

RADIAL BASIS FUNCTIONS - FINITE DIFFERENCE (RBF-FD) FRAMEWORK TO
SOLVE THE TRANSPORT EQUATION IN ENVIRONMENTAL MODELING

A Thesis
by
DAVID MARHAO

Submitted to the School of Graduate Studies
at Appalachian State University
in partial fulfillment of the requirement for the degree of
MASTER OF ARTS

May 2023
Department of Mathematical Sciences

RADIAL BASIS FUNCTIONS - FINITE DIFFERENCE (RBF-FD) FRAMEWORK TO
SOLVE THE TRANSPORT EQUATION IN ENVIRONMENTAL MODELING

A Thesis
by
DAVID MARHAO
May 2023

APPROVED BY:

Nadun Kulasekera Mudiyansele, Ph.D.
Chairperson, Thesis Committee

Holly Hirst, Ph.D.
Member, Thesis Committee

Quinn Morris, Ph.D.
Member, Thesis Committee

Eric S. Marland, Ph.D.
Chairperson, Department of Mathematical Sciences

Marie Hoepfl, Ed.D.
Interim Dean, Cratis D. Williams School of Graduate Studies

Copyright by David Marhao 2023
All Rights Reserved

Abstract

David Marhao
B.S., Appalachian State University
M.A., Appalachian State University

Chairperson: Dr. Nadun Kulasekera Mudiyansele

In this project, a proposal for a framework to use the local radial basis functions (RBFs) method to approximate solutions for variations of the transport equation is introduced. The approach is also called the Radial Basis Functions - Finite Difference (RBF-FD) method. We discuss the difficulties of the method while also introducing solutions to resolve some of the issues that arise. The approach was used to solve the 1D transport equation that models contaminant water transport in a semi-infinite domain. Furthermore, the RBF-FD method was also used to solve the transport equation when decay and degradation are present in the problem with both finite and infinite domains. We then observed that the RBF-FD numerical approach gives high-accuracy approximations of the solutions transport equation up to machine accuracy.

Acknowledgments

First I want to acknowledge the great inspiration that Dr. Nadun Kulasekera Mudiyanse-
lage had on me to even start this project. Without his initial presentation, and also the help
he has given me both as a directed research and thesis advisor, this thesis would not be here
today. I want to also acknowledge both Dr. Holly Hirst and Dr. Bill Cook who took time
to help with some of the mathematics behind the thesis. I also want to acknowledge the
tremendous support from Graduate Student Office 209A, where the best colleagues I could
ask for resided and gave encouragement even in the most stressful periods. In this note I
also want to acknowledge Lisa Maggiore who also as a mentor pushed me to strive for more
while writing this thesis. Lastly I want to acknowledge the many prayers from my church
and prayer groups, and most importantly the wisdom provided by God to help persevere
and finish.

Table of Contents

1	Introduction	1
1.1	History of radial basis functions	1
1.2	RBF method definition and non-singularity	1
1.3	Challenges of using the RBF method	3
1.3.1	Non-singularity of the collocation matrices	3
1.3.2	Computational cost	4
1.3.3	Shape parameter	4
1.4	Remedies to some of the issues with the RBF method	6
1.4.1	Local RBF method	6
1.4.2	Enriching the basis with polynomials	7
1.5	Overview of partial differential equations	9
1.6	Using RBF-FD method to solve PDEs	11
1.7	Conclusion	13
2	Derivation of the transport equation	14
2.1	Transport equation and its physical properties	14
2.2	Mass continuity	15
2.3	Advection	17
2.4	Diffusion/Dispersion	18
2.5	General transport equation	19
2.6	Conclusion	20
3	RBF-FD framework for solving the transport equation	21

3.1	Governing equation	21
3.2	RBF-FD framework	22
3.3	Numerical results	25
3.4	Conclusion	28
4	Transport equation with degradation and decay	29
4.1	Decay and degradation	29
4.2	Problems with an infinite domain	30
4.3	Numerical results	32
4.4	Problems with a finite domain	35
4.5	Conclusion	39
5	Overview	40

1 Introduction

1.1 History of radial basis functions

The radial basis functions (RBF) method was first introduced in 1970 by Rolland Hardy when multiquadratic (MQ) radial functions were used for interpolation [7]. While polynomial interpolation looks decent on paper, it runs into serious issues when solving higher dimensional problems because the non-singularity of the interpolation matrix cannot be guaranteed. However, this issue in higher dimensional problems does not arise when using selected radial basis functions. For MQ RBFs, Micchelli, in 1986, proved non-singularity by showing the interpolation matrices were positive definite [15].

The method was then further pioneered by M.J.D. Powell and his collaborators at the University of Cambridge. With technology becoming more prevalent in the early 2000s, several monographs on RBFs (or with extensive RBF content) appeared in rapid succession between 2003 and 2007 written by Buhmann [2], Iske [13], Wendland [19], and Fasshauer [6]. These papers reflected an increasing interest that RBFs could be used as a practical computational procedure for increasingly larger scale applications [7].

1.2 RBF method definition and non-singularity

A radially symmetric real valued function $\phi : [0, \infty) \rightarrow \mathbb{R}$ is said to be a radial basis function centered at \vec{x}_k , if $\phi(\vec{x}) = \phi(\|\vec{x} - \vec{x}_k\|)$ for $\vec{x} \in \mathbb{R}^n$. The norm used in the definition is the Euclidean 2-norm. There are several RBFs that are used in literature. These RBFs can be generally separated into two categories: infinitely smooth and piecewise smooth

RBFs. Infinitely smooth RBFs include the Gaussian, Multiquadric, Inverse Multiquadric, and Inverse Quadratic RBFs, while piecewise smooth RBFs include Polyharmonic Spline and Thin plate spline RBFs. These RBFs that are commonly found in prevalent literature are defined in Table 1 [7].

Table 1: Most widely used RBFs

Types of RBFs	The Function Form
Gaussian (GA)	$e^{-(\varepsilon r)^2}$
Multiquadric (MQ)	$\sqrt{1 + (\varepsilon r)^2}$
Inverse Multiquadric (IMQ)	$\frac{1}{\sqrt{1+(\varepsilon r)^2}}$
Inverse Quadratic (IQ)	$\frac{1}{1+(\varepsilon r)^2}$
Polyharmonic Spline (PHS)	r^{2m+1} or $r^{2m} \log r$
Thin plate spline (TPS)	$r^2 \log r$

An important note about infinitely smooth RBFs is the use of a shape parameter (ε). This parameter governs how steep or shallow the RBFs become. A higher ε leads to a steeper RBF, while a smaller shape parameter value yields a flatter basis function. The conditioning of the algorithm is impacted by the choice of the shape parameter as well. To date, no computationally efficient method to choose an optimal shape parameter for a given problem has been found.

To solve an interpolation problem using the RBF methodology, assume we are given a set of data (\vec{x}_i, f_i) , where $i = 1, \dots, N$, $\vec{x}_i \in \mathbb{R}$, and $f_i \in \mathbb{R}$. The goal is to find a continuous function S that satisfies the interpolation conditions $S(\vec{x}_i) = f_i$. We then assume that the interpolant $S(\vec{x})$ is of the form:

$$S(\vec{x}) = \sum_{i=1}^N \lambda_i \phi(\|\vec{x} - \vec{x}_i\|) \quad (1)$$

From here, we can obtain a system of equations: $A\vec{\lambda} = \vec{f}$, where, \vec{f} is a vector of f_i , A is the $n \times n$ matrix of the coefficients of $\phi(\|\vec{x} - \vec{x}_i\|)$, and $\vec{\lambda}$ as a $n \times 1$ vector of unknown values. A is also typically called the collocation matrix [7]. The sketch of the RBF method

using this system of equations can be seen in the form of:

$$\begin{bmatrix} \phi(\|x_1 - x_1\|) & \phi(\|x_1 - x_2\|) & \cdots & \phi(\|x_1 - x_n\|) \\ \phi(\|x_2 - x_1\|) & \phi(\|x_2 - x_2\|) & \cdots & \phi(\|x_2 - x_n\|) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\|x_n - x_1\|) & \phi(\|x_n - x_2\|) & \cdots & \phi(\|x_n - x_n\|) \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix} \quad (2)$$

In higher dimensional pseudospectral (PS) methods, if we move two nodes (switch two rows), they could end up changing the sign of the determinant, which then implies that the determinant must be zero. Therefore, by the Mairhuber-Curtis Theorem, for a well-posed multivariate scattered data interpolation problem, the basis depends on the data locations, essentially ruling out the use of PS methods. The RBF method navigates around the shortcomings of the PS methods by using radial functions with Euclidean-2 norms defined by $\phi(\|\vec{x} - \vec{x}_i\|)$. Now when moving two nodes, this interchanges not only two rows but also two columns of A , leaving the sign of the determinant unaffected. Therefore, the singularity argument found using PS methods no longer applies.

This approach is called the Global RBF method because, per each node, the method utilizes all the nodes in the data set. The attractiveness of this method is that it is capable of producing highly accurate interpolants to sufficiently smooth underlying functions. However, there are several disadvantages to using the global RBF method, which will be explored in the next subsection.

