2} |\delta_h| \, \, dx = \mathcal{O}(h^m). \tag{3.5.6}$$

Note that the absolute value of δ_h only makes sense in terms of the function associated with this distribution. Setting the terms inside the brackets in (3.5.5) to zero we obtain the following sequence of conditions that should be satisfied by δ_h

$$\begin{aligned} \int_{-h/2}^{h/2} \delta_h \, dx &= 1, \\ \int_{-h/2}^{h/2} x \delta_h \, dx &= 0, \\ \int_{-h/2}^{h/2} x^2 \delta_h \, dx &= 0, \\ &\vdots \end{aligned} \tag{3.5.7}$$

which are known as the moment conditions. It is interesting to note that the first condition states that the approximate delta distribution should integrate to 1 over its support, while the second condition forces δ_h to be even. It is easy to see that a piecewise constant function

$$\delta_h := \begin{cases} 1/h, & \text{for } |x| \leq h, \\ 0, & \text{otherwise}, \end{cases} \tag{3.5.8}$$

satisfies the first two conditions and hence yields second order convergence in the sense of (3.5.6). This choice is not unique and so if we also want continuity, then we can choose a

hat function or the cosine approximation given by

$$\delta_h := \begin{cases} \frac{1}{2h}\left(1+\cos\left(\frac{2\pi x}{h}\right)\right), & \text{for } |x| \le h, \\ 0, & \text{otherwise.} \end{cases} \tag{3.5.9}$$

The only conclusion we can draw from the third condition is that δ_h cannot be positive over its whole support. To satisfy more than the first two conditions in (3.5.8) is a difficult task in general.

3.5.3 Solving the Moment Problem

Finding a function that satisfies all the conditions of (3.5.7) is an *infinite dimensional* moment problem whereas satisfying a finite number of these conditions results in the *finite dimensional* moment problem. For an introduction to moment problems and their solution, see section 5.3 of [21]. This is an ill-posed problem with conditional existence or uniqueness of solution. Clearly the finite dimensional case lacks uniqueness of solution, since any symmetric function that integrates to 1 can satisfy the first two conditions. Since our numerical method is only first order, we are already satisfied with the piecewise constant approximation in (3.5.8). But we take a short detour here to derive a higher degree approximation to the δ that satisfies the third moment condition as well. This is an exercise in solving ill-posed integral equations. Consider summarizing the conditions in (3.5.7) as

$$(\delta_h, \psi_n) = \beta_n, \qquad n \in \mathbb{N}, \tag{3.5.10}$$

for functions ψ_n that we assume are polynomials whose degree increases with n. In order to find a solution to this problem we need the ψ_n to be linearly independent. We also need a basis for the ψ_n that is orthogonal with respect to the inner product

$$(f_1, f_2) = \int_{-1}^{1} f_1(x) f_2(x) dx, \tag{3.5.11}$$

which is the familiar L^2 inner product. We restrict our attention to functions that are supported on the interval $[-1, 1]$ and note that a simple rescaling will convert the solution so that it is supported on $[-h/2, h/2]$. Using the L^2 inner product as well as the first two conditions in (3.5.7) suggests choosing the Legendre polynomials as a basis; which are given

by the Rodriguez formula

$$\begin{aligned} P_m(x) &= \frac{1}{2^m m!}\frac{\partial^m}{\partial x^m}\left((x^2-1)^m\right), \\ \|P_m\|_{L^2} &= \frac{2}{2m+1}, \quad \text{for} \quad m = 0, 1, 2, ... \end{aligned} \tag{3.5.12}$$

Expanding the first three ψ_n in terms of the Legendre basis gives

$$\begin{aligned} \psi_0(x) &= 1 = P_0(x), \\ \psi_1(x) &= x = P_1(x), \\ \psi_2(x) &= x^2 = \frac{2}{3}P_2(x) + \frac{1}{3}P_0(x). \end{aligned} \tag{3.5.13}$$

We can also represent the approximation $\delta_h(x)$ in terms of the basis polynomials as

$$\delta_h(x) = a_0 P_0(x) + a_1 P_1(x) + a_2 P_2(x). \tag{3.5.14}$$

Then, substituting (3.5.13) and (3.5.14) into (3.5.10) and using orthogonality we have

$$\begin{aligned} a_0 \|P_0\| &= 1, \\ a_1 \|P_1\| &= 0, \\ \frac{a_0}{3}\|P_0\| + \frac{2a_2}{3}\|P_2\| &= 0, \end{aligned} \tag{3.5.15}$$

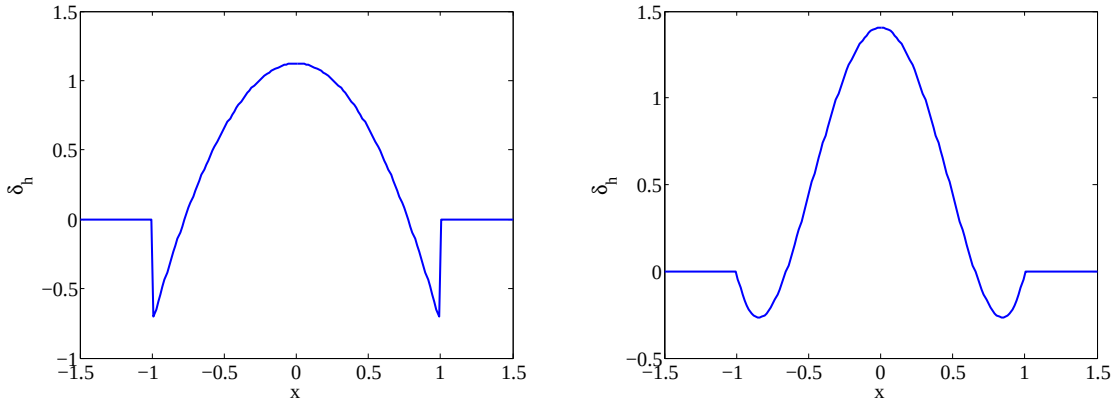
which can be easily solved to obtain

$$a_0 = \frac{1}{2}, \quad a_1 = 0, \quad a_2 = -\frac{5}{4}. \tag{3.5.16}$$

This solution clearly satisfies the first three moment conditions and is pictured in Figure 3.4a. Note that we could add an arbitrary number of terms to (3.5.14) using higher order Legendre polynomials and the resulting function would still satisfy the desired moment conditions–this is another way of showing the lack of uniqueness in the solution. Since the function is only supported on $[-1, 1]$, the solution in Figure 3.4a is clearly not continuous, but we can take advantage of the higher order Legendre polynomials to get a solution that is as smooth as we want. Consider adding a term involving $P_4(x)$ to the solution (3.5.14) so that

$$\delta_h(x) = \frac{1}{2}P_0(x) - \frac{5}{4}P_2(x) + a_4 P_4(x), \tag{3.5.17}$$

Now we can evaluate this function at the end points and set $a_4 = 3/4$ to get the symmetric function pictured in Figure 3.4b. Note that we could also add a multiple of $P_6(x)$ or any other higher order Legendre polynomial with an even index, and the solution would still satisfy the same requirements. This framework allows us to create approximations to the delta distribution that have arbitrary order of accuracy.



(a) (b)

Figure 3.4: (a) Approximate delta distribution with the first three moment conditions. (b) Approximation is made continuous by adding a multiple of $P_4(x)$.

3.5.4 Discretizing the Source Term

Once an analytical approximation to the delta distribution is obtained we can move on to discretizing the approximation in our numerical scheme. For simplicity, we consider the simple equation

$$c_t(x) = s(x), \tag{3.5.18}$$

which is actually the stage in our split step scheme that applies the source term. For simplicity, consider the 1D case with a source that is constant, so s is only a function of the spatial coordinates. To discretize this equation, all we need to do is integrate the source term over each grid cell to obtain

$$C_i^{n+1} = C_i^n + \frac{\Delta t}{\Delta x} \int_{\mathcal{C}_i} s(x)\, dx. \tag{3.5.19}$$

Now our approximation of the integral on the right hand side dictates the spatial accuracy of this formula. We will use the composite trapezoidal rule, which is formally $\mathcal{O}(\Delta x^2)$, to approximate source terms that have support independent of the mesh size (see for example case 4 in section 3.5.2 where we consider a smooth cosine function).

Another case of the source terms are delta distributions that have support of the order of the mesh size; for example, see the function defined in (3.5.20) below. Note that in this case the composite trapezoid rule does not give satisfactory results because the function is

supported on a very small interval and we do not gain accuracy by refining the mesh. To remedy this problem we compute the exact value of the integral in (3.5.19) and build the answer into our numerical scheme. As an example, consider the delta source term centered at x_s that was previously defined in (3.5.8). Also suppose the solution is discretized over a grid of size Δx so that the source term can be written as

$$\delta_h(x) = \begin{cases} \dfrac{1}{\Delta x}, & \text{for } |x - x_s| \le \Delta x, \\ 0, & \text{otherwise.} \end{cases} \tag{3.5.20}$$

Assume that $x_{\ell-1/2} \le x_s \le x_{\ell+1/2}$ for some index ℓ, so that the source term is only supported on the grid cells $\mathcal{C}_{\ell-1}, \mathcal{C}_\ell, \mathcal{C}_{\ell+1}$. Assume further that $\left|x_s - x_{\ell+1/2}\right| > \left|x_s - x_{\ell-1/2}\right|$ so that the source is closer to the right interface of $\mathcal{C}_\ell$ Then we have

$$\int \delta_h(x)\, dx = \int_{\mathcal{C}_\ell} \delta_h\, dx + \int_{\mathcal{C}_{\ell+1}} \delta_h\, dx, \tag{3.5.21}$$

which indicates the contribution of the source to the solution at each cell is

$$\begin{aligned} C_\ell^{n+1} - C_\ell^n &= \frac{\Delta t}{\Delta x}\left(1 - \frac{x_s - x_\ell}{\Delta x}\right), \\ C_{\ell+1}^{n+1} - C_{\ell+1}^n &= \frac{\Delta t}{\Delta x}\left(\frac{x_s - x_\ell}{\Delta x}\right). \end{aligned} \tag{3.5.22}$$

3.6 Pollutant Deposition

The last component in our numerical scheme is the ground deposition computation. By definition, the deposition of pollutants is given by the space-time integral of the vertical flux across the boundary at ground level and over time. Denote by $g(x, y, t)$ the value of deposition at $z = 0$ which satisfies

$$g(x, y, t_{n+1}) = g(x, y, t_n) + \int_{t_n}^{t_{n+1}} W_{dep} c(x, y, 0)\, dt, \tag{3.6.1}$$

and can be discretized as

$$G_{i,j}^{n+1} = G_{i,j}^n + \frac{W_{dep}\Delta t}{2}\left(C_{i,j,1}^n + C_{i,j,0}^n\right) \tag{3.6.2}$$

where the indices $i, j = 1, ..., N$ indicate grid cells in x and y directions, the last index indicates grid cells in z direction, and the index zero refers to a ghost cell. In order to omit the value of the solution at the ghost cell we need to refer to the Robin boundary condition

$$-u_{set} c(x, y, 0) - K_z(0)\frac{\partial c}{\partial z}(x, y, 0) = -W_{dep} c(x, y, 0). \tag{3.6.3}$$

We define $\bar{W}_{dep} = W_{dep} - u_{set}$ and rewrite this equation as

$$-K_z(0)\frac{\partial c}{\partial z}(x,y,0) = -\bar{W}_{dep}c(x,y,0), \tag{3.6.4}$$

using a discretization similar to that of (3.3.14) to obtain

$$-K_z(0)\left(\frac{C^n_{i,j,1} - C^n_{i,j,0}}{\Delta x}\right) = -\bar{W}_{dep}\left(\frac{C^n_{i,j,1} + C^n_{i,j,0}}{2}\right). \tag{3.6.5}$$

We solve this equation for the ghost value to get

$$C^n_{i,j,0} = \frac{\gamma - 1}{\gamma + 1}C^n_{i,j,1}, \tag{3.6.6}$$

where $\gamma = \frac{2K_z(0)}{\Delta x \bar{W}_{dep}}$. Now we substitute this result back into (3.6.2) to obtain an explicit relationship for updating the ground deposition values

$$G^{n+1}_{i,j} = G^n_{i,j} + W_{dep}\Delta t\left(\frac{\gamma}{\gamma+1}\right)C^n_{i,j,1}. \tag{3.6.7}$$

3.7 Extension to Multiple Dimensions

In the previous sections, we laid out the main ideas behind our numerical scheme while focusing on 1D. In this section we will extend the scheme to 3D by means of directional splitting. This section can be viewed as the place where we bring all the preceding ideas together and assemble them into a 3D solver to model dispersion and deposition of pollutants as discussed in Chapter 2.

3.7.1 Internal Grid Cells

Consider the full advection diffusion equation introduced in (2.1.2), which we present in expanded form

$$\begin{aligned}\frac{\partial c}{\partial t} + \frac{\partial cu(z)}{\partial x} + \frac{\partial cv(z)}{\partial y} =& \frac{\partial}{\partial x}\left(K_x\frac{\partial c}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y\frac{\partial c}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_z(z)\frac{\partial c}{\partial z}\right) \\ &+ Q\delta(x-x_s)\delta(y-y_s)\delta(z-z_s).\end{aligned} \tag{3.7.1}$$

The velocity field is assumed to have the form $\vec{u}(x,y,z) = (u(z), v(z), u_{set})$. We assume that K_z and $\vec{u}$ depend on z as discussed in section 2.4. We are interested in splitting the

differential operators to break the problem into simpler subproblems involving advection, diffusion and the source term:

$$c^{(1)}(t_n) = c(t_n) + e^{\Delta t\left(u(z)\frac{\partial}{\partial x}+v(z)\frac{\partial}{\partial y}+u_{set}\frac{\partial}{\partial z}\right)}c(t_n), \tag{3.7.2a}$$

$$c^{(2)}(t_n) = c^{(1)}(t_n) + e^{-\Delta t\left(\frac{\partial}{\partial x}\left(K_{xx}\frac{\partial}{\partial x}\right)+\frac{\partial}{\partial y}\left(K_{yy}\frac{\partial}{\partial y}\right)+\frac{\partial}{\partial z}\left(K_{zz}(z)\frac{\partial}{\partial z}\right)\right)}c^{(1)}(t_n), \tag{3.7.2b}$$

$$c(t_{n+1}) = c^{(2)}(t_n) + \Delta t\, Q\delta(x-x_s)\delta(y-y_s)\delta(z-z_s), \tag{3.7.2c}$$

where we have suppressed the dependence c on position to keep the equations simple. There is a further splitting that separates the spatial operators in (3.7.2a) to obtain

$$c^{(1a)}(t_n) = c(t_n) + e^{\Delta t\left(u(z)\frac{\partial}{\partial x}\right)}c(t_n), \tag{3.7.3a}$$

$$c^{(1b)}(t_n) = c^{(1a)}(t_n) + e^{\Delta t\left(v(z)\frac{\partial}{\partial y}\right)}c^{(1a)}(t_n), \tag{3.7.3b}$$

$$c^{(1)}(t_n) = c^{(1b)}(t_n) + e^{\Delta t\left(u_{set}\frac{\partial}{\partial z}\right)}c^{(1b)}(t_n). \tag{3.7.3c}$$

We can solve this system using the 1D advection solvers introduced earlier. We also split (3.7.2b) to give

$$c^{(2a)}(t_n) = c^{(1)}(t_n) + e^{-\Delta t\left(\frac{\partial}{\partial x}\left(K_{xx}\frac{\partial}{\partial x}\right)\right)}c^{(1)}(t_n), \tag{3.7.4a}$$

$$c^{(2b)}(t_n) = c^{(2a)}(t_n) + e^{-\Delta t\left(\frac{\partial}{\partial y}\left(K_{yy}\frac{\partial}{\partial y}\right)\right)}c^{(2a)}(t_n), \tag{3.7.4b}$$

$$c^{(2)}(t_n) = c^{(2b)}(t_n) + e^{-\Delta t\left(\frac{\partial}{\partial z}\left(K_{zz}(z)\frac{\partial}{\partial z}\right)\right)}c^{(2b)}(t_n). \tag{3.7.4c}$$

Expressions (3.7.3) and (3.7.4) along with (3.7.2c) represent the fully split problem that we wish to solve. Discretization is done naturally using the upwind scheme (3.4.14) for advective terms, the implicit centered scheme (3.3.6) for diffusive terms and a multidimensional version of (3.5.22) for the source term at the internal grid cells away from the boundaries.

3.7.2 Boundary Conditions

Proper boundary conditions are crucial for capturing the correct physics of our problem. There are two main issues regarding boundary conditions for the 3D advection-diffusion equation (3.7.1) in the discrete setting. First, the original PDE is defined on a half space which is infinite in x, y and $z > 0$, but in our numerical scheme we consider a finite domain. Therefore, we need to prescribe boundary conditions on all faces of the domain except for the boundary at ground level, which we will treat separately. We assume the problem is advection dominated and therefore we prescribe outflow boundary conditions for the advection

solves in all directions. As for the diffusion diffusion, we prescribe homogeneous Neumann boundary conditions in the x and y directions, as well as in the positive z direction– this aims to to conserve the total mass of the contaminant.

The second issue concerns proper splitting of the Robin boundary condition (3.6.3) which incorporates the sum of the advection and diffusion fluxes:

$$u_{set}c(x,y,0) + K_z(0)\frac{\partial c}{\partial z}(x,y,0) = W_{dep}c(x,y,0).$$

Our aim here is to choose the split boundary conditions so that the total flux at the ground equals the required ground deposition flux. We prescribe outflow boundary conditions for the advection solver at $z = 0$ which gives the advective flux $f_A = u_{set}c(x,y,0)$. Next, we choose the diffusion flux so that it makes up for the rest of the vertical flux,

$$f_D = K_z(0)\frac{\partial c}{\partial z}(x,y,0) = (W_{dep} - u_{set})c(x,y,0) = \bar{W}_{dep}c(x,y,0), \tag{3.7.5}$$

which gives the proper boundary condition for the diffusion solver.

3.8 Convergence Studies

Having discussed the details of the numerical scheme, we now proceed to test the scheme to ensure convergence and the correct order of accuracy. Here we present a series of tests starting with simple settings and continuing to more complicated problems. We are generally interested in measuring the rate of convergence over the whole 3D domain as well as convergence at ground level (being the $z = 0$ plane). We use the discrete ℓ_1, ℓ_2 and ℓ_∞ norms defined as

$$\begin{aligned} \left\|\vec{V}\right\|_{\ell_q} &= \left(\frac{1}{N}\sum_{i=1}^{N}|V_i|^q\right)^{\frac{1}{q}}, \quad \text{for} \quad 1 \le q < \infty, \\ \left\|\vec{V}\right\|_{\ell_\infty} &= \max_i |V_i|, \end{aligned} \tag{3.8.1}$$

for a generic vector $\vec{V}$ of length N.

We are concerned only with studying the effect of changes in the spatial grid size since the advection solver works best with a Courant number that is close to 1, and so we have for example $\Delta t \approx \nu \Delta x$. Note that a scheme that is $\mathcal{O}(\Delta t, \Delta x^2)$ will now only be first order since $\Delta t \propto \Delta x/u$. Except for the test case of simple diffusion we will report convergence results with Courant number $\nu \approx 0.9$. This is also desirable from a performance perspective since this ensures the largest possible stable time step.

In order to measure the numerical convergence rate we measure the difference between solution values on successively finer grids. That is, we take sequences of three successive grids with number of gridpoints $N, 2N, 4N$ and estimate the error by computing the difference between the solutions

$$\vec{E_N} = \vec{C}_N - \vec{C}_{2N}(1:2:\text{end}), \tag{3.8.2}$$

where we are borrowing MATLAB's notation to emphasize that solutions are compared only at the coarse grid points. Given this definition of $\vec{E}_N$, in the limit of large N, the convergence rate λ of the numerical scheme is given by the expression

$$\lambda = \log_2 \left(\frac{\|E_N\|_{\ell q}}{\|E_{2N}\|_{\ell q}} \right). \tag{3.8.3}$$

In the studies to follow, we choose 5 grids of size N^3 with $N = [16, 32, 64, 128, 256]$ and present error norms for all grid sizes in the accompanying plots to verify that the error is in fact being reduced at the expected rate.

3.8.1 Preliminary Tests

We start with a simple advection test having no diffusion, next we consider isotropic diffusion with no advection, and finally we report on an advection-diffusion test. All three of these preliminary tests take zero source term and smooth initial conditions, and results are presented in Table 3.1. We see that the desired first order convergence rate is maintained in all test cases.

Case 1: Simple Advection

The first test case has a velocity $\vec{u} = (1, 0, 0)$ and omits the diffusion and source terms, taking $K_{x,y,z} = Q = 0$. We take the initial condition to be a smooth cosine bump centered at $(3, 0, 3)$ and having support within the cube supp $g = \{2 \le x \le 4, -1 \le y \le 1, 2 \le z \le 4\}$ given by

$$g(x, y, z) = \begin{cases} \tilde{g}(x, y, z), & \text{for } (x, y, z) \in \text{supp } g, \\ 0, & \text{otherwise,} \end{cases} \tag{3.8.4}$$

where

$$\tilde{g}(x, y, z) = \frac{1}{8} \left[1 + \cos\left(\pi(x-3)\right)\right] \cdot \left[1 + \cos\left(\pi(y)\right)\right] \cdot \left[1 + \cos\left(\pi(z-3)\right)\right]. \tag{3.8.5}$$

We apply outflow boundary conditions on all sides of the domain and take the domain to be a cube given by $\{0 \leq x, z \leq 10, -5 \leq y \leq 5\}$. Godunov splitting is employed for advection and solutions are computed up to time $t = 8.4$, which is equivalent to taking 240 time steps on the finest mesh with $N = 256$, and with a Courant number $\nu = 0.896$.

Case 2: Pure Diffusion

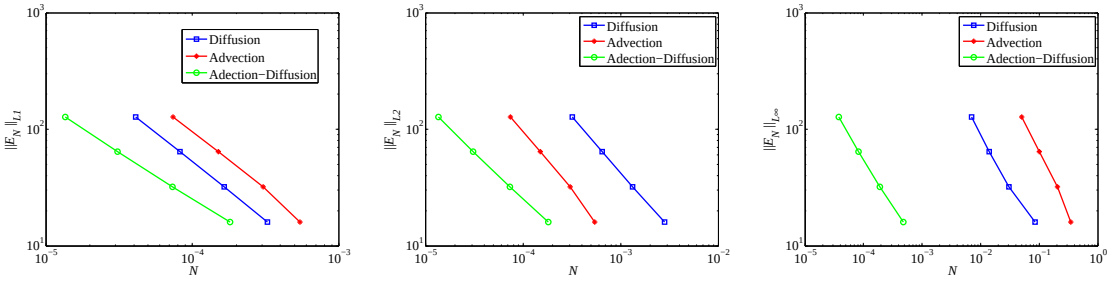
In this test we omit advection and consider only the diffusion problem. We use the same initial condition as (3.8.4) with the velocity field and the source term set to zero. Isotropic diffusion with $K_x = K_y = K_z = 0.25$ is considered, and the solution domain is the same as in case 1. We consider homogeneous Neumann boundary conditions on all faces and choose a constant time step of size $\Delta t = 0.003$, computing the solution over various grids for 300 time steps.

Case 3: Advection-Diffusion

After separately testing the advection and diffusion solvers we now consider both advection and diffusion. The domain and initial conditions are the same as in the previous two cases, the velocity field is $\vec{u} = (1, 0, 0)$ and the diffusion coefficients are taken to be $K_x = K_y = K_z = 0.25$. The Courant number is again taken to be $\nu = 0.9$ and solutions are computed for 240 time steps on the finest mesh. We choose outflow boundary conditions on all sides for the advection terms, and use homogeneous Neumann for all sides for the diffusion terms. We choose these boundary conditions since they will be used in our real world application in section 3.4.6.

	norm		
Case	ℓ_1	ℓ_2	ℓ_∞
1	1.0215	1.0131	0.9929
2	1.0091	1.0120	1.0277
3	1.1829	1.1789	1.1140

Table 3.1: Convergence study for cases 1, 2 and 3. All convergence rates are for the finest mesh and errors are computed over the whole domain.



(a) (b) (c)

Figure 3.5: (a) Plot of ℓ_1 norm of error for preliminary cases for various mesh sizes.