1.3 Challenges of using the RBF method

1.3.1 Non-singularity of the collocation matrices

To obtain a unique solution for equation 2, matrix A has to be non-singular. There are theoretical proofs that guarantee the non-singularity of the matrix A for all infinitely smooth RBFs presented in Table 1. However, the methodology of the proofs is not universal for

all infinitely smooth RBFs. There are certain RBFs that produce strictly positive definite matrices. Given that positive definite matrices produce strictly positive eigenvalues, this guarantees non-singularity in the matrix. Some examples of RBFs that produce strictly positive definite matrices are GA and IMQ RBFs. These RBFs belong to a class of completely monotonic functions, which have properties that give rise to producing strictly positive definite matrices [6]. Some other RBFs, such as PHS RBFs, to date, do not have any theoretical results that guarantee the well-posedness of the interpolation problem. There are ways to circumnavigate this issue, which will be discussed in a later section.

1.3.2 Computational cost

The global RBF method can produce spectrally accurate interpolates for infinitely smooth underlying functions, provided that an infinitely smooth RBFs is used. However, the method is computationally expensive. The global approach leads to dense matrices. The computational cost associated with inverting the collocation matrix is then $O(N^3)$, where N is the total number of nodes. This means that using the global RBF method is only efficient for small data sets, causing a serious limitation as most problems have data sets that are sufficiently large in size to produce serious computation costs [16].

1.3.3 Shape parameter

One of the benefits of using the RBF methodology is its capacity to produce highly accurate interpolants to sufficiently smooth underlying functions. For example, if an infinitely smooth RBF is used, we can expect error decay as fast as $O(e^{-c/h})$, where h is the average distance between two nodes and c is a constant. This error decay changes for piece-wise smooth functions. Alternatively, when a PHS RBF of order m is used ($\phi(r) = r^m$), the error decay will then be $O(h^{m+1})$. To demonstrate, a cubic radial function $\phi(r) = r^3$ has a discontinuity in the 3rd derivative; therefore, the approximation will be $O(h^4)$ [16].

Even though infinitely smooth RBFs have the potential to yield spectral convergence,

the accuracy is heavily dictated by the shape parameter. An interesting phenomenon that occurs with infinitely smooth RBFs is if we find a shape parameter that gives great accuracy. Using this shape parameter could lead to a collocation matrix with a high condition number. The effect of the shape parameter on accuracy and stability is known as the uncertainty, or trade-off principle [7]. The effects of the shape parameter can be seen by graphing the interpolation error for a function $f(x)$ versus the shape parameter. Consider Figure 1.

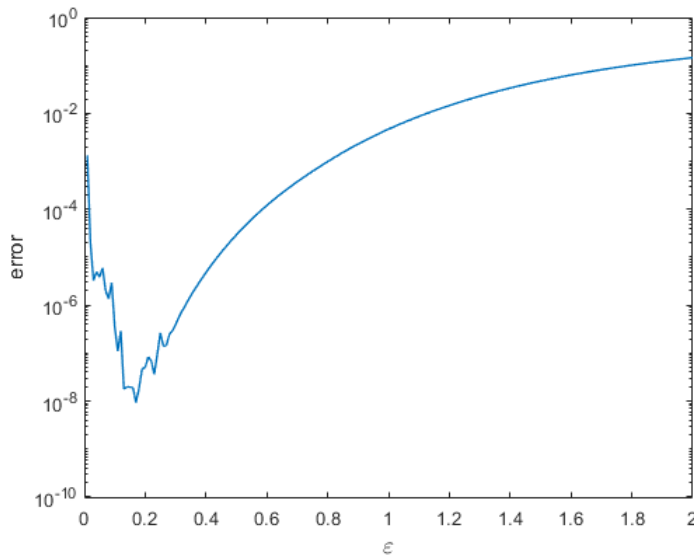


Figure 1: The interpolation error of $f(x) = \frac{59}{67+(x+1/7)^2+(y-1/11)^2}$ when $\varepsilon \rightarrow 0$

As shown in Figure 1, when $\varepsilon \rightarrow 0$, the error seems to decay up to an optimal point. If ε continues to decrease after the optimal point, the error starts to become erratic. This is due to the ill-conditioning of the collocation matrix. In 2002, Driscoll and Fornberg [4] showed that the underlying problem is indeed well posed, but the numerical algorithm is ill-conditioned. That means that the error is increasing mainly because of the numerical ill-conditioning. Furthermore, this means that for RBFs, we have to find an optimal shape parameter to produce the minimal error. However, to date, there is no clear way to find an optimal shape parameter as it depends on many factors.

There are computationally expensive ways to circumnavigate this issue. For instance, in 2004, Fornberg and Wright [9] introduced a new method called the Contour-Pade method,

which allows computations to be done stably when $\varepsilon \rightarrow 0$. However, similar to the global RBF method, this is limited to small data sets as it is computationally expensive. Later, a variable shape parameter approach was theorized by Fornberg and Zuev [10]. Furthermore, some other stable algorithms that are not limited by the size of data sets, such as RBF-QR and RBF-GA, were also introduced. However, again, they are still computationally expensive to use. Therefore, a solution to the issue of picking a shape parameter is to choose an RBF that does not use one, but the RBFs that do not use the shape parameter ε run into the issues of not producing non-singular matrices, as discussed earlier.

1.4 Remedies to some of the issues with the RBF method

1.4.1 Local RBF method

The global RBF method is not a computationally efficient method as it produces dense matrices. One workaround is to use the local RBF method. The idea is to use the n nearest neighboring nodes per each center node to gather information ($n \ll N$, N is the total number of nodes) instead of using the information from all the nodes in the domain. For instance, as shown in Figure 2, for the center node x_c , we calculate the distance to only its nearest neighbors.

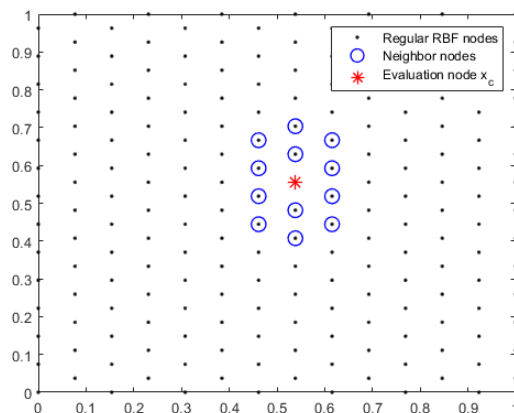


Figure 2: A node distribution for RBF-FD with local neighbor search

This means the approximation is a local one rather than being global one. When using this with the RBF method, this produces sparse matrices and cuts down on computation costs. There are several methods built into various mathematical programs which can facilitate choosing local nodes. Sparse matrices are desired when working with large volumes of nodes as they do not require as much storage or computation cost to use [6,16].

1.4.2 Enriching the basis with polynomials

A convenient way to circumnavigate the usage of the shape parameter is to use a piece-wise smooth RBF function such as PHS. However, the drawback is that piece-wise smooth RBFs such as PHS do not inherently produce non-singular matrices. One solution to this issue is to add low-order polynomial terms to the interpolation basis. Several of these polynomial terms are found in Table 2 [14].

Table 2: Augmented polynomial terms

Polynomial Degree	1D	2D	3D
0	1	1	1
1	x	x, y	x, y, z
2	x^2	x^2, xy, y^2	$x^2, xy, xz, y^2, yz, z^2$
3	x^3	x^3, x^2y, yx^2, y^3	$x^3, x^2y, x^2z, xy^2, xyz, xz^2, y^3, y^2z, yz^2, z^3$

This leads to a redefined RBF interpolant of:

$$S(\vec{x}) = \sum_{i=1}^N \lambda_i \phi(\|\vec{x} - \vec{x}_i\|) + \sum_{k=1}^M \mu_k p_k(\vec{x}), \vec{x} \in \mathbb{R}^s \quad (3)$$

Subject to the constraints:

$$\sum_{j=1}^N \lambda_j p_k(\vec{x}_j) = 0, k = 1, 2, \dots, M \quad (4)$$

Where p_1, \dots, p_M is a basis for the M dimensional space of polynomials of degree at most $m - 1$ in s variables and μ_k is an unknown constant. The constraint (4) will ensure that the system of equations will produce a unique solution and will have a square symmetric

system of equations [7]. A side effect of using augmented RBFs of the form (3) is that if the interpolation data represents a polynomial of degree $m - 1$, then the polynomial will be represented exactly by the new equation. Even further, the approximation will display algebraic convergence.

To implement the new equation (3) and the constraints from (4), a new system of equations is used as shown below:

$$\left[\begin{array}{ccc|ccc} & & & 1 & x_1 & y_1 \\ & A & & \vdots & \vdots & \vdots \\ & & & 1 & x_n & y_n \\ - & - & - & - & - & - \\ 1 & \dots & 1 & & & \\ x_1 & \dots & x_n & & 0 & \\ y_1 & \dots & y_n & & & \end{array} \right] \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ - \\ \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \\ - \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (5)$$

When constructing the new system of equations, an assumption was used that the augmented polynomial of up to degree one was used. However, generally the system of equations in 5 can be generalized as:

$$\begin{bmatrix} A & P \\ P^T & O \end{bmatrix} \begin{bmatrix} \vec{\lambda} \\ \vec{\mu} \end{bmatrix} = \begin{bmatrix} \vec{f} \\ \vec{0} \end{bmatrix} \quad (6)$$

Where A is the $N \times N$ collocation matrix defined in (1) and (2) and $\vec{\lambda}$ is a $N \times 1$ vector, and $\vec{\mu}$ is a $M \times 1$ vector. P is the transpose of the Vandermonde matrix, which is an important tool for interpolation [12].