3.8.2 Testing with Source Terms

Next, we test the advection-diffusion solver when both smooth and delta sources are used. In the case of the delta source we present the convergence rate over the entire domain and also over a volume that is separated from the source. The reason for this choice is to determine whether convergence degrades close to the singular source. Results are presented in Table 3.2 accompanied by Figure 3.6.

Case 4: Smooth source

In this test we consider homogeneous initial conditions. All parameters are the same as in Case 3 and the solution is computed to the same time. We choose outflow boundary conditions for advection on all faces and diffusion is solved using Neumann boundary conditions. The termination time is the same as in the previous cases, which is long enough for the solution to hit one of the boundaries, and therefore we see the effect of the boundary conditions. We specify the source as a smooth cosine bump centered at $\vec{X}_s = (4, 0, 4)$ and supported on a cube of size $h = 2$, given by

$$s(x,y,z) = \begin{cases} \tilde{s}(x,y,z) & \text{for } (x,y,z) \in \text{supp } s, \\ 0 & \text{otherwise,} \end{cases} \tag{3.8.6}$$

where

$$\begin{aligned} \tilde{s}(x,y,z) = \frac{1}{8h^3} &\left[1 + \cos\left(\pi \frac{(x-x_s)}{h}\right)\right] \cdot \\ &\left[1 + \cos\left(\pi \frac{(y-y_s)}{h}\right)\right] \cdot \left[1 + \cos\left(\pi \frac{(z-z_s)}{h}\right)\right], \end{aligned} \tag{3.8.7}$$

and

$$\text{supp } s = \{|x - x_s|, |y - y_s|, |z - z_s| \leq h\}. \tag{3.8.8}$$

The trapezoid rule is employed for integration of the smooth source. Results of this convergence study is presented in Table 3.2 and we see the expected first order convergence.

Case 5: Point source

In this case we consider a source term corresponding to the 3D version of the delta distribution defined in (3.5.20)-(3.5.22) and centered at $\vec{X}_s = (4, 0, 4)$–this is a constant valued function supported on a rectangular prism of size $\Delta x \times \Delta y \times \Delta z$. Figure 3.7 pictures the plume generated by this source term at the final time.

As we see in table 3.2, the convergence in the ℓ_2 and ℓ_∞ norms is lost, which is clearly due to the singularity introduced by the delta distribution. In other words, the source blows up too fast for the solution to converge in its vicinity. Looking at the Gaussian plume solution in equation (2.2.8) we realize that even the analytic solution has a singularity at the source; therefore, this loss of convergence is expected. In order to check whether the solution is converging away from the source, we also compute the convergence rates over half of the domain where $5 \leq x \leq 10$ which is well separated from the point source. In Table 3.2 we see that the solution is converging in the ℓ_2 norm away from the source but over the entire domain, the solution is only converging in the ℓ_1 norm. Also note that the convergence rates are lower as compared to case 4 where a smooth source term was considered. Figure 3.8 shows two slices of the point wise ℓ_1 error of the solution between the two finest grids. We can see that large error occurs around the source which is consistent with our observation of the fact that convergence rate improves away from the source.

	norm		
Case	ℓ_1	ℓ_2	ℓ_∞
4	1.0275	1.0489	1.0471
5 (entire domain)	0.6267	-0.5206	-2.0227
5 (away from the source)	0.7050	0.4921	-0.7156

Table 3.2: Convergence rates for smooth and point sources (cases 4 and 5).

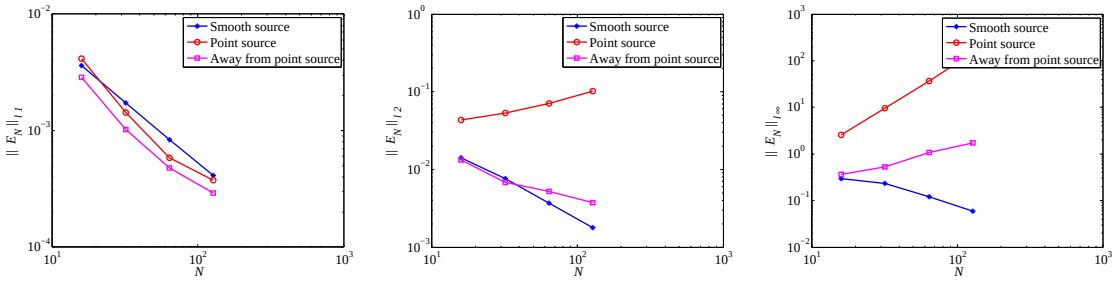


Figure 3.6: Plot of errors for cases 4 and 5 for different grid sizes in ℓ_1, ℓ_2 and ℓ_∞ norms.

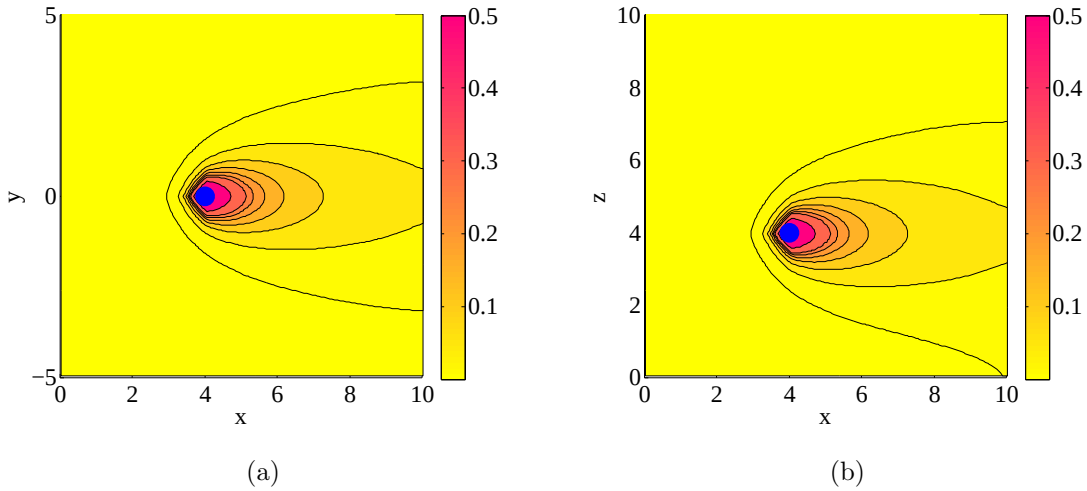
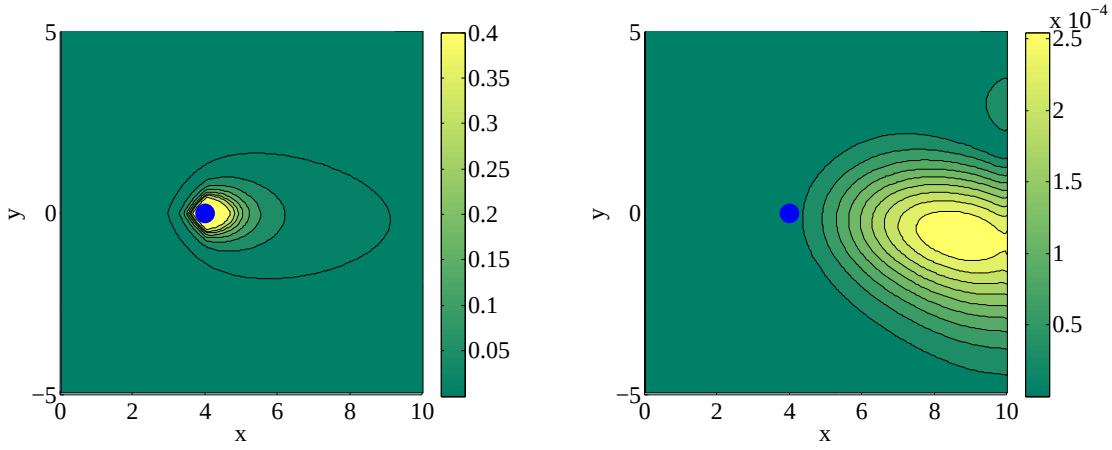


Figure 3.7: Contour slices of the plume of case 5 computed with N = 128 and the source denoted by the blue circle. (a) Horizontal slice of the solution at source height $z = 4$ and (b) vertical slice of the solution at $y = 0$.



(a) (b)

Figure 3.8: Contour slices of the point wise error in ℓ_1 norm for case 5 between $N = 128$ and 256. (a) Shows a horizontal slice of the error at the source height $z = 4$ and (b) shows a horizontal slice at $z = 0$.

3.8.3 Variable Velocity and Diffusivity

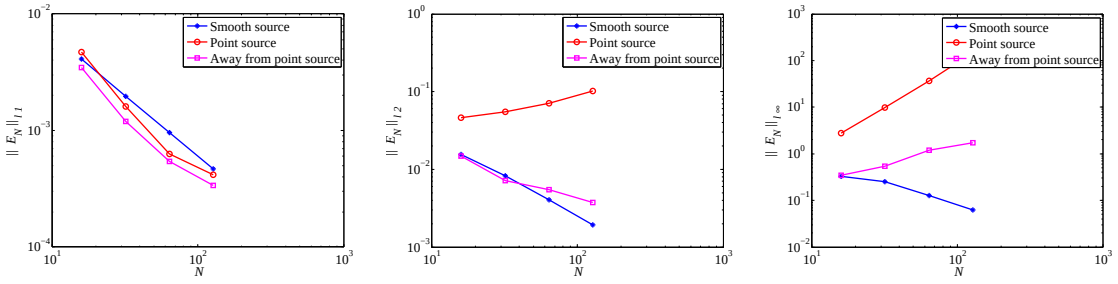
For the next set of convergence tests we consider variable velocity and diffusion coefficients as in Section 2.4. Since the point source had a negative effect on convergence in case 5, we will present results for both smooth and singular sources so as to clearly identify any degradation in convergence.

Case 6: Variable Velocity

The setup of the problem is exactly the same as cases 4 and 5 except that we use the vertical velocity profile from (2.3.1) for the velocity field $\vec{u} = (u(z), 0, 0)$. We choose the reference height $z_r = 10$, the reference velocity $u_r = 1$ and the exponent $m = 0.3$ with a cutoff height $z_{\text{cutoff}} = 2$ (see sections 2.3.1-2.3.2). Results are presented in Table 3.3 and Figure 3.9. Clearly convergence rates are satisfactory for the smooth source, but for the point source we see that convergence rates are similar to those of case 5. This indicates that the convergence rate is degrading due to the singular source and the variable velocity field is affecting convergence.

Case	norm		
	ℓ_1	ℓ_2	ℓ_∞
6 (smooth source)	1.0270	1.0499	1.0494
6 (point source)	0.5932	-0.5208	-2.0229
6 (away from the source)	0.6928	0.5525	-0.5319

Table 3.3: Convergence study for case 6 with smooth and point sources with variable velocity field.



(a) (b) (c)

Figure 3.9: Solution errors for case 6 with variable velocity field for various grid sizes.

Case 7: Variable Diffusivity

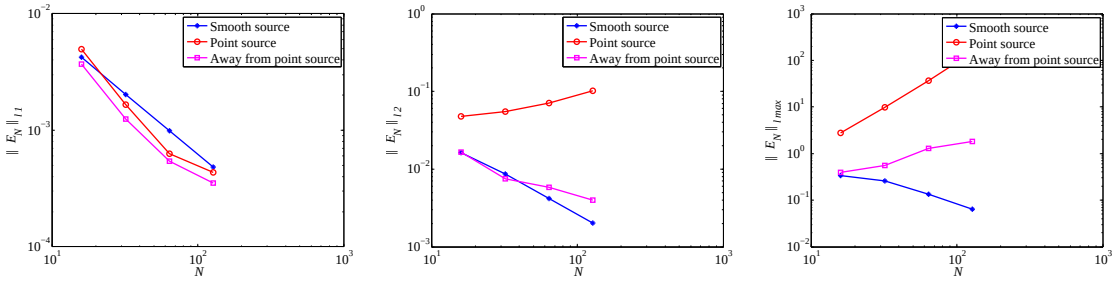
To study the effect of variable diffusion coefficients, we consider both a smooth and a point source as in cases 4–6. We enable variations of the vertical diffusivity $K_z(z)$ and keep $K_x = K_y = 0.25$ constant. We also enable variations in the velocity profile with the same parameters as in case 6. The atmosphere is assumed to be under stable conditions (stability class F) so we use (2.3.3) with the function $\phi(z/L)$ corresponding to classes E or F. Expression (2.3.7) is used with the reference height taken to be $z_r = 10$ and the reference diffusivity is $K_z(z_r) = 0.25$. For the Monin-Obukhov length L we use expression (2.3.6) and parameter values $a = 0.035, b = -0.036$ corresponding to moderately stable conditions. We assume the terrain is uncut grass which gives a roughness length of $z_0 = 0.05$ from Table 2.1 which together with our other assumptions yields $L = 7$. We also keep the diffusion coefficient constant below the height $z_{\text{cutoff}} = 2$ since the current form of our corrections result in a zero diffusivity at ground level which is not desirable specially when we consider ground deposition (see section 2.4.2 for further details on this assumption). Table 3.4 and Figure 3.10 show results of this convergence study. We can see the expected first order convergence rate for the smooth source but convergence degrades for the singular source in a manner similar to cases 4–6; hence, the degradation is due to the singular source and not the variable diffusion coefficient.

	norm		
Case	ℓ_1	ℓ_2	ℓ_∞
7 (smooth source)	1.0268	1.0481	1.0457
7 (point source)	0.5365	-0.5236	-2.0260
7 (away from the source)	0.6227	0.5560	-0.5073

Table 3.4: Convergence study for case 7 with smooth and point sources and corrections to both velocity profile and diffusivity coefficients.

Case 8: Deposition at ground level

For the final numerical convergence study, we consider enabling ground deposition along with all other components of the model including variable velocity and diffusivity. We take $u_{set} = 0$ and consider $W_{dep} = 0.01$ which is taken much larger than the real value in order to



(a) (b) (c)

Figure 3.10: Approximate solution errors for case 7 on various grid sizes with variable velocity field and diffusion coefficients.

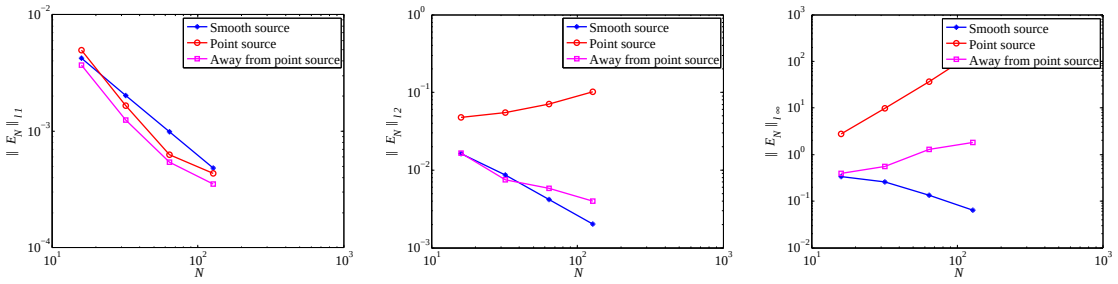
accentuate the effects of ground deposition. All other parameters are the same as in cases 4–7. Table 3.5 presents the results for smooth and point sources. Similar to cases 4–7 we see that first order convergence for the smooth source is achieved but convergence degrades due to the singular source term.

	norm		
Case	ℓ_1	ℓ_2	ℓ_∞
8 (smooth source)	1.0240	1.0479	1.0456
8 (point source)	0.5365	-0.5236	-2.0258
8 (away from the source)	0.6227	0.5560	-0.5073

Table 3.5: Convergence study for case 7 with smooth and point sources with ground deposition.

3.9 Sensitivity to Model Parameters

Since we have so many parameters in the model it is desirable to know the sensitivity of the computed results to each parameter. To do a comprehensive sensitivity analysis is not an easy task and there are many methods we could use. For example, a more sophisticated approach makes use of a package such as DASPK that automatically calculates sensitivity coefficients. Other approaches incorporate solution of the adjoint equations and Lagrange multiplier approaches which reformulate the problem as a constrained optimization problem (see for example the work of Petzold [5]). Here we use a simple "manual" approach and



(a) (b) (c)

Figure 3.11: Plot of errors for case 8 with ground deposition and variable velocity and diffusivity.

solve the problem for various parameter values and look at how much the solution changes.

The problem set up for the sensitivity study is slightly different from that of convergence studies, the reason being that we would like to have a setting that is closer to the real application. We therefore need a test problem that is advection-dominated and has a domain size of the same size as that of the real problem considered in chapter 4. Simulations are performed for $0 \leq t \leq 300$ with $\Delta t = 0.15$ and $N = 100$ mesh points in each direction. The solution domain is the cube $\{0 \leq x, z \leq 300, |y| \leq 150\}$, and we consider a point source of unit strength positioned at $\vec{X}_s = (150, 0, 10)$. Outflow boundary conditions are employed on all faces of the domain for advection, and homogeneous Neumann boundary conditions are used for diffusion except at ground level where a deposition condition with $\bar{W}_{dep} = 0.005$ is used. As for the velocity field we consider a horizontal reference velocity field of magnitude $|(u_r, v_r, 0)| = 10$. We also include the effect of changes in wind direction by taking the wind velocity vector to rotate sinusoidally with period $\omega = \frac{2\pi}{300}$ (that is the field does a full rotation in the computation time). To summarize, the velocity field is given by

$$(u_r, v_r) = 10 \cdot (\cos(\omega t), \sin(\omega t)). \tag{3.9.1}$$

We choose a reference state for the simulations and consider relative deviation of the solution in comparison to this state. Since our convergence studies indicate the solution with the point source has better convergence in the ℓ_1 norm we will utilize this norm to quantify how much the solution varies. Denote by C_{α_i} the numerical solution of the problem for the i^{th} value of parameter α and let C_r be the solution with the reference parameters. Then the deviation is given by

$$\Delta_{\alpha_i}(C_r) = \|C_{\alpha_i} - C_r\|_{\ell_1}. \tag{3.9.2}$$

The absolute deviation of the solution is not very informative so we will normalize (3.9.2) to get a sense of how much the solution departs relative to the reference solution, that is

$$\bar{\Delta}_{\alpha_i}(C_r) = \frac{\|C_{\alpha_i} - C_r\|_{\ell_1}}{\|C_r\|_{\ell_1}}. \tag{3.9.3}$$

As for the choice of the reference parameters, the horizontal diffusion coefficients are constant $K_x = K_y = 0.025$ and the reference value of the vertical diffusivity is that of stability class F over a terrain of roughness $z_0 = 0.05$ which gives $K_z(10) = 0.0391$. For the reference velocity profile parameterization we consider $m = 0.14$. All parameter values for the reference state in the sensitivity studies are summarized in Table 3.6.

Parameter	Symbol	Value
Stability class		F
Reference height	z_r	10 $[m]$
Roughness length	z_0	0.05 $[m]$
Reference diffusion coefficient	$K_z(z_r)$	0.0391 $[m^2\ s^{-1}]$
Vertical diffusion coefficient	K_x, K_y	0.025 $[m^2\ s^{-1}]$
Velocity profile exponent	m	0.14
Cutoff height for velocity and diffusion	z_{cutoff}	2 $[m]$
Deposition coefficient	W_{dep}	0.005 $[m\ s^{-1}]$
Settling velocity	u_{set}	0.0027 $[m\ s^{-1}]$

Table 3.6: Values of the parameters corresponding to the reference state in the sensitivity analysis.

3.9.1 Sensitivity to Stability Classes and Terrain

We start by considering sensitivity to the stability class and terrain type which affect parameters z_0, a and b in expression (2.3.6), and in turn give different values of L and K_z. This case is important since in the real world we are dealing with variable wind over a period of months and during this time the atmosphere will gradually go from one stability class to another and furthermore, the terrain type will also change, although we often have to assume that it is constant over the entire domain.

Table 3.7 shows a grid of values of L for different terrain and stability class. A similar table can be constructed for values of the diffusion coefficient at reference height $K_z(z_r)$ using equations (2.3.3) to (2.3.5) and Table 3.8 shows these values. Note that the vertical diffusion coefficient can vary by nearly two orders of magnitude for different choices of stability conditions. Table 3.9 shows the relative deviation of the solution at termination time; that is, we are comparing the solutions in one instance in time. Table 3.10 shows the relative deviation of the accumulative ground depositions during the whole simulation run. As expected, deviations in the ground deposition values are larger since these values are being integrated over the solution time, therefore the differences accumulate. Our results indicate that an inappropriate choice of stability class could result in relative variations of the solution by up to two orders of magnitude. This is not desirable but still not very surprising because we know the values of $K_z(z_r)$ also vary by a few orders of magnitude.

L	z_0(terrain type)				
Stability Class	10^{-3}	10^{-2}	0.05	0.1	1
D	∞	∞	∞	∞	∞
E	7.7	11	17	22	250
F	3.5	4.9	7	8.5	2.8

Table 3.7: Values of the Monin-Obukhov length L for various stability classes and terrain types.

$K_z(z_r)$	z_0(terrain type)				
Stability Class	10^{-3}	10^{-2}	0.05	0.1	1
D	1.7372	2.3162	3.0198	3.4744	6.9487
E	0.0247	0.0456	0.0811	0.1108	0.5849
F	0.0121	0.0222	0.0391	0.0531	0.2627

Table 3.8: Values of the diffusion coefficient at reference height $K_z(z_r)$ for various stability classes and terrain types.

$\bar{\Delta}_{(\text{stability},z_0)}(C_r)$	z_0(terrain type)				
Stability Class	10^{-3}	10^{-2}	0.05	0.1	1
D	1.1137	1.2069	1.2762	1.3092	1.4315
E	0.0231	0.0124	0.0673	0.1095	0.5383
F	0.0447	0.0273	0	0.0222	0.2626

Table 3.9: Relative deviation of solution at the final time ($t = 30$).

$\bar{\Delta}_{(\text{stability},z_0)}(D_r)$	z_0(terrain type)				
Stability Class	10^{-3}	10^{-2}	0.05	0.1	1
D	71.870	93.539	106.13	111.64	122.34
E	0.5408	0.2223	2.7579	5.2005	43.725
F	0.8526	0.6326	0	0.3240	23.588

Table 3.10: Relative deviation of accumulated ground deposition during simulation time ($0 \leq t \leq 30$).

3.9.2 Sensitivity to Velocity Profile

To study the effects of changing the vertical velocity profiles, we start from the reference setting that was used in the previous sensitivity study and consider five different values of the exponent $m = [0.1, 0.14, 0.2, 0.3]$ in equation (2.3.1). Results are presented in Table 3.10, which tells us the solution can vary up to 35% for various values of the exponent m.

m	0.1	0.14	0.2	0.3
$\bar{\Delta}_m(C_r)$	0.0329	0	0.0488	0.1277
$\bar{\Delta}_m(D_r)$	0.1062	0	0.1440	0.3509

Table 3.11: Relative deviation of the solution at the final time as well as accumulated ground deposition for various values of exponent m.

Chapter 4

Application to an Industrial Setting

Having discussed the details of our numerical algorithm and extensively tested the scheme we now consider a real world application of our solver. This setting is introduced in the paper by Stockie and Lushi [27] where dispersion of zinc from a lead-zinc smelter was studied. The site is located in Trail, British Columbia, Canada and is operated by Teck Resources Limited. In [27] the authors used a Gaussian plume type solution and also solved the inverse source estimation problem where they estimated source emissions based on readings at receptors. Here we wish to compare our results with those of Stockie and Lushi and assess the accuracy of their results against a more complex model.

4.1 Problem Description

An aerial photo of the Trail site is presented in Figure 4.1. The four sources S_s for $s = 1, 2, 3, 4$ as well as the nine receptors R_r for $r = 1, 2, ..., 9$ are marked on the map. The site dimensions are relatively small ($1600 \times 800m$) and therefore it is justifiable to use our short range transport model here. The company has performed a series of measurements of zinc, strontium, sulfur and other species of contaminants over the period of two years 2001-2002 consisting of accumulations of particulates within dustfall jars (receptors). Meteorological data is available for the same period which consists of wind speed and direction measurements averaged over 10 minute intervals. We now list a number of assumptions

regarding our problem, which are imposed by Stockie [27]:

(i) All emission sources are considered to be point sources and are modelled by delta distributions as discussed in chapters 2 and 3. We also assume that all sources have constant emission rates which is a good approximation at least for monthly-averaged data.

(ii) The wind velocity $\vec{u}(t)$ is a function of time only and is uniform over the domain. This is a huge simplification but is justifiable given the relatively small size of the domain and short time intervals.