The PHS radial functions belong to a class of strictly conditionally positive definite functions. It can be proved that the general equation (6) produces a uniquely solvable

system when matrix A is composed of PHS basis functions. Therefore, the combination of PHS RBFs and polynomials is an excellent solution for the issue of choosing an optimal shape parameter as it avoids the issue entirely. Coupled with the local RBF method, the issues discussed earlier are primarily solved. The polynomial augmentation, along with the local approach, is called the Radial Basis Function based - Finite Difference Method (RBF-FD).

RBF-FD method also mitigates the Runge Phenomenon when using polynomial approximations. This phenomenon indicates that by interpolating over equally spaced nodes on a fixed interval, the effect of increasing the number of nodes increases the error near boundaries significantly. This phenomenon, also known as the far field effect, is mitigated by the constraints in equation (4) [1]. The RBF-FD method is primarily used to discretize space operators of partial differential equations (PDEs). Therefore, the next subsection will provide a brief introduction to PDEs.

1.5 Overview of partial differential equations

A differential equation is defined as an equation that relates the derivatives of one or more unknown functions depending on one or more variables. These equations can be classified into two broad categories: ordinary differential equations (ODEs) and partial differential equations (PDEs). An ODE is a differential equation that contains only one independent variable. A PDE is a differential equation that contains one or more independent variables and partial derivatives of unknown functions. The order of a differential equation is defined as the highest-order derivative that appears in the equation [18].

Differential Equations can be further classified as linear or non-linear equations. An n th order differential equation is said to be linear if it can be written in the form:

$$a_n(x) \frac{d^n y}{dx^n} + a_{n-1}(x) \frac{d^{n-1} y}{dx^{n-1}} + \dots + a_1(x) \frac{dy}{dx} + a_0(x)y = g(x) \quad (7)$$

A linear ODE has to satisfy two conditions. First, the dependent variables and all their derivatives in the equation are of power one. Second, all the coefficients and the function in equation (7) are dependent only on the independent variable. If either of these two conditions are not satisfied, the differential equation is considered non-linear. Another classification is whether the differential equation is homogeneous or non-homogeneous. The homogeneous linear equation has $g(x)$ is 0; otherwise, it is considered a non-homogeneous linear equation. This nomenclature applies to non-linear equations as well.

PDEs can usually be further classified into three broad categories: elliptic, parabolic, and hyperbolic [18]. Consider the 2nd-order partial differential equation:

$$a(x, y) \frac{\partial^2 u}{\partial x^2} + b(x, y) \frac{\partial^2 u}{\partial x \partial y} + c(x, y) \frac{\partial^2 u}{\partial y^2} + d(x, y) \frac{\partial u}{\partial x} + e(x, y) \frac{\partial u}{\partial y} + f(x, y) u = g(x, y) \quad (8)$$

For the equation (8) to be elliptic, it has to satisfy the condition that $b^2 - 4ac < 0$. Subsequently for (8) to be considered hyperbolic if $b^2 - 4ac > 0$ and parabolic if $b^2 - 4ac = 0$. The approach to solving PDEs can differ based on which of these three categories the PDE falls into [18].

In solving differential equations or systems of differential equations, we could find a general solution or a particular solution. However, in practice, the particular solution that satisfies some specified conditions is most commonly preferred as it usually arises from an area of application. These specified conditions usually fall under two general types.

The first type of condition requires to have specific values at a single point, usually at the initial starting point. The problems that use this condition are traditionally called initial value problems because the system has an assumption to start evolving from a fixed initial point. The second type of condition requires to have specified values at different points, usually at the beginning and end of the domain for the problem. The problems with these conditions are traditionally known as boundary value problems. Boundary value

problems can also be further categorized based on how the boundary conditions are defined. The two most common types of boundary conditions are Dirichlet and Neumann boundary conditions. Dirichlet boundary conditions specify the values which the solution needs to have along the boundary, while Neumann boundary conditions specify the values of the derivatives along the boundary. The final step in obtaining the particular solution involves mostly algebraic operations that require either the initial or boundary conditions to be accounted for [18, 21]. If the equations are nonlinear, however, the solution is not nearly as easily obtained. Even finding analytical solutions to many linear differential equations can be impossible or cumbersome. Therefore, a common technique to circumnavigate this is to use numerical approximations. The accuracy of the approximations is dependent on the numerical technique being used. [18, 21].

The transport equation solved in this paper is a linear, elliptic partial differential equation with both Dirichlet and Neumann boundary conditions.

1.6 Using RBF-FD method to solve PDEs

To solve PDEs using the RBF-FD method, a Method of Lines approach is used. First, the RBF method is used to discretize the PDE in space. If the equation is time-dependent, this particular approach converts the original equation to a system of ODEs. This is then later solved using an appropriate numerical ODE solver. Looking at the methodology in more detail, consider the initial-boundary value problem:

$$\begin{cases} \frac{\partial}{\partial t} u(\vec{x}, t) = Lu(\vec{x}, t), & \vec{x} \in \Omega, t \in [0, T] \\ u(\vec{x}, t) = g(\vec{x}, t), & \vec{x} \in \partial\Omega, t \in [0, T] \\ u(\vec{x}, 0) = h(\vec{x}), & \vec{x} \in \Omega \end{cases} \quad (9)$$

Where L is any linear operator. An assumption is also given that $u(\vec{x}, t)$ is a linear

combination of RBFs and polynomials. However, n is the number of neighbors and hence, the following process takes place locally in each cluster of nodes centered around a center node x_c . Considering the equation:

$$u(\vec{x}, t) = \sum_{i=1}^n \lambda_i \phi(\|\vec{x} - \vec{x}_i\|) + \sum_{k=1}^M \mu_k p_k(\vec{x}), \quad \vec{x} \in \mathbb{R}^s \quad (10)$$

As discussed in Section 1.4.2, the equation leads to the system of equations:

$$\tilde{A} = \begin{bmatrix} A & P \\ P^T & O \end{bmatrix} \begin{bmatrix} \vec{\lambda} \\ \vec{\mu} \end{bmatrix} = \begin{bmatrix} \vec{u} \\ \vec{0} \end{bmatrix} \quad (11)$$

The next step is to apply the spatial operator L to the equation. This yields the equation:

$$Lu(\vec{x}, t) = \sum_{i=1}^n \lambda_i L\phi(\|\vec{x} - \vec{x}_i\|) + \sum_{k=1}^M \mu_k Lp_k(\vec{x}), \quad \vec{x} \in \mathbb{R}^s \quad (12)$$

Since we are finding the derivative at $\vec{x} = \vec{x}_c$, we evaluate the equation at $\vec{x} = \vec{x}_c$, and rewrite it as a system of equations. However, neither $\vec{\lambda}$ and $\vec{\mu}$ change when the operator L is applied. Therefore, since the collocation matrix A is non-singular, we can find both $\vec{\lambda}_i$ and $\vec{\mu}_i$ values from the equation and substitute them into the system of equations produced by equation (12). This then produces:

$$\begin{aligned} Lu(\vec{x}_c, t) &= \begin{bmatrix} L\phi_{\vec{x}=\vec{x}_c} & Lp_{\vec{x}=\vec{x}_c} \end{bmatrix} \begin{bmatrix} \vec{\lambda} \\ \vec{\mu} \end{bmatrix} \\ &= \left(\begin{bmatrix} L\phi_{\vec{x}=\vec{x}_c} & Lp_{\vec{x}=\vec{x}_c} \end{bmatrix}_{1 \times (n+M)} \begin{bmatrix} A & P \\ P^T & O \end{bmatrix}_{(n+M) \times (n+M)}^{-1} \right) \begin{bmatrix} \vec{u} \\ \vec{0} \end{bmatrix} \end{aligned} \quad (13)$$

The $1 \times (n + M)$ row vector found in the right-hand side of equation (13) contains

differentiation weights of the operator L . The first n weights are then extracted, while the last M weights are discarded. These n weights are used to populate the differentiation matrix D . The differentiation matrix (DM) defined as $DM = BA^{-1}$ where B is the $1 \times (n + M)$ row vector in equation (13). This is the discretized version of the operator L . The process used to construct the DM is parallelizable, or in other words, the differentiation weights for each of the center nodes can be calculated simultaneously. The computation cost for this process is $O(n^2 \cdot N)$, where N is the total number of nodes.

1.7 Conclusion

There are advantages to using the RBF-FD method to solve PDEs. It is not computationally expensive to run and can easily be adapted into higher dimensions. If the algorithms for the 1D case are known, they can easily be adapted/programmed to solve higher dimensional problems. Since the RBFs can be used with a scattered node distribution, the method is meshless [6]. Recent node placing algorithms have also become readily available [8] to use generate quasi-uniform node distributions in 2D and 3D. The goal of this thesis is to use the RBF-FD method to solve the transport equation and its variants. This has several applications in various geosciences. Chapter 2 of this thesis defines what the transport equation is and derives the equation based on mass continuity principles. Finally, in Chapters 3 and 4, the RBF method is adopted to solve the transport equation under different scenarios numerically.

2 Derivation of the transport equation

2.1 Transport equation and its physical properties

Transport is defined as the processes that determine the distribution of biogeochemical species or heat in an environmental compartment [5]. This definition can be further simplified by representing transport as interactions of physical processes with an effect on either species or heat. There are other relevant processes for the environment, like degradation and decay and sorption. However, these processes are not usually considered pure transport processes. Degradation and decay are discussed in their own chapter, while sorption is beyond the scope of this thesis.

The process that defines transport can be split into two broad general types: advection and diffusion/dispersion. The advection process refers to a particle that is being purely moved from one place to another by a flow field [5]. Diffusion and dispersion refer to transport processes that originate from concentration differences [5], essentially to a tendency to equalize concentration gradients in a system. Differential equations can further describe these processes. The process of deriving the differential equation for transport is by applying the principle of mass conservation and Fick's Law. However, when dealing with heat transport, a different equation is derived for the temperature by applying the principle of energy conservation and by using Fourier's Law [5].

2.2 Mass continuity

The general continuity equation expresses the fundamental formulation of a conservation principle. The mathematical formulation of mass conservation considers the change of mass during a small time Δt within a control volume of spacing; Δx , Δy , and Δz , representing the directions in three-dimensional space [5]. There are two ways to approach the calculation. The first is to consider the mass within the control volume at the beginning and at the end of the time period, then calculate the difference. The other method is to balance all the fluxes across the boundaries of the volume. This means that fluxes into the volume have to be taken as positive, while the fluxes leaving the volume are negative. For three-dimensional space, there are six faces of control volume to account for, while in one-dimensional space, there are only two to account for.

Using the first method, consider the mass at the beginning and end of the period t and $t + \Delta t$ given by the equation:

$$\theta \cdot c(x, t) \cdot \Delta x \Delta y \Delta z \quad \text{and} \quad \theta \cdot c(x, t + \Delta t) \cdot \Delta x \Delta y \Delta z,$$

where θ denotes the share of the total volume, or more often the porosity [7]. $\Delta x \Delta y \Delta z$ is the volume, and c denotes the concentration, measured as mass/volume. This gives the change of mass per time written as:

$$\theta \cdot \frac{c(x, t + \Delta t) - c(x, t)}{\Delta t} \cdot \Delta x \Delta y \Delta z$$

Using the second method, fluxes in the x -direction across the faces of the control volume are given by:

$$\theta j_{x-}(x, t) \Delta y \Delta z \quad \text{and} \quad \theta j_{x+}(x, t) \Delta y \Delta z$$

Where j_{x-} denotes the flux in mass per area in the negative x -direction, and analogously j_{x+} denotes the flux in mass per area in the positive x -direction. The term $\theta \Delta y \Delta z$, in this case, denotes the area through which the flow takes place. The balance between both flux terms is thus given by:

$$\theta(j_{x-}(x, t) - j_{x+}(x, t)) \Delta y \Delta z$$

For simplicity, only the one-dimensional state is considered here, and the other fluxes across the four other faces are assumed to be zero. Since both formulations measure the change of mass and thus need to be equal, then:

$$\theta \frac{c(x, t + \Delta t) - c(x, t)}{\Delta t} \cdot \Delta x \Delta y \Delta z = \theta (j_{x-}(x, t) - j_{x+}(x, t)) \Delta y \Delta z \quad (14)$$

Division through the volume $\Delta x \Delta y \Delta z$ and θ in equation (14) yields:

$$\frac{c(x, t + \Delta t) - c(x, t)}{\Delta t} = - \frac{j_{x-}(x, t) - j_{x+}(x, t)}{\Delta x} \quad (15)$$

From equation (15), a differential equation can be derived by the transition of the finite grid spacing Δx and time step Δt to infinitesimal expressions by the limits $\Delta x \rightarrow 0$ and $\Delta t \rightarrow 0$. It follows then as:

$$\frac{\partial c}{\partial t} = - \frac{\partial}{\partial x} j_x, \quad (16)$$

which is a differential formulation for the principle of mass conservation. There is a presumption for the differentiation procedure that the functions c and j_x are sufficiently smooth and differentiable. Equation (16) is valid for one-dimensional transport and is the basis for the mathematical analysis of transport processes. Equation (16) is also valid if there are no internal sources or sinks for the concerned biogeochemical species [5]. These are understood as any process that can destroy or create mass. A source is a process that creates mass, while a sink is a source that destroys mass [5]. The mathematical formulation of equation (16) can be extended to consider sources and sinks. If the source or sink rate can be represented by $q(x, t)$, which may vary spatially and temporally, we only have to add a corresponding integral term.

$$\int_{\Delta x} \int_{\Delta t} q(x, t) dt dx$$

on the right side of equations (14) and (15). The term is positive if mass is added by a source and negative if the mass is removed by a sink. Coupling the integral term with the derivation of (16), the integral term has to be differentiated. This leads to the general transport equation in one-dimensional space:

$$\theta \frac{\partial c}{\partial t} = -\frac{\partial}{\partial x} \theta j_x + q \quad (17)$$

2.3 Advection

The process that governs advection is addressed in this section. While equation (17) is the general transport equation, the equation is hard to adapt to higher dimensions. By addressing the concept of advection, an adaption that can be used in a higher dimension is formed. Using the general derivation (17) and a variable A representing mass, momentum, or energy, the equation can be expressed as the one-dimensional equation:

$$\frac{\partial}{\partial t} A = \frac{\partial}{\partial x} j_{Ax} + Q \quad (18)$$

where j_{Ax} represents the fluxes in the x -direction. In the term Q , all sources and sinks are gathered. The flux term is a component of the flux vector j_A corresponding to the spatial directions. In the small but finite time interval Δt , the amount of A per unit volume changes from $A(x, t)$ to $A(x, t + \Delta t)$. The total amount of change in the control volume is thus given by $(A(x, t + \Delta t) - A(x, t))\Delta x$. In the x -direction, the fluxes across the faces are given by $(j_{Ax}(x + \Delta x/2, t) - j_{Ax}(x - \Delta x/2, t))\Delta t$. The assumption is that the time step Δt is small so that the change of the flux terms and also of the sinks and sources during that time can be neglected. Both expressions of the change within the control volume with a time step have to be equal. So by applying the changes to (18), a new equation is formed, which is

expressed as:

$$(A(x, t + \Delta t) - A(x, t))\Delta x = (j_{Ax}(x + \Delta x/2, t) - j_{Ax}(x - \Delta x/2, t))\Delta t + Q\Delta x\Delta t \quad (19)$$

Equation (19) is simplified in two steps. First, we divide through the product of all spatial extensions and the finite time step $\Delta x\Delta t$ and obtains:

$$\frac{A(x, t + \Delta t) - A(x, t)}{\Delta t} = \frac{j_{Ax}(x + \Delta x/2, t) - j_{Ax}(x - \Delta x/2, t)}{\Delta x} + Q \quad (20)$$

The second step is the transition from finite steps to infinitesimal steps in equation (20), $\Delta x \rightarrow \partial x, \Delta t \rightarrow \partial t$, according to the differential calculus in order to get the continuity equation in the formulation given by (18). Using the vector notation, the same equation can be expressed with higher dimensions as:

$$\frac{\partial A}{\partial t} = \nabla \cdot \mathbf{j}_A + Q \quad (21)$$

2.4 Diffusion/Dispersion

In order to achieve this for diffusive/dispersive flux, an empirical relationship has to be introduced, e.g., Fick's Law [5]. A system with initial concentration differences will finally reach a constant concentration level if no other processes are present. The empirical (1st) Fick's Law is a quantification of diffusive flux for fluid phase stated by the equation:

$$\mathbf{j} = -D\nabla c \quad (22)$$

The diffusive flux is proportional to the negative concentration gradient. The factor of proportionality is the diffusion constant or diffusivity D . The ∇ operator is also used here to represent the vector formed by multiplying the scalar variable c .

Another generalization of Fick's Law is necessary if advection is also present. It can be observed that in a fluid flowing through a homogeneous porous medium, the diffusivity, as shown in (22), is not constant but shows itself a strong dependency on the flow velocity. This is what is referred to as dispersion[5]. This can be written as the equation:

$$\mathbf{j} = -\mathbf{D}\nabla c \quad (23)$$

Where \mathbf{D} represents the dispersion tensor, defined by the equation:

$$\mathbf{D} = (\tau D_{mol} + \alpha_T v)\mathbf{I} + \frac{\alpha_L - \alpha_T}{v}\mathbf{v}\mathbf{v}^T$$

Where \mathbf{I} is the unity matrix, the elements of matrix $\mathbf{v}\mathbf{v}^T$ contain the products of the velocity components, α_L is the longitudinal dispersivity, and α_T is the transversal dispersivity [7]. As noted, the dispersion equation is essentially the same as the diffusivity.