(iii) We neglect variations in topography therefore the wind is only horizontal. There are mountains around the site but they are far enough away for this assumption to be reasonable.

(iv) Dry deposition is assumed for all simulations given that measurements are performed over months when rainfall is rather small. The main effect of considering wet deposition is in the deposition velocity.

(v) Effects of plume rise are incorporated as effective heights of point sources.

(vi) The atmosphere is of the neutral stability class (class D of the Pasquill-Gifford classification) for all monthly periods. We do not have enough meteorological data to vary the stability class in time and therefore we choose the class D as a compromise that takes into account predominant atmospheric conditions during the months of the simulations.

4.1.1 Parameter Values

Physical values corresponding to Zn as the main contaminant are presented in Table 4.1 and are taken from [27]. The effective height of the sources is $H_s = [15, 35, 15, 15]\, m$ which takes into account the effects of plume rise. Dustfall jars are cylindrical glass containers of diameter 0.162 m and cross-sectional area $A = 0.0206\ m^2$. We assume that the atmosphere is of stability class D and therefore the corresponding value of vertical diffusivity at reference height $z_r = 10\ m$ can be found from equation (2.3.3) for a neutral atmosphere. We assume a roughness parameter of $z_0 = 0.05\ m$ as a compromise for the overall terrain of the site.

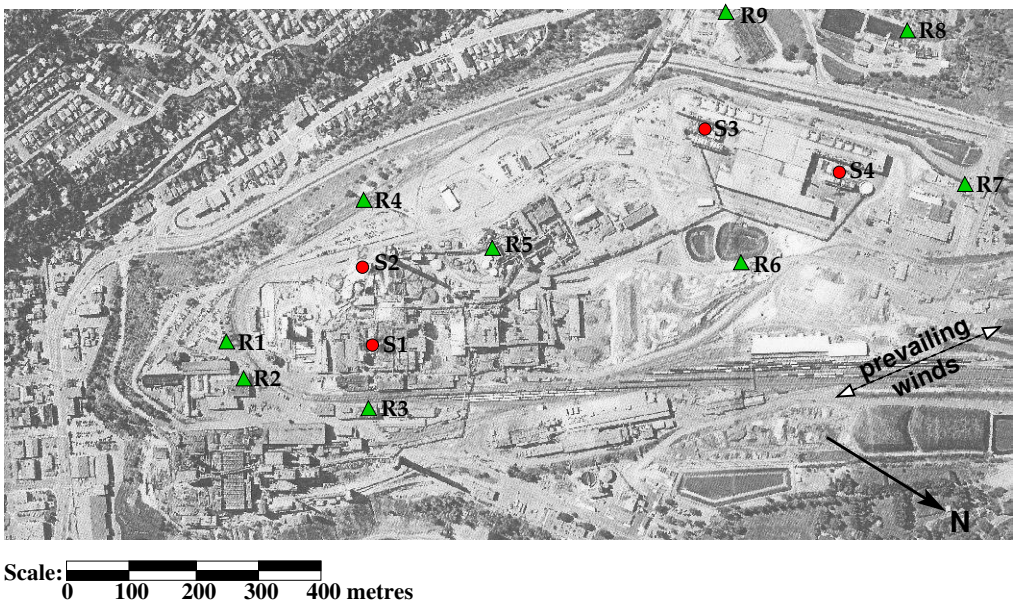


Figure 4.1: An aerial view of the Trail site showing the approximate positions of sources (red circles) and receptors (green triangles). The Direction of prevailing winds and compass north are also indicated in the bottom right corner (source [27]).

Putting these assumptions together with an averaged wind velocity of about 1.6 m/s we get $K_z(z_r) \approx 0.52\ m^2/s$. We will also regularize the wind and velocity profiles and assume that they are constant below a height of 2 m. This is to prevent the wind speed and diffusion coefficients from vanishing near the ground on very fine grids. Table 4.2 summarizes the model parameters just discussed.

Parameter	Symbol	Units	Value for Zn
Density	ρ	$kg\, m^{-3}$	3540
Molar mass	M	$kg\, mol^{-3}$	0.161
Deposition velocity	W_{dep}	$m\, s^{-1}$	0.005
Settling velocity	u_{set}	$m\, s^{-1}$	0.0027

Table 4.1: Values of physical parameters for zinc, which actually appears in the form of a sulfate $ZnSO_4$ (source [27]).

Parameter	Symbol	Value
Stability class		D
Reference height	z_r	$10\,[m]$
Roughness length	z_0	$0.05\,[m]$
Reference diffusion coefficient	$K_z(z_r)$	$0.52\,[m^2\,s^{-1}]$
Velocity profile exponent	m	0.14
Cutoff height for velocity and diffusion	z_{cutoff}	$2\,[m]$

Table 4.2: Values of model parameters based on average wind velocity and recommended stability class.

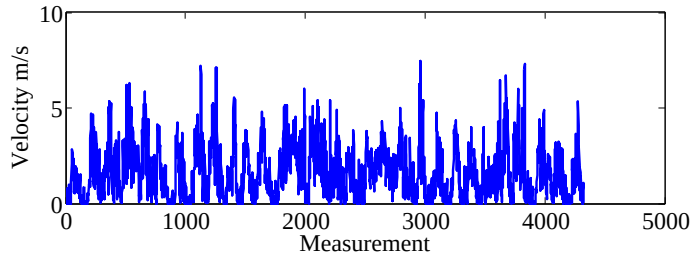
4.1.2 Wind Data

The wind measurements, which are provided in 10 minute intervals for a monthly period, are perhaps the most important input for our simulations. We are working with raw measurements and therefore we need to clean up the data before they can be used for the simulations.

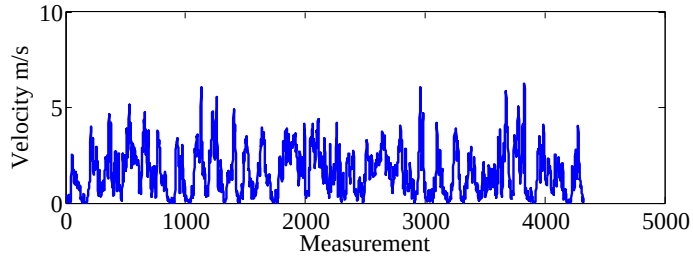
First, for calm wind conditions where $u < 0.1\,m/s$, we set the velocity to $u_{\min} = 0.1\,m/s$ since the anemometers used for the measurements are not accurate for small wind velocities. We also compute the wind direction at these data points by interpolation of wind direction at the closest valid points having a non-zero velocity. Next, we apply a Laplacian smoothing operator to both the wind direction and velocity measurements in order to reduce the noise. The effect of this process for 5 iterations is visible in Figure 4.2 which indicates some reduction in noise as well as the maximum wind velocity. This smoothing process is attractive for our numerical simulations since the smoothness of velocity field is a key assumption for our discretization.

The histogram and windrose diagram of the wind data are pictured in Figure 4.3. Looking at the data we realize that the wind velocity is below 4 m/s for most of the time; indeed only 4.8% of the measurements are above this value. Therefore, it is crucial for our simulations to use adaptive time stepping, otherwise the code performance would degrade dramatically. This also affects the accuracy of our scheme, since a small time step arising from large wind velocity results in a small Courant number, and therefore the advection solver introduces unnecessary smearing of the solution at times when the velocity is much

smaller.



(a)

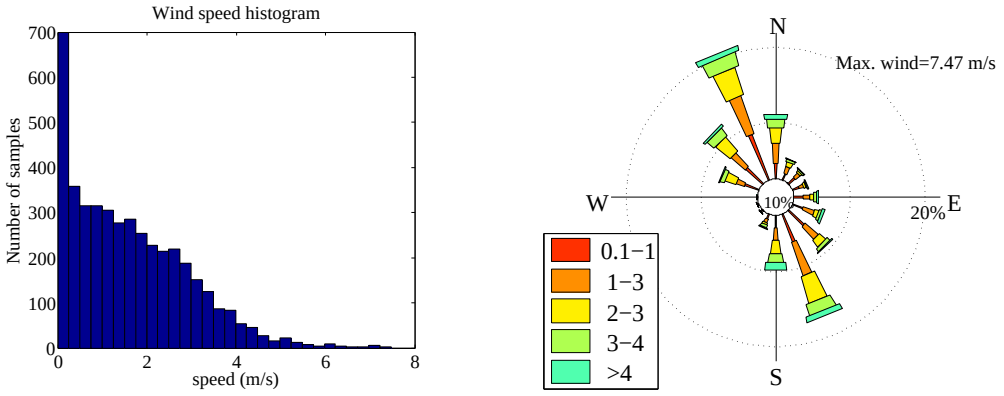


(b)

Figure 4.2: (a) Raw wind velocity measurements for June 3-July 2, 2002 and (b) smoothed velocity after five iterations of the averaging operator.

4.2 Results

We consider a computational domain $\Omega = \{-2000 \leq x \leq 4000, -2000 \leq x \leq 4000, 0 \leq z \leq 500\}$, that is large enough so that boundary conditions have little effect on the solution. A grid of size $100 \times 100 \times 100$ is used which means that the horizontal grid cells are $60 \times 60\, m^2$ and the vertical resolution is $5\, m$. For this set of results we use the engineering estimates of the source emission rates, $Q_s = [35, 80, 5, 5] \times 10^3\ kg/yr$ for zinc identified in [27]. The x-coordinates of the sources are $x_s = [748, 625.6, 255, 251.6]\, m$ and the y-coordinates are $y_s = [224.4, 176.6, 646, 867]\, m$. As mentioned in Section 4.1.2 we will employ adaptive time stepping so that we maintain a Courant number that is close to 0.9 in order to minimize the number of time steps required for the simulation and improve the accuracy of the advection solver; this means that the diffusion matrix must be reassembled at each time step. We perform our simulations for the period of June 3-July 2, 2002, which is a relatively long simulation time (approximately $2.9 \times 10^7\ s$ and requires on the order of 1 to 2 days



(a) (b)

Figure 4.3: (a) Histogram of measured wind data for the period of June 3-July 2, 2002 with the mean value of wind velocity being $1.62\,m/s$. (b) Wind rose diagram of wind data with the portion of calm winds identified by the inner circle. Note the dominant direction of the wind is northwest and southeast.

of clock time for each simulation at this resolution). We do not allow the solver to take time steps that are larger than $\Delta t_{max} = 300\,s$ so that no wind measurements are skipped. Simulated deposition of pollutants using both our method and the Gaussian plume model used in [27] at the 9 receptors (pictured in Figure 4.2) are presented in Table 4.3 alongside actual measurements of the depositions. We can see that both our model and the Gaussian plume model over estimate the total deposition of the pollutants in all of the receptors. Comparing the depositions at single receptors we realize that our results are not close to the actual measurements. For example our model predicts maximum deposition in R1 and R2 but actual deposition in R1 is a lot smaller than R2. Overall our model predicts less variation between deposition in receptors that are positioned close to each other which is due to the more amplified diffusive behavior of our model.

4.2.1 Comparison to Gaussian Plume Solution

Here we compare the results of our simulation to those of Stockie [27] where the Ermak Gaussian plume solution has been utilized for estimation of ground depositions. Gaussian plume solutions are widely utilized and are known to give acceptable results but they require certain assumptions that do not always hold in our simulations. Some of these assumptions

	Deposition (mg)									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	Total
Advection-diffusion	40.3	44.5	17.5	14.9	22.1	14.2	12.0	12.0	14.7	187.2
Gaussian plume [27]	28.2	80.5	16.6	9.2	25.1	12.9	4.3	5.0	9.5	191.7
Actual measurements	8.4	68.0	33.0	4.2	11.0	8.2	2.9	2.2	0.9	138.8

Table 4.3: Simulated and actual measurements of Zn at the 9 receptors for the period of June 3-July 2, 2002 using the engineering estimates for source emission rates.

were discussed in Section 2.2.

From Table 4.2 we realize that the reference diffusion coefficient $K_z(z_r)$ is not small compared to the wind velocities. Approximately 40% of the measurements have a wind speed below 1 m/s, and for these time intervals the diffusive behavior of the problem can be comparable to the advection. In such circumstances, one of the main assumptions of the Gaussian plume solution no longer holds namely that the problem is advection dominated.

The Ermak solution that was used by Lushi and Stockie in [27] assumes that the solution reaches steady state in each wind measurement interval (of length 600 s) and therefore does not consider the transient behavior of the problem. Figure 4.4 shows an instant in our simulations where the change in direction of the wind has resulted in mixing of the plume, hence demonstrating that this assumption does not necessarily hold.

Looking at Table 4.3 we realize that the Gaussian plume solution predicts lower deposition values compared to our model, and that our simulations give nearly twice the total deposition. Figures 4.5 and 4.6 show contour plots of deposition over the Trail site as computed by our model and the Gaussian plume model successively. We can see that the results are qualitatively similar in the sense that both predictions have a higher deposition in the direction of the prevailing winds (southeast and northwest). Apart from predicting less deposition, the Gaussian plume simulations give less deposition away from the site, especially in the southwest direction where the receptor R4 is located. We can also see that contours of the Gaussian plume solution have more spikes while our model results in smoother contours. This is partly due to the fact that our model captures the transient behavior of the plume as pictured in Figure 4.4 and also a consequence of choosing a constant vertical diffusion coefficient for the entire simulation. The main difference between the two results occurs at the source locations where our simulations predict maximum deposition while the Gaussian

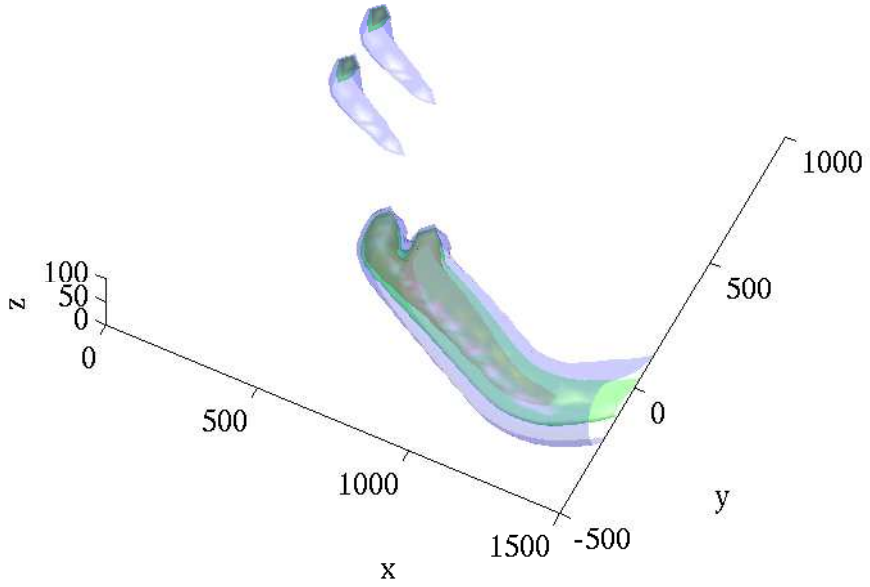


Figure 4.4: An isosurface plot of the solution at $t = 2137905s$ showing the transient behavior of the solution. Red, green and blue isosurfaces correspond to concentrations of 1.7, 1.6 and 1.5 mg/m^3 respectively.

plume solution predicts maximum deposition away from the sources.

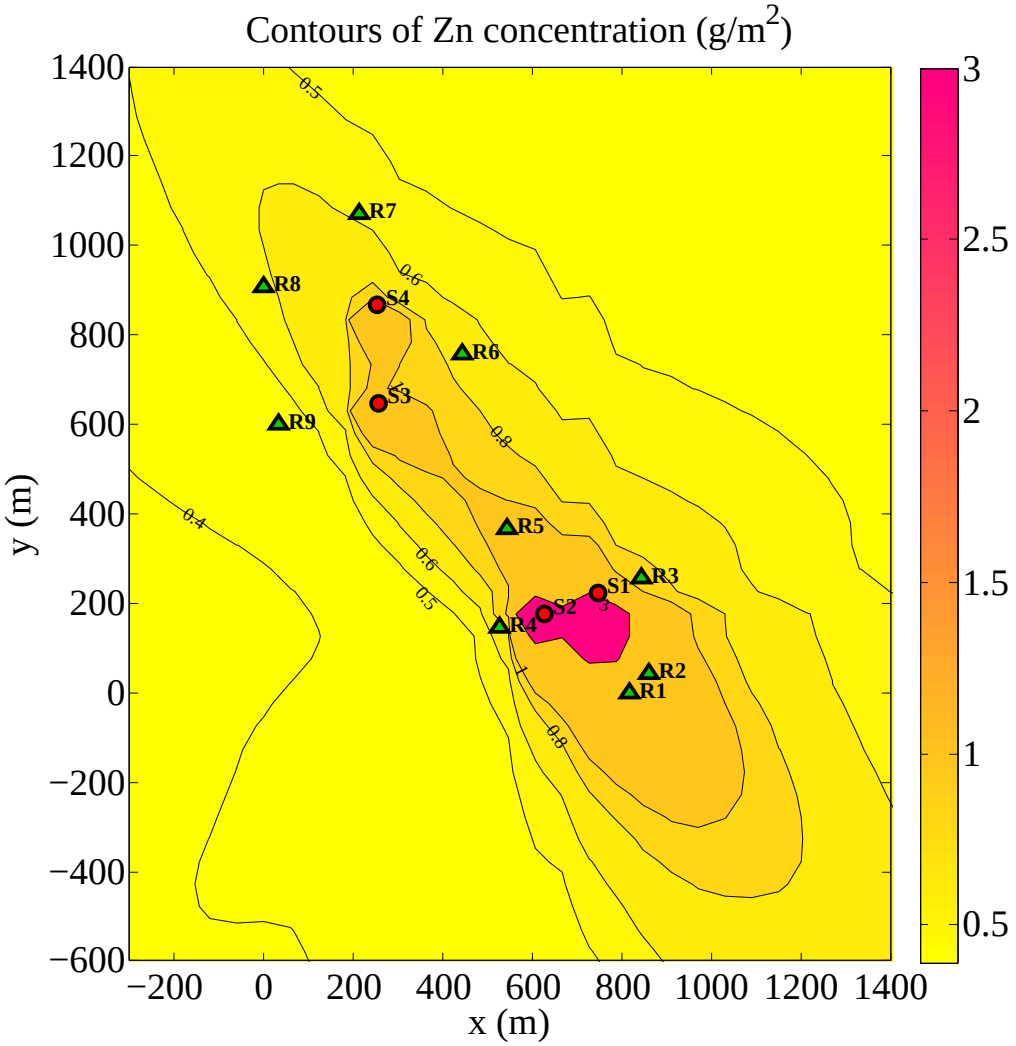


Figure 4.5: Contour plot of deposition as simulated by our advection-diffusion solver. The model predicts a maximum deposition at the source locations.

4.2.2 Comparison to Estimated Emission Rates

In this section we perform another set of simulations using the source emission rates that were estimated by solving the inverse problem in [27]. The predicted emission rates are

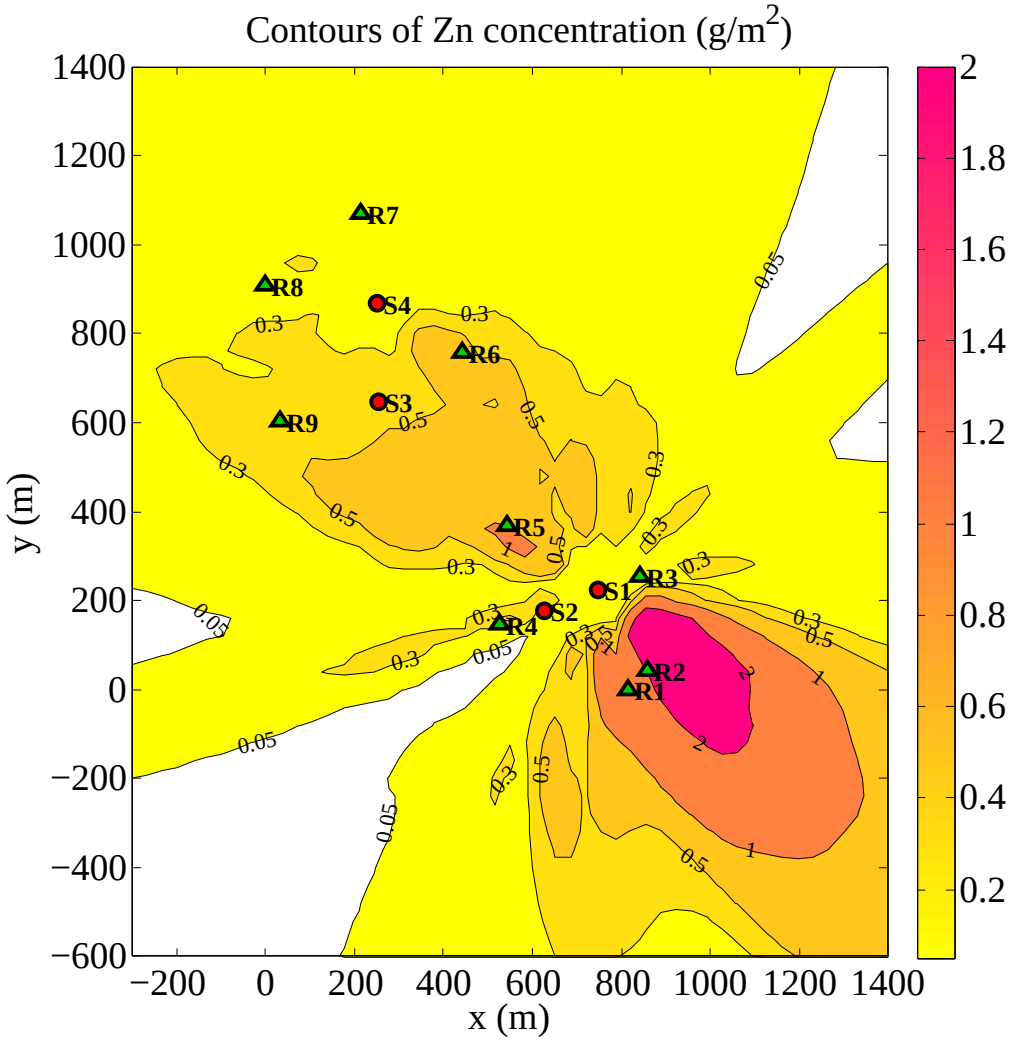


Figure 4.6: Contour plot of deposition simulated by the Gaussian plume model of [27]. Note that apart from giving less deposition overall, this model predicts that maximum deposition occurs away from the sources.

$Q = [17, 60, 5, 5] \times 10^3\, kg/yr$ of zinc while all other parameter values are the same as in our previous simulation. Table 4.4 shows simulated deposition values of this case and a contour plot of this solution is presented in Figure 4.7 where, as expected, deposition levels are lower over the whole domain. We can see that the results are closer to actual measurements but our model still over-estimates the measurements. In other words, our model with the given parameters needs smaller source emission rates to give results that are consistent with the measurements. Also note that while the total deposition in all receptors is closer to actual measurements, the difference at separate receptors is still quite large. For example, we are severely over-estimating the deposition in receptors R6-R9 while our predictions are close to measurements for R2 and R3. Note that actual measurements at the receptors are very noisy due to large measurement errors and therefore the total deposition in all of the receptors is in fact a better measure of the results.

	Deposition (mg)									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	Total
Advection-diffusion	27.4	26.9	17.2	15.2	16.6	14.3	12.2	12.3	10.2	152.7
Actual measurements	8.4	68.0	33.0	4.2	11.0	8.2	2.9	2.2	0.9	138.8

Table 4.4: Simulated and actual measurements of Zn at the 9 receptors for the period of June 3-July 2, 2002 using source emission rates estimated by the inverse problem in [27].