2.5 General transport equation

When both advection and diffusion/dispersion are taken into account, the flux vector in x -direction results as the sum of both contributions. This gives us the equation:

$$j_x = -D\frac{\partial c}{\partial x} + vc \quad (24)$$

where in the diffusivity D , different contributions have to be considered. This can also be written for higher dimensions as:

$$\mathbf{j} = -\mathbf{D}\nabla c + \mathbf{v}c$$

From here, we use equation (24) to replace the flux terms in the mass conservation equation (17). This gives us:

$$\theta \frac{\partial c}{\partial t} = \frac{\partial}{\partial x} \theta \left(D \frac{\partial c}{\partial x} - vc \right) + q$$

In the case of a constant D , the equivalent formulation is written as:

$$\theta \frac{\partial c}{\partial t} = \theta D \frac{\partial^2 c}{\partial x^2} - \theta v \frac{\partial c}{\partial x} + q \quad (25)$$

If we want to use equation (25) for higher dimensional problems, we can use the ∇ -operator to obtain:

$$\theta \frac{\partial c}{\partial t} = \nabla \cdot \theta (\mathbf{D} \nabla c - \mathbf{v}c) + q$$

2.6 Conclusion

In chapter 2, the one-dimensional transport equation (25) was derived by taking using mass continuity, advection, and diffusion/dispersion phenomenon. While the equation derived is one-dimensional, it can be easily modified to derive a higher-dimensional equation. In chapters 3 and 4, we will explore how to solve the transport equation using the RBF-FD method numerically.

3 RBF-FD framework for solving the transport equation

3.1 Governing equation

In this chapter, we develop an RBF-FD framework to approximate solutions to the 1D transport equation derived by equation (25) in Chapter 2. Recall the equation is:

$$\theta \frac{\partial c}{\partial t} = \theta D \frac{\partial^2 c}{\partial x^2} - \theta v \frac{\partial c}{\partial x} + q$$

An analytical solution for the transport equation (25) with constant coefficients and $q = 0$ was given by Ogata and Banks [17]. In their analysis, the authors were looking for solutions to the differential equations of longitudinal dispersion in porous media. There, they defined D to represent the dispersion coefficient, v as the average velocity of fluid or superficial velocity/porosity of medium, and c as the concentration of solute in the fluid. In addition, q is assumed to be zero implying that there are no sources or sinks present. The analytical solution is:

$$c(x, t) = \frac{c_{in}}{2} \left(\operatorname{erfc} \left(\frac{x - vt}{2\sqrt{Dt}} \right) + \exp \left(\frac{v}{D} x \right) \operatorname{erfc} \left(\frac{x + vt}{2\sqrt{Dt}} \right) \right) \quad (26)$$

“erfc” denotes the complementary error function, which is defined as follows:

$$\operatorname{erfc}(\xi) := 1 - \frac{2}{\sqrt{\pi}} \int_0^\xi \exp(-\varsigma^2) d\varsigma$$

where

$$\operatorname{erfc}(\xi) := 1 - \operatorname{erf}(\xi) \quad ; \quad \operatorname{erf}(\xi) := \frac{2}{\sqrt{\pi}} \int_0^{\xi} \exp(-\zeta^2) d\zeta$$

The solution given by (26) is valid only for a semi-infinite half-space: $x \geq 0$ for the initial condition

$$c(x, t = 0) = 0$$

and the boundary conditions:

$$c(x = 0, t) = c_{in} \quad c(x = \infty, t) = 0$$

The boundary conditions model is a scenario where the solute is not present in the system initially but is later introduced into the system with an inflow of concentration c_{in} at position $x = 0$. In addition, far away from the boundary, the concentration is assumed to be constant.

3.2 RBF-FD framework

An RBF-FD method-based numerical method is proposed to approximate the solutions of equation (25). The known true solution allows us to conduct a numerical convergence analysis at the end. Furthermore, from there we can develop RBF-FD approaches to approximate the solutions of other variants of the transport equations that may or may not have analytical solutions. In terms of solving the equation (25), the semi-infinite domain poses an issue. A discrete method needs a finite domain. There are no node distributions that can cover the entire real line.

We propose a workaround for this issue is to perform a transformation on the transport equation using the hyperbolic tangent function. This will remap the domain of the problem

from an infinite domain to a finite domain between 0 and 1. Taking the equation (25), and also the assumption made by Ogata and Banks that there are no sources or sinks, the term θ can be eliminated. This gives us the equation:

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} - v \frac{\partial c}{\partial x}$$

Let $\xi = \tanh(x)$, and note that $\tanh^2(x) + \operatorname{sech}^2(x) = 1$. Then:

$$\frac{\partial \xi}{\partial x} = \operatorname{sech}^2(x) = 1 - \xi^2$$

Now by applying the chain rule, the term $\frac{\partial c}{\partial x}$ becomes:

$$\frac{\partial c}{\partial x} = \frac{\partial c}{\partial \xi} \frac{\partial \xi}{\partial x} = (1 - \xi^2) \frac{\partial c}{\partial \xi}$$

and the term $\frac{\partial^2 c}{\partial x^2}$ becomes:

$$\frac{\partial^2 c}{\partial x^2} = \frac{\partial}{\partial x} \left(\frac{\partial c}{\partial x} \right) = \frac{\partial}{\partial \xi} \left(\frac{\partial c}{\partial \xi} \frac{\partial \xi}{\partial x} \right) \frac{\partial \xi}{\partial x}$$

Now by applying the chain rule and the product rule, the second derivative term becomes:

$$\left[\frac{\partial^2 c}{\partial \xi^2} \frac{\partial \xi}{\partial x} + \frac{\partial c}{\partial \xi} \frac{\partial}{\partial \xi} \left(\frac{\partial \xi}{\partial x} \right) \right] \frac{\partial \xi}{\partial x} = \left[(1 - \xi^2) \frac{\partial^2 c}{\partial \xi^2} - 2\xi \frac{\partial c}{\partial \xi} \right] (1 - \xi^2)$$

When redistributing, the final equation becomes:

$$\frac{\partial^2 c}{\partial x^2} = (1 - \xi^2)^2 \frac{\partial^2 c}{\partial \xi^2} - 2\xi(1 - \xi^2) \frac{\partial c}{\partial \xi}$$

The final proposed equation using the transformation is:

$$\frac{\partial c}{\partial t} = D \left[(1 - \xi^2)^2 \frac{\partial^2 c}{\partial \xi^2} - 2\xi(1 - \xi^2) \frac{\partial c}{\partial \xi} \right] + v(1 - \xi^2) \frac{\partial c}{\partial \xi} \quad (27)$$

With equation (27), the proposed RBF-FD method has all the necessary conditions satisfied prior to the setup. The method requires some knowledge of coding. First, a node distribution is generated from 0 to 1. Let us assume that we have N total nodes. Then the differentiation matrices are set up. These matrices represent the terms $\partial c/\partial \xi$ and $\partial^2 c/\partial \xi^2$. This is done by first differentiating the original RBF once and twice with respect to ξ . Afterward, the local collocation matrices and the local weights for the differential operators are found, then they are assigned to the differentiation matrices.

To resolve the discretization of the time steps, a Backward Euler approach was taken. This is an implicit method where the unknown future value is on both sides of the equation. Using the equation:

$$\frac{c^{i+1} - c^i}{\Delta t} = \tilde{D}c^{i+1} \quad (28)$$

where c^i represents the known value at the current time step, c^{i+1} the unknown value one time step forward, and \tilde{D} represents the discretized differential operators from equation (27). That is, \tilde{D} is RBF-FD discretized $D \left[(1 - \xi^2)^2 \frac{\partial^2}{\partial \xi^2} - 2\xi(1 - \xi^2) \frac{\partial}{\partial \xi} \right] + v(1 - \xi^2) \frac{\partial}{\partial \xi}$. Solving equation (28) for c^i we obtain the equation:

$$c^{i+1} - \Delta t \tilde{D} c^{i+1} = c^i$$

After factoring the term c^{i+1} , the equation becomes

$$c^{i+1}(I - \Delta t \tilde{D}) = c^i$$

Where I is the $N \times N$ identity matrix. Assume, $(I - \Delta t \tilde{D})$ as \mathbf{D} , and then we have a

linear system of equations:

$$\mathbf{D}c^{i+1} = c^i \quad (29)$$

that can be solved using any linear system of equations solver. We used MATLAB's backslash operator to solve the system of equations (29). In order to handle Dirichlet boundary conditions, the first and the last rows of \mathbf{D} were replaced by the rows $\begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}_{1 \times N}$ and $\begin{bmatrix} 0 & 0 & \dots & 1 \end{bmatrix}_{1 \times N}$ respectively. Then the first and last entries of the right-hand side of the equation (29) were replaced with appropriate boundary concentration values.

3.3 Numerical results

This subsection provides numerical results for approximating the solution for the equation:

$$\frac{\partial c}{\partial t} = D \left[(1 - \xi^2)^2 \frac{\partial^2 c}{\partial \xi^2} - 2\xi(1 - \xi^2) \frac{\partial c}{\partial \xi} \right] + v(1 - \xi^2) \frac{\partial c}{\partial \xi} \quad (30)$$

Subjected to:

$$c(\xi, t = 0) = 0$$

and the boundary conditions:

$$c(\xi = 0, t) = c_{in} \quad c(\xi = 1, t) = 0$$

The parameters used for the analysis were that the velocity v and inflow value c_{in} were set to 1, and the diffusivity D was set to 0.1. The convergence depends on the degree of the augmented polynomial. We observed that the convergence is two less than the polynomial degree, i.e., is, $p - 2$ is the order of convergence, where p is the degree of the polynomial. The convergence analysis is done by solving the equation of $E = ch^p$, where p is the order

of convergence, E is the error, h is the number of nodes, and c is a constant. By applying logarithms to both sides, we obtain the linear equation $\log(E) = p \log(h) + \log(c)$ where p represents the slope. The number of nodes for this analysis started at $N = 100$ and doubled in number to $N = 1600$ nodes.

In Figure 3, a third, fourth, and fifth-degree polynomial is compared to the first, second, and third-order convergence, respectively. The results show that the error decay of the approximations is parallel to their respective orders of convergence. For instance, following the pattern, we can expect 4th-order convergence when a 6th-degree polynomial is used with RBFs.

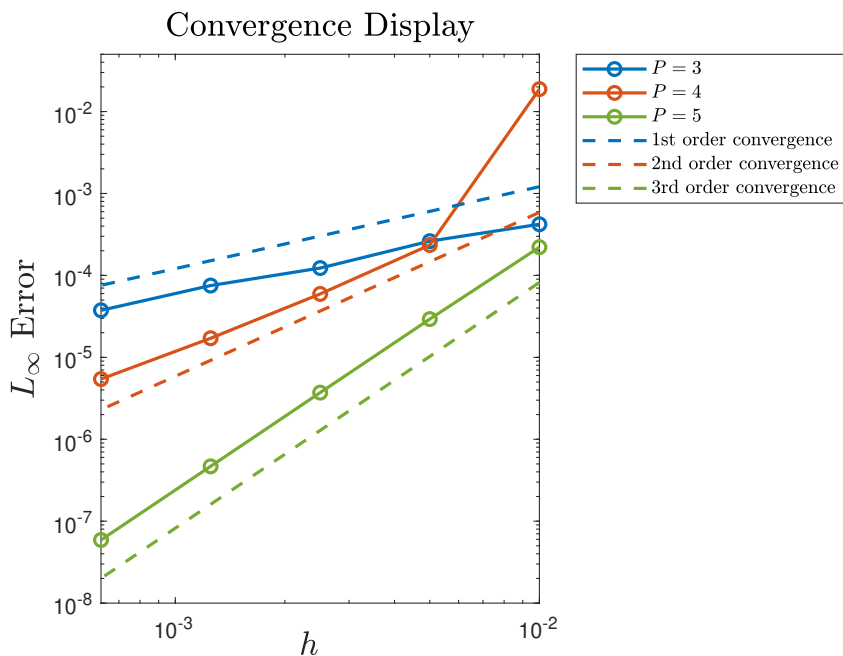


Figure 3: Error decay with respect to $h \rightarrow 0$ for different augmented polynomials.

Furthermore, Figure 4 shows the approximated solution as a space-time surface plot. In addition, another numerical test was conducted with different parameters. For the second test, in Figure 5, the solution is visualized for the first 12 inches of the domain. For this test, the chosen parameters were $D = 0.5$, $v = 0.6$, and $c_{in} = 1$ [20]. The approximated solution at each hour was plotted along with the corresponding true solution. The true solution at

each hour is plotted with dashed lines, and the approximated solution is with other curves. Again, we could observe that the RBF-FD method provides a good approximation of the true solution.

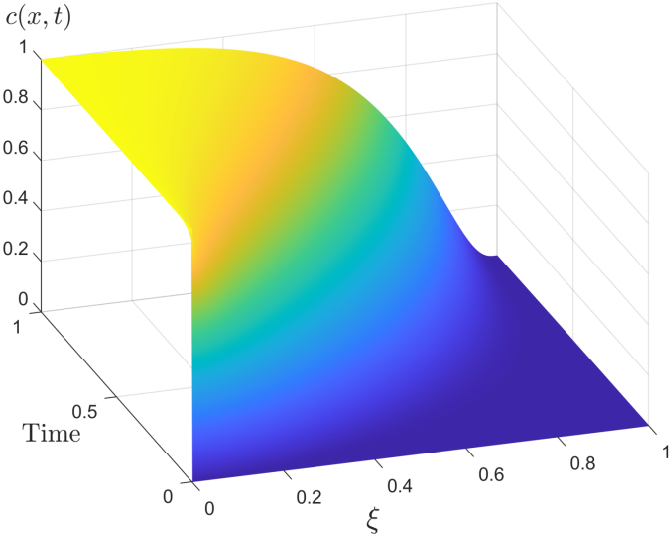


Figure 4: The solution of 1D transport visualized as a surface plot.

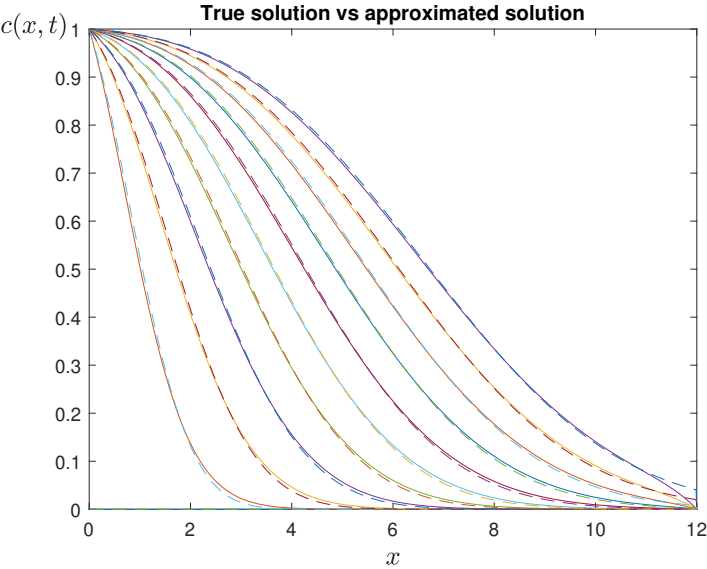


Figure 5: The true solution vs. approximated solution at each hour. The true solution is depicted from dashed lines and the approximated solution from other lines.

3.4 Conclusion

In this chapter, we introduced a methodology for solving the 1D transport equation on a semi-infinite domain. We proposed a hyperbolic tangent transformation to map the semi-infinite domain to a finite domain. We also showed that the RBF-FD method could produce high-accuracy results when solving the said equation. This latter part is shown by Figure 3. For instance, we showed that when a 4th-degree polynomial was appended to the RBF, the error decay followed a 2nd-order convergence. Overall, the rate of convergence is observed to be $O(h^{p-2})$, where p is the degree of the appended polynomial. The results of the simulation in Figure 5 show that the true and approximate solutions are highly similar.

4 Transport equation with degradation and decay

4.1 Decay and degradation

In this chapter, an exploration of the effects of decay and degradation on the transport equation will be discussed. While not a true transport phenomenon, all organic matter and organic substances are subject to degradation. Bacteria usually cause degradation in a biochemical process, and the conditions around the bacteria determine the activity. However, decay generally is used for physical or chemical processes that cause a loss of substance. Both of these processes can be simply expressed mathematically by the term q in equation (25), which represents the presence of sources or sinks. The general approach recognizes losses q being proportional to a power of the concentration c . This is written as:

$$q = -\lambda c^n \quad (31)$$

Where the integer n is the order of degradation, and λ is the so-called decay or degradation constant. λ is generally dependent on many variables present in the environment of the system that is in question. This new term given by (31) is substituted into the equation given by (25) and becomes:

$$\theta \frac{\partial c}{\partial t} = \theta D \frac{\partial^2 c}{\partial x^2} - \theta v \frac{\partial c}{\partial x} - \lambda c^n \quad (32)$$

With this equation, it is possible to treat transport, decay, and degradation simultaneously. Several common physical examples of this equation were explored by Wexler in

his 1992 paper [20]. In the paper, the author explores analytical solutions for the one-dimensional form of the transport equation (32) for the dispersion phenomenon in soil or adsorption columns. In the analysis, four different cases are observed for infinite and finite domains. The work done by Wexler is comparable to our research. From these cases, we consider two of them.

4.2 Problems with an infinite domain

The first analysis is on the 1D transport equation on an infinite domain. An example of transport in an infinite system might be the injection of a solute into the center of a long soil column. In this example the interest is in the spread of the solute in both the up-gradient and down-gradient of the source. The conditions of the problem are similar to that of the equation solved in chapter 3. Here the partial differential equation has initial conditions:

$$c(x, t = 0) = 0$$

and the boundary conditions:

$$c(x = 0, t) = c_0 \quad \& \quad \frac{\partial c}{\partial x}(x = \infty, t) = 0$$

Van Genuchten has already produced the analytical solution for this partial differential equation[11], and was modified by Wexler [20]. The solution we will use is a modified version of the one provided by Wexler, written as:

$$c(x, t) = c_0 \exp(-\lambda t) \left(1 - \frac{1}{2} \operatorname{erfc} \left(\frac{x - vt}{2\sqrt{Dt}} \right) - \frac{1}{2} \exp \left(\frac{vx}{D} \right) \operatorname{erfc} \left(\frac{x + vt}{2\sqrt{Dt}} \right) \right) + \frac{c_{in}}{2} \left(\exp \left(\frac{v - u}{2D} x \right) \operatorname{erfc} \left(\frac{x - ut}{2\sqrt{Dt}} \right) + \exp \left(\frac{v + u}{2D} x \right) \operatorname{erfc} \left(\frac{x + ut}{2\sqrt{Dt}} \right) \right), \quad (33)$$

where $u = \sqrt{v^2 - 4\lambda D}$. The solution consists of two parts; the first describes the decline

of the original concentration c_o , and the second is the change of the inflow concentration c_{in} in the 1D set-up. As this situation is on an infinite domain, a similar issue arises as in chapter three. There we proposed a transformation of the 1D equation using the hyperbolic tangent function. The equation derived in (32) incorporates the decay factor. However, it does not introduce any new derivatives. Therefore the equation derived in chapter 3 (27) becomes:

$$\frac{\partial c}{\partial t} = D \left[(1 - \xi^2)^2 \frac{\partial^2 c}{\partial \xi^2} - 2\xi(1 - \xi^2) \frac{\partial c}{\partial \xi} \right] + v(1 - \xi^2) \frac{\partial c}{\partial \xi} - \lambda c^n \quad (34)$$