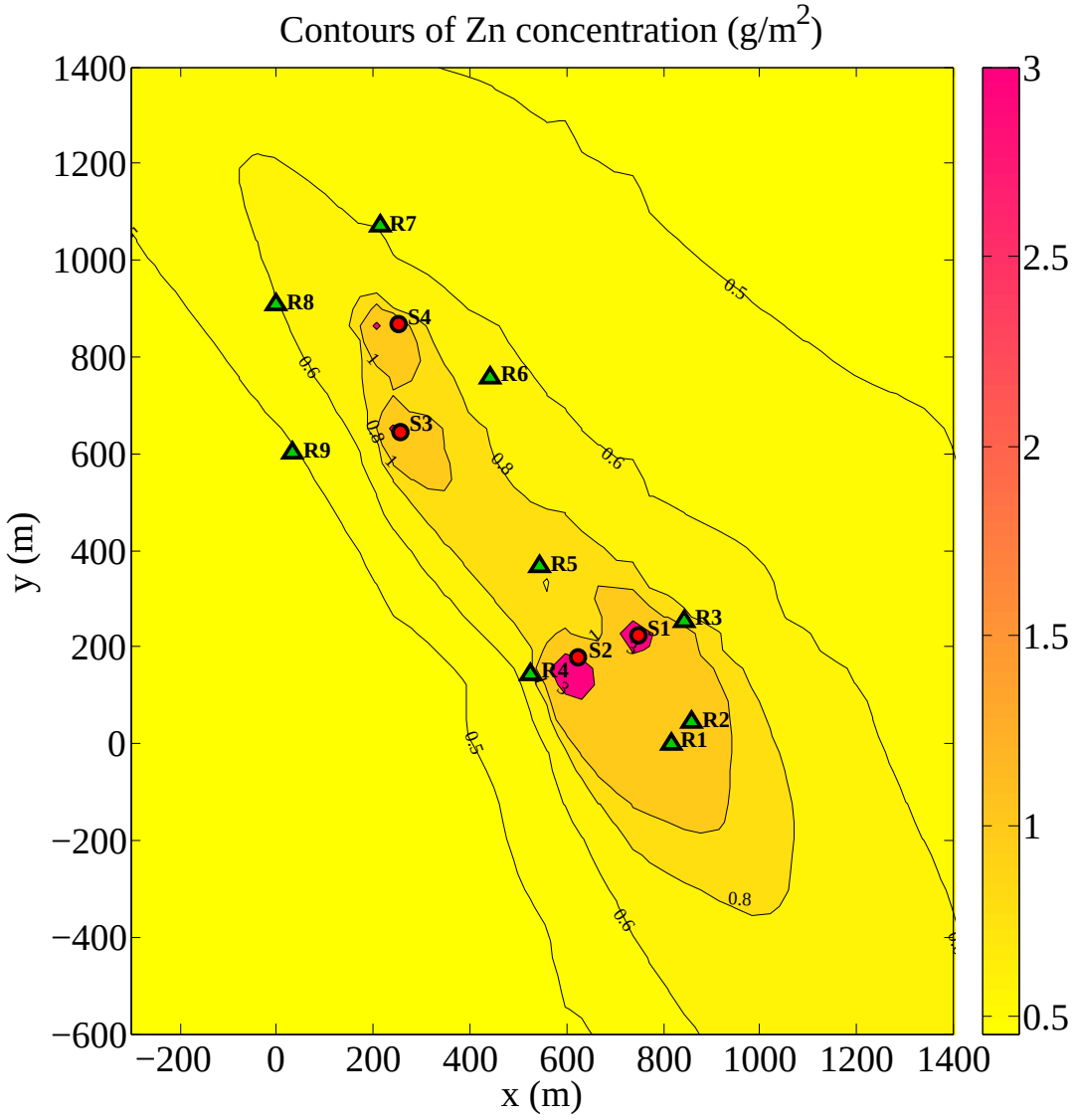


Figure 4.7: Contour plot of deposition values for source emission rates estimated by solution of the inverse problem in [27].

Chapter 5

Adjoint Equations

In this chapter we present the basis for a promising scheme for solution of the inverse source estimation problem which is the problem of finding the source emission rates given monthly measurements of the depositions. This is not intimately related to the main topic of this thesis and can be viewed as an idea for future work on this subject.

5.1 Introduction

The problem corresponding to estimating source emissions from ground deposition or point measurements of pollutant concentration falls under the category of source inversion problems. This is a well studied area of research with extensive applications in tomography, gravimetry and electrostatics. The reader is referred to the monograph by Isakov [19] for a comprehensive study of the theory of source inversion problems. This problem is ill-posed in the sense of Hadamard and often lacks uniqueness and stability. We assume that point sources are represented as a finite number of delta distributions at known positions, which reduces the ill-posedness of the problem since it imposes restrictions on the form of the source term. Loosely speaking, the set of point sources over the domain of our problem is a subset of all the possible sources and this narrows down our search for a possible solution. With this knowledge we present a method that obtains the source emission rates via solution of the adjoint equation. It is noteworthy that the source inversion problem with these assumptions will have a unique solution but as we will see later on, our approach can be generalized for solution of the problem when less information is available.

Adjoint operators are a well known concept in functional analysis, theory of PDEs

and inverse problems (see [23, 19]). Adjoint equations are extensively used in the control and shape optimization literature for sensitivity analysis and computing the derivatives of objective functionals with respect to a large number of parameters [5, 36]. A comprehensive study of adjoint equations, their attributes, definitions and applications is presented in the monographs by Marchuk et al. [29, 30].

We start with a brief definition. Consider two elements $x, y \in H$ where H is a real Hilbert space. Now let $A : H \to H$ be an operator acting on x. This is often a differential operator that defines our PDE

$$Ax = f. \tag{5.1.1}$$

Now the *Hilbert adjoint* of the operator A, denoted by A^*, is defined through the relationship

$$\langle Ax, y\rangle = \langle x, A^* y\rangle. \tag{5.1.2}$$

where $A^* : H^* \to H*$. For simplicity, we assume that A^* maps elements of H back to itself, i.e. x and y belong to the same space. Then an equation of the form

$$A^* y = g, \tag{5.1.3}$$

with proper choice of g is called the equation adjoint to (5.1.1). Proceeding informally, if we take equations (5.1.1)–(5.1.3) together then we obtain

$$\langle f, y\rangle = \langle x, g\rangle. \tag{5.1.4}$$

This equation allows us to evaluate a functional on the solution of (5.1.1) by evaluating a functional on the solution of the adjoint equation. In the case of our problem, the functional on x (the solution of the original equation) is a finite number of point measures (that is, ground depositions or concentration measurements) and the source term f is the object we would like to find which also has the form of a finite number of point measures. The perfect setting for application of the adjoint equations is when the operator A is linear and hence the adjoint equation is independent of the solution x. In this case, a single solution of the adjoint equation (5.1.3) can be used to evaluate the functional on the right hand side of (5.1.4) for a large number of source terms. This feature is very promising in areas such as Bayesian inversion where one might need to sample the forward problem a large number of times.

With this as motivation, we proceed to derive the adjoint equation for the pollutant dispersion problem and consider how this idea can help with the solution of the source inversion problem.

5.2 Adjoint Formulation at Steady State

Let us start by considering the steady state advection-diffusion equation in 2D, that is

$$\vec{u} \cdot \nabla c - K\Delta c = f, \qquad c(\vec{X}) \to 0 \quad as \quad \left|\vec{X}\right| \to 0, \tag{5.2.1}$$

where the velocity field and diffusion coefficients are assumed to be constant and independent of space. We make use of the L^2 inner product to derive the adjoint equation. Multiply both sides of (5.2.1) by a function c^* and integrate by parts to transfer all derivatives onto c^*,

$$\langle c, -\vec{u} \cdot \nabla c^* - K\Delta c^* = g \rangle = \langle c, g \rangle, \tag{5.2.2}$$

where we have also assumed that $c^*(\vec{X}) \to 0$ as $\left|\vec{X}\right| \to \infty$ to eliminate the boundary terms. This tells us that the adjoint equation is

$$-\vec{u} \cdot \nabla c^* - K\Delta c^* = g, \qquad c^*(\vec{X}) \to 0 \quad as \quad \left|\vec{X}\right| \to 0, \tag{5.2.3}$$

which is the same advection-diffusion equation, except that the velocity field points in the opposite direction. Note that the adjoint equation is independent of the source term that appears in the original problem. The choice of g can be determined by considering the right hand side of (5.2.2) as a measure that is imposed on the solution of the original problem.

We demonstrate with a simple example: consider a single point measurement of the emissions from a point source at $\vec{X}_s$ at a receptor positioned at $\vec{X}_r$. Then, the source terms for the original and adjoint equations are given by

$$\begin{aligned} f &= Q\delta(\vec{X} - \vec{X}_s), \\ g &= \delta(\vec{X} - \vec{X}_r), \end{aligned} \tag{5.2.4}$$

where Q is the emission rate. Substituting into (5.1.4), with c and c^* as the main and adjoint variables, we obtain

$$Qc^*(\vec{X}_s) = c(\vec{X}_r), \tag{5.2.5}$$

which allows us to solve for Q given the solution of the adjoint equation and the measurement $c(\vec{X}_r)$ at the receptor.

This framework is ideal for point sources but for more general integrable source terms, equation (5.2.5) results in a moment condition

$$\int f(\vec{X})c^*(\vec{X})d\vec{X} = c(\vec{X}_r). \tag{5.2.6}$$

It is easy to see the lack of uniqueness in this setting. In general, for a finite number of measurements $M_i = c(\vec{X}_{r_i})$ at $i = 1, 2, ...m$ receptors we can consider m adjoint equations for adjoint variables c_i^* for each measurement and get the finite dimensional moment problem

$$\int f(\vec{X}) c_i^*(\vec{X}) d\vec{X} = M_i. \tag{5.2.7}$$

Going back to our discussions in section 3.4.3, this problem is solvable if f and c_i^* are elements of a separable space and c_i^* are linearly independent. This setting is attractive for solving many source inversion problems since we only need to compute the solution of the adjoint equations once, and in some cases we can even solve the equations analytically.

5.3 Adjoint Formulation in Transient Case

Here we consider the adjoint formulation for time dependent problems. The theory for these operators is, in essence, the same as what we discussed in the previous section, but in this case the adjoint formulation gives information regarding the initial condition as well as the source terms involved in the problem. Let us consider a PDE

$$c_t + Ac = f, \qquad c(\vec{X}, 0) = h(\vec{X}), \tag{5.3.1}$$

where A is some differential operator such as the advection-diffusion operator and the solution $c(\vec{X}, t)$ exists for the domain $\{0 < t \leq T,\ \vec{X} \in \mathbb{R}^N\}$. We also prescribe the far field boundary conditions $u \to 0$ as $\left|\vec{X}\right| \to \infty$ so that boundary terms vanish.

Multiplying both sides of (5.3.1) by the adjoint variable $c^*(\vec{X}, t)$ and integrating by parts over both time and space we get the adjoint equation

$$-c_t^* + A^* c^* = g, \qquad c^*(\vec{X}, T) = l(\vec{X}), \tag{5.3.2}$$

where the initial conditions and source terms are linked through the relationship

$$\begin{aligned} \int h(\vec{X}) c^*(\vec{X}, 0) dx + \int_0^T \int f(\vec{X}, \tau) c^*(\vec{X}, \tau) dx\, d\tau \\ = \int c(\vec{X}, T) l(\vec{X}) dx + \int_0^T \int c(\vec{X}, \tau) g(\vec{X}, \tau) dx\, d\tau. \end{aligned} \tag{5.3.3}$$

Here we have assumed that all functions are sufficiently regular for us to be able to perform integration by parts and possibly switch order of integration. Note that the adjoint equation is solved backwards in time and so based on equation (5.3.3) we can think of the initial

condition of the adjoint equation $l(t)$ as a measure on the solution of the original problem at the final time. The second term on the right hand side of (5.3.3) is also a measure on the solution both in time and space, so we can think of this quantity as a measurement of accumulated values of the solution. From this perspective, the right hand side of (5.3.3) is the set of measurements from the forward problem or our data set.

In the case of the pollutant source inverse problem mentioned in this thesis, the accumulated ground deposition measurement $d_r(t)$ at a receptor at $\vec{X}_r$ can be written as

$$d_r(t) = \int_0^t \int c(\vec{X}, \tau)\delta(\vec{X} - \vec{X}_r)dx\, d\tau, \tag{5.3.4}$$

and the corresponding adjoint equation for this problem would be

$$-c_t^* + A^*c^* = \delta(\vec{X} - \vec{X}_r), \qquad c^*(\vec{X}, T) = 0, \tag{5.3.5}$$

where we assume that $l(\vec{X}) = 0$, since we are only measuring the accumulated deposition of pollutants and not their point values. Note that the adjoint equation is again independent of the solution of the original problem and therefore this framework can be used to sample the forward problem efficiently.