A similar approach to chapter 3 is used to resolve the discretization of the time steps. The main difference that equation (28) does not resolve is the λc^n term. For simplicity, the order of decay will be set to 1. The λc term is then added to the end of (28) resulting in:

$$\frac{c^{i+1} - c^i}{\Delta t} = \tilde{D}c^{i+1} - \lambda c^{i+1}, \quad (35)$$

where c^i represents the known value at the current time step, c^{i+1} the unknown value one time step forward, and \tilde{D} represents the discretized differential operators from equation (27). That is \tilde{D} is RBF-FD discretized $D \left[(1 - \xi^2)^2 \frac{\partial^2}{\partial \xi^2} - 2\xi(1 - \xi^2) \frac{\partial}{\partial \xi} \right] + v(1 - \xi^2) \frac{\partial}{\partial \xi}$. The difference between this discretization and that of equation (34) is that the λc term is taken out. Solving equation (35) for c^i we obtain the equation:

$$c^{i+1} - \Delta t \tilde{D} c^{i+1} - \Delta t \lambda c^{i+1} = c^i$$

After factoring the term c^{i+1} , the equation becomes

$$c^{i+1}(I - \Delta t \tilde{D} - \Delta t \lambda I) = c^i,$$

where I is the $N \times N$ identity matrix. We then assume the matrix $(I - \Delta t \tilde{D} - \Delta t \lambda I)$ as \mathbf{D} , and then we have a linear system of equations

$$\mathbf{D}c^{i+1} = c^i, \quad (36)$$

that can be solved using any linear system of equations solver. We used MATLAB's backslash operator to solve the system of equations (36). In order to handle both Dirichlet Neumann boundary conditions, the first and the last rows of \mathbf{D} were replaced by the rows $\begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}_{1 \times N}$ and $\begin{bmatrix} D_x & D_x & \dots & D_x \end{bmatrix}_{1 \times N}$ respectively. The term D_x represents the terms from the differentiation matrix for the first derivative. Then the first and last entries of the right-hand side of the equation (36) were replaced with appropriate boundary concentration values.

4.3 Numerical results

This subsection provides numerical results for approximating the solution for the equation (32):

$$\theta \frac{\partial c}{\partial t} = \theta D \frac{\partial^2 c}{\partial x^2} - \theta v \frac{\partial c}{\partial x} - \lambda c^n$$

Subject to:

$$c(x, t = 0) = 0$$

and the boundary conditions:

$$c(x = 0, t) = c_0 \quad \text{and} \quad \frac{\partial c}{\partial x}(x = \infty, t) = 0$$

First, we solved the equation with $\lambda = 0$ and later compared the solutions for $\lambda \neq 0$. The

parameters used were: the velocity $v = 0.6$ in/hr, the dispersion factor $D = 0.6$ in²/hr, the decay factor $\lambda = 0.0$ per hr, and the initial concentration $c_0 = 1.0$ mg/L. We also assume that the fluid is of constant density and viscosity, the solute may be subject to first-order chemical transformations, the flow is in the x-direction only with constant velocity, and the longitudinal dispersion coefficient (D) is constant. Figure 6 shows the solutions for times T - 2.5 hrs, 5 hrs, 10 hrs, 15 hrs, and 20 hrs.

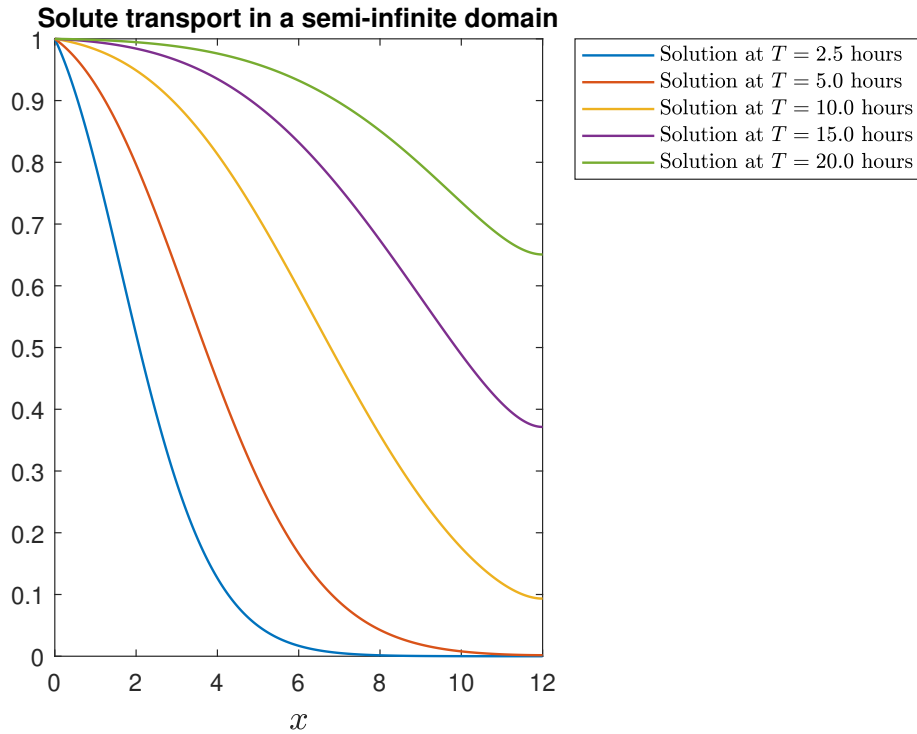


Figure 6: The solution of equation (32) (infinite domain) for times (hours) T - 2.5, 5, 10, 15, and 20.

Next, we solved the equation (32) with the same parameters and assumptions, except we chose $\lambda = 0.01$ per hr and $\lambda = 0.5$ per hr. Figures 7 and 8 show the solutions for times T - 2.5 hrs, 5 hrs, 10 hrs, 15 hrs, and 20 hrs for $\lambda = 0.01$ per hr and $\lambda = 0.5$ per hr, respectively. The effect of λ can be clearly seen on the approximations as when we increase λ , we observe a faster decay in concentration.

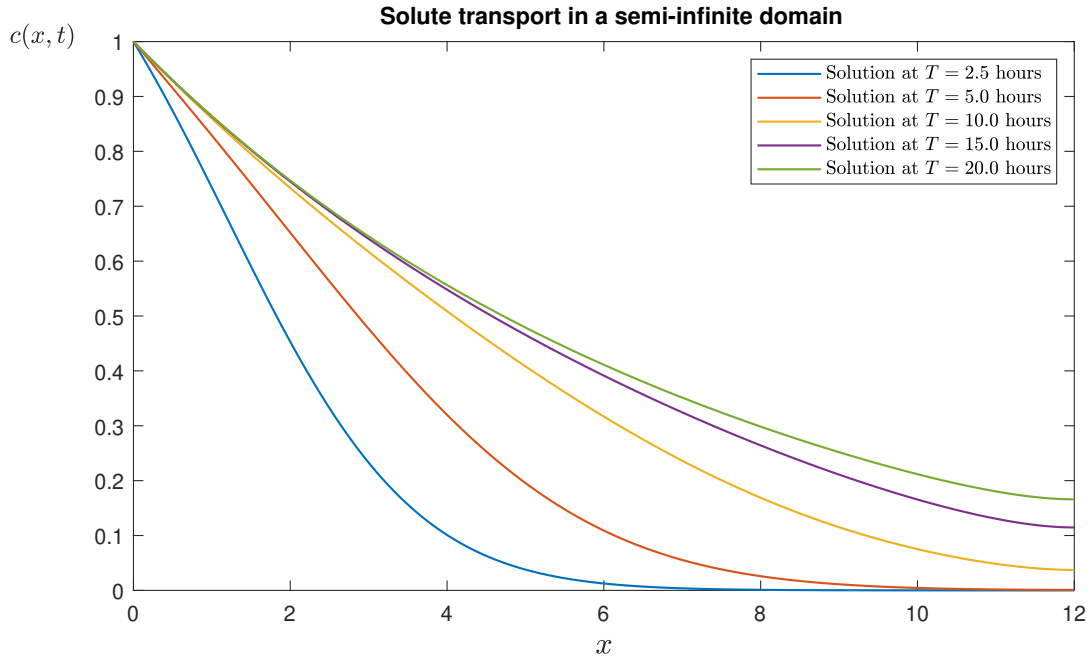


Figure 7: The solution of equation (32) (infinite domain) for times (hours) T - 2.5, 5, 10, 15, 20, and $\lambda = 0.01$ per hr.