Chapter 6

Summary and Future Work

6.1 Summary

In this thesis we presented a numerical algorithm for solution of the advection-diffusion equation in 3D. We used a first order operator splitting to solve the advection and diffusion terms separately and also used dimensional splitting to decouple the spatial derivatives in each term. Therefore, we ended up with a series of 1D tridiagonal linear solves in each time step. This allowed us to perform sophisticated simulations in a reasonable time and with acceptable resolution. We also set up our algorithm to be able to handle variable advection and diffusion coefficients as well as various boundary conditions. Our code uses the CLAWPACK software package for the advection solves, and is coupled to our own diffusion solver that which utilizes LINPACK routines for better performance.

We were particularly interested in using our numerical algorithm to model dispersion of pollutants in the atmosphere. We considered the problem of contaminant emission from point sources and their deposition onto the ground. Some models for altitude-dependent wind speed and diffusion coefficients were discussed and incorporated into our numerical scheme.

One of the main goals of this project was to perform simulations of dispersion of zinc on an actual industrial site in Trail, British Columbia, Canada. Actual measurements of wind speed and monthly accumulative pollutant depositions in 9 dustfall jars were available from the company along with engineering estimates of source emission rates. We used this data to perform simulations and compared our results to those of a Gaussian plume model developed by Lushi and Stockie [27]. We also used our solver to verify the results of the

inverse emission estimation problem that was solved in [27].

Finally, we discussed the basis of a promising new approach for solution of the inverse source estimation problem using the solution of adjoint equations. We presented a short discussion on how this new approach will allow us to estimate source emission rates from ground deposition measurements very efficiently.

6.2 Future Work

As we mentioned earlier in this thesis, there is great uncertainty underlying the parameters used in our model. For example, the stability class and diffusion coefficients all vary based on the meteorological data. Our sensitivity analysis in Chapter 3 raised some concerns regarding how trustworthy our results may be if we are not sure what the stability class is. Better estimation of the model parameters and perhaps enabling the model to automatically change parameters throughout the simulations is a possible subject for future work. Such a study can be accompanied by a rigorous sensitivity analysis to identify which parameters are most important.

Even though our method is efficient enough to perform simulations at an acceptable spatial resolution, there is still room for improvement in terms of efficiency and performance. For example, to construct and factor the diffusion matrices in 3D at each time step is very time consuming. A possible remedy for this issue is to use an iterative solver instead of the current tridiagonal solves. Another possible approach is to use parallel implementations of the solvers since our current code is only serial.

Our convergence results in Chapter 3 indicated that our current implementation of the delta distribution is the main cause of reduction in convergence rate. This raises the question of whether we can use a different approximation to the delta distribution in order to achieve better accuracy.

The inverse source estimation problem is an extremely challenging and interesting problem. The cases of unknown number of sources or unknown source locations are both extremely ill-posed and difficult to solve. We are therefore interested in pursuing this path of research in the future and applying the idea of adjoint equations to construct a scheme for solution of these problems.

The problem of identifying the position of the sources is the perfect setting for application of Bayesian inversion algorithms. In such problems we often want to obtain the *posterior*

probability density of the unknown x, $\pi_{\text{post}}(x)$, given a set of observations y and a *prior probability density*, $\pi_{\text{pr}}(x)$. Here x can be any unknown parameter in our model such as source emission rates or number of sources. By Theorem 3.1 from [22] (Bayes formula) we have

$$\pi_{\text{post}}(x) = \frac{\pi_{\text{pr}}(x)\pi(y|x)}{\pi(y)}, \tag{6.2.1}$$

where $\pi(y|x)$ is the probability of observing y given the input x and $\pi(y)$ is the probability of observing y. The prior probability distribution on the unknown is an assumption that we impose on the model and we often take the observations to have a certain probability density therefore the only unknown in (6.2.1) is $\pi(y|x)$. A common technique for approximating this probability distribution is a *Markov Chain Monte Carlo* (MCMC) algorithm which estimates $\pi(y|x)$ via sampling the probability space and in turn requires a solution of the forward problem. By Proposition 3.11 from [22], accuracy of the MCMC algorithm improves as we increase the number of samples. Solution of the adjoint equations is a promising approach to be incorporated in this framework. We expect to achieve major performance improvements since, for linear operators, the solution to the adjoint equation is independent of the source term in the original problem–this allows us to sample the forward problem more efficiently.

Bibliography

[1] S. Pal Arya. *Air pollution meteorology and dispersion*. Oxford University Press, New York, 1999.

[2] Uri M. Ascher, Steven J. Ruuth, and Raymond J. Spiteri. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. *Applied Numerical Mathematics*, 25(2):151–167, 1997.

[3] S. Baroutian, A. Mohebbi, and A. Soltani Goharrizi. Measuring and modeling particulate dispersion: A case study of Kerman cement plant. *Journal of Hazardous Materials*, 136(3):468–474, 2006.

[4] Brent M. Bowen. Long-term tracer study at Los Alamos, New Mexico. Part II: Evaluation and comparison of several methods to determine dispersion coefficients. *Journal of Applied Meteorology*, 33(11):1236–1254, 1994.

[5] Yang Cao, Shengtai Li, Linda Petzold, and Radu Serban. Adjoint sensitivity analysis for differential-algebraic equations: The adjoint DAE system and its numerical solution. *SIAM Journal on Scientific Computing*, 24(3):1076–1089, 2003.

[6] M.D. Carrascal, M. Puigcerver, and P. Puig. Sensitivity of Gaussian plume model to dispersion specifications. *Theoretical and Applied Climatology*, 48(2-3):147–157, 1993.

[7] John P. Chastain and Francis J. Wolak. Application of a Gaussian plume model of odor dispersion to select a site for livestock facilities. *Proceedings of the Water Environment Federation*, 2000(3):745–758, 2000.

[8] T.J. Chung. *Computational Fluid Dynamics*. Cambridge University Press, 2010.

[9] I. Dimov, K. Georgiev, Tz. Ostromsky, and Z. Zlatev. Computational challenges in the numerical treatment of large air pollution models. *Ecological Modelling*, 179(2):187–203, 2004.

[10] S. Emeis, R. Forkel, W. Junkermann, K. Schäfer, H. Flentje, S. Gilge, W. Fricke, M. Wiegner, V. Freudenthaler, S. Groß, et al. Measurement and simulation of the 16/17 April 2010 Eyjafjallajökull volcanic ash layer dispersion in the northern alpine region. *Atmospheric Chemistry and Physics*, 11(6):2689–2701, 2011.

[11] Donald L. Ermak. An analytical model for air pollutant transport and deposition from a point source. *Atmospheric Environment*, 11(3):231–237, 1977.

[12] Lawrence C. Evans. *Partial Differential Equations*. American Mathematical Society, Providence, 1998.

[13] B.E.A. Fisher and J.F. Macqueen. A theoretical model for particulate transport from an elevated source in the atmosphere. *IMA Journal of Applied Mathematics*, 27(3):359–371, 1981.

[14] C. Geels, P. Løfstrøm, L.M. Frohn, J. Brandt, and G. Kjellsson. Wind dispersal of genetically modified pollen from oilseed rape and rye fields. *DARCOFenews*, (2), 2004.

[15] Donald Golder. Relations among stability parameters in the surface layer. *Boundary-Layer Meteorology*, 3(1):47–58, 1972.

[16] D.M. Hamby. The Gaussian atmospheric transport model and its sensitivity to the joint frequency distribution and parametric variability. *Health Physics*, 82(1):64–73, 2002.

[17] Charles Hirsch. *Numerical Computation of Internal and External Flows: The Fundamentals of Computational Fluid Dynamics*, volume 1. Butterworth-Heinemann, 2007.

[18] Willem Hundsdorfer and Jan G. Verwer. *Numerical Solution of Time Dependent Advection-Diffusion-Reaction Equations*. Springer, 2007.

[19] Victor Isakov. *Inverse Source Problems*. American Mathematical Society, Providence, 1990.

[20] Hyo-Joon Jeong, Eun-Han Kim, Kyung-Suk Suh, Won-Tae Hwang, Moon-Hee Han, and Hong-Keun Lee. Determination of the source rate released into the environment from a nuclear power plant. *Radiation Protection Dosimetry*, 113(3):308–313, 2005.

[21] Sergey I. Kabanikhin. *Inverse and Ill-posed Problems*. Siberian Scientific Publishers, Novosibirsk, 2009.

[22] Jari P. Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*. Springer, 2005.

[23] Erwin Kreyszig. *Introductory Functional Analysis with Applications*, volume 1. Wiley, New York, 1989.

[24] R.L. Lee, J.R. Albritton, D.L. Ermak, and J. Kim. Computational Fluid Dynamics modeling for emergency preparedness and response. *Environmental Modelling & Software*, 12(1):43–50, 1997.

[25] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.

[26] Randall J. LeVeque. *CLAWPACK Version 4.3 User's Guide*. University of Washington, 2006.

[27] E. Lushi and J. M. Stockie. An inverse Gaussian plume approach for estimating atmospheric pollutant emissions from multiple point sources. *Atmospheric Environment*, 44:1097–1107, March 2010.

[28] Guri I. Marchuk. Splitting and alternating direction methods. *Handbook of Numerical Analysis*, 1:197–462, 1990.

[29] Guri I. Marchuk. *Adjoint Equations and Analysis of Complex Systems*. Kluwer, 1995.

[30] Guri I. Marchuk, Valeri I. Agoshkov, and V.P. Shutiaev. *Adjoint Equations and Perturbation Algorithms in Nonlinear Problems*. CRC Press, 1996.

[31] Gregory J. McRae, William R. Goodin, and John H. Seinfeld. Development of a second-generation mathematical model for urban air pollution-I. Model formulation. *Atmospheric Environment*, 16(4):679–696, 1982.

[32] A.S. Monin and A.M. Obukhov. Basic laws of turbulent mixing in the surface layer of the atmosphere. *Contributions of the Geophysical Institute of Slovakian Academy of Sciences, USSR*, 151:163–187, 1954.

[33] S.M.S. Nagendra and Mukesh Khare. Line source emission modelling. *Atmospheric Environment*, 36(13):2083–2098, 2002.

[34] Stelios Pashardes and Constantinos Christofides. Statistical analysis of wind speed and direction in Cyprus. *Solar Energy*, 55(5):405–414, 1995.

[35] I.A. Perez, M.A. Garcia, M.L. Sanchez, and B. De Torre. Analysis and parameterisation of wind profiles in the low atmosphere. *Solar Energy*, 78(6):809–821, 2005.

[36] J. Reuther, J.J. Alonso, M.J. Rimlinger, and A. Jameson. Aerodynamic shape optimization of supersonic aircraft configurations via an adjoint formulation on distributed memory parallel computers. *Computers and Fluids*, 28(45):675 – 700, 1999.

[37] O.F.T. Roberts. The theoretical scattering of smoke in a turbulent atmosphere. *Proceedings of the Royal Society of London. Series A*, 104(728):640–654, 1923.

[38] Walter Rudin. *Functional Analysis. International series in pure and applied mathematics*. McGraw-Hill, New York, 1991.

[39] John H. Seinfeld and Spyros N. Pandis. *Atmospheric Chemistry and Physics: from Air Pollution to Climate Change*. John Wiley & Sons Inc., 1997.

[40] John. M. Stockie. The mathematics of atmospheric dispersion modeling. *SIAM Review*, 53(2):349–372, 2011.

[41] O.G. Sutton. A theory of eddy diffusion in the atmosphere. *Proceedings of the Royal Society of London. Series A*, 135(826):143–165, 1932.

[42] Geoffrey I. Taylor. Eddy motion in the atmosphere. *Philosophical Transactions of the Royal Society of London. Series A*, 215:1–26, 1915.

[43] Richard Turner and Tony Hurst. Factors influencing volcanic ash dispersal from the 1995 and 1996 eruptions of Mount Ruapehu, New Zealand. *Journal of Applied Meteorology*, 40(1):56–69, 2001.

[44] U.S. Environmental Protection Agency. *Guideline on Air Quality Models, Appendix W to Part 51, Title 40: Protection of the Environment, Code of Federal Regulations*, 2010.

[45] Y. Yang, L.T. Wilson, M.E. Makela, and M.A. Marchetti. Accuracy of numerical methods for solving the advection–diffusion equation as applied to spore and insect dispersal. *Ecological Modelling*, 109(1):1–24, 1998.

[46] Z. Zlatev. *Computer Treatment of Large Air Pollution Models.* Kluwer, 1995.

[47] Z. Zlatev and I. Dimov. *Computational and Numerical Challenges in Environmental Modelling.* Elsevier, 2006.