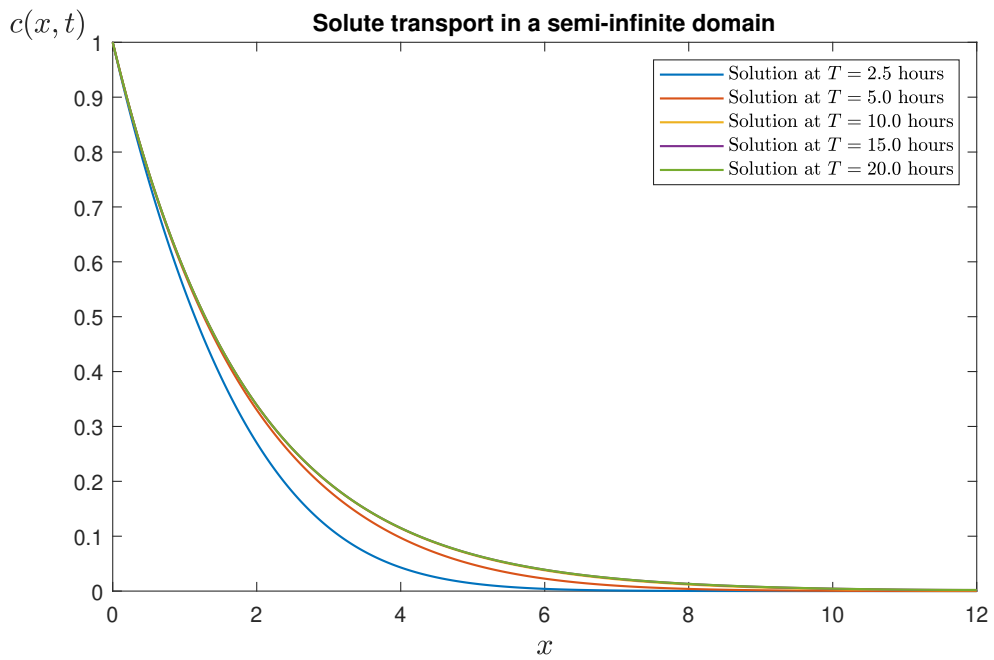


Figure 8: The solution of equation (32) (infinite domain) for times (hours) T - 2.5, 5, 10, 15, 20, and $\lambda = 0.5$ per hr.

4.4 Problems with a finite domain

While the infinite domain has some physical applications, many common examples have known distances as the domain. One such example is that of a septic tank drain field. In this scenario, the net rate of change of solute mass within a volume of porous media is equal to the difference between the flux of solute in and out of the volume adjusted for the loss or gain of the solute mass because of chemical reactions. The flux is in controlled by both advection and dispersion, while the decay and degradation are controlled by the chemical reactions the contaminant has as it travels through the medium. A key observation is that the distance between where the septic field lies and the water table is a known quantity and is close enough to have an effect on the magnitude of the concentrations [20].

In the case of the finite domain, the analytical solutions are not compact as in the previous case. Consider the partial differential equation defined in (32) with initial condition:

$$c(x, t = 0) = 0$$

subject to boundary conditions:

$$c(x = 0, t) = c_o \quad \text{and} \quad \frac{\partial c}{\partial x}(x = L, t) = 0$$

The solution was solved analytically by Van Genuchten and Alves [11] and modified by Wexler [20]. It is expressed as:

$$C(x, t) = C_0 \left\{ \frac{\exp \left[\frac{(V - U)x}{2D} \right] + \frac{(U - V)}{(U + V)} \exp \left[\left(\frac{V + U}{2D} \right) x - \frac{UL}{D} \right]}{\left[1 + \frac{(U - V)}{(U + V)} \exp \left(\frac{-UL}{D} \right) \right]} - 2 \exp \left[\frac{Vx}{2D} - \lambda t - \frac{V^2 t}{4D} \right] \sum_{i=1}^{\infty} \frac{\beta_i \sin \left(\frac{\beta_i x}{L} \right) \left[\beta_i^2 + \left(\frac{VL}{2D} \right)^2 \right] \exp \left[-\frac{\beta_i^2 Dt}{L^2} \right]}{\left[\beta^2 + \left(\frac{VL}{2D} \right)^2 + \frac{VL}{2D} \right] \left[\beta_i^2 + \left(\frac{VL}{2D} \right)^2 + \frac{\lambda L^2}{D} \right]} \right\} \quad (37)$$

where $U = \sqrt{V^2 + 4\lambda D}$ and β_i are the roots of the equation

$$\beta \cot \beta + \frac{VL}{2D} = 0. \quad (38)$$

This equation approximates the solution to (32) on a finite domain. However, it is an approximation where the roots β_i are also approximated. The roots of equation (38) were first given in Carslaw and Jaeger in 1959 [3] for various values of the constant $VL/2D$. To find more roots for the equation, we could use a non-linear root-finding algorithm such as Newton's method. We propose that we use the RBF-FD method to approximate these solutions.

Discretizing the time steps requires a similar approach to the infinite domain problem with a small change. In the Backward Euler approach, \tilde{D} represents the discretization of equation (32) rather than the transformed equation (34). This leads to a similar result:

$$\mathbf{D}c^{i+1} = c^i$$

Using the same parameters that Wexler used, Figure 8 was produced. These parameters are: $v = 0.6$ in/hr, $D = 0.6$ in²/hr, $L = 12$ in, $\lambda = 0.0$ per hr, and $c_0 = 1.0$ mg/L. We also assumed that the fluid is of constant density and viscosity, the solute may be subject to first-order chemical transformations, the flow is in the x-direction only with constant velocity, and the longitudinal dispersion coefficient D is constant. [20].

The results from Figure 9 are very similar to the infinite domain case when compared to Figure 6). We could think that the infinite domain functions as a limiting solution of the finite domain case $L \rightarrow \infty$. Solving for a finite domain is essentially the same as solving for the much nicer infinite domain; however, the finite domain PDE does not require any transformation to solve.

Another interesting test case to look at is when the equation (32) has the Cauchy bound-

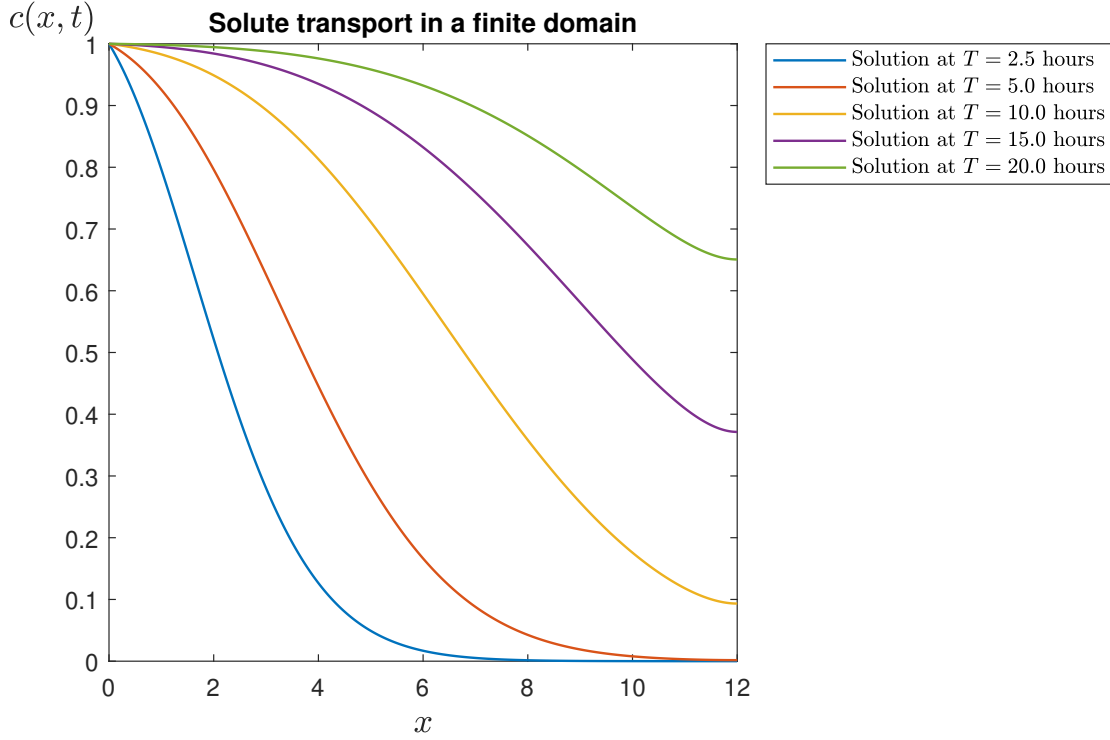


Figure 9: The solution of equation (32) (finite domain) for times (hours) T - 2.5, 5, 10, 15, and 20.

any conditions. Physically, this depicts that the flux across a boundary is proportional to the difference between concentration values on the two opposite sides of the boundary. In other words, when the inflow is a mixed solute, it is transported through the domain by advection and dispersion [20]. Consider the partial differential equation defined in (32) with initial condition:

$$c(x, t = 0) = 0$$

Subjected to boundary conditions:

$$vc_o = vc - D \frac{\partial c}{\partial x} \quad \text{and} \quad \frac{\partial c}{\partial x}(x = L, t) = 0 \quad (39)$$

and similar parameters and assumptions from the infinite and finite domain problems [20].

The approximations for this PDE for times (hours) T - 2.5, 5, 10, 15, and 20 are shown in figure 10. By observing Figure 9, we could clearly see the effect of the new boundary condition near $x = 0$.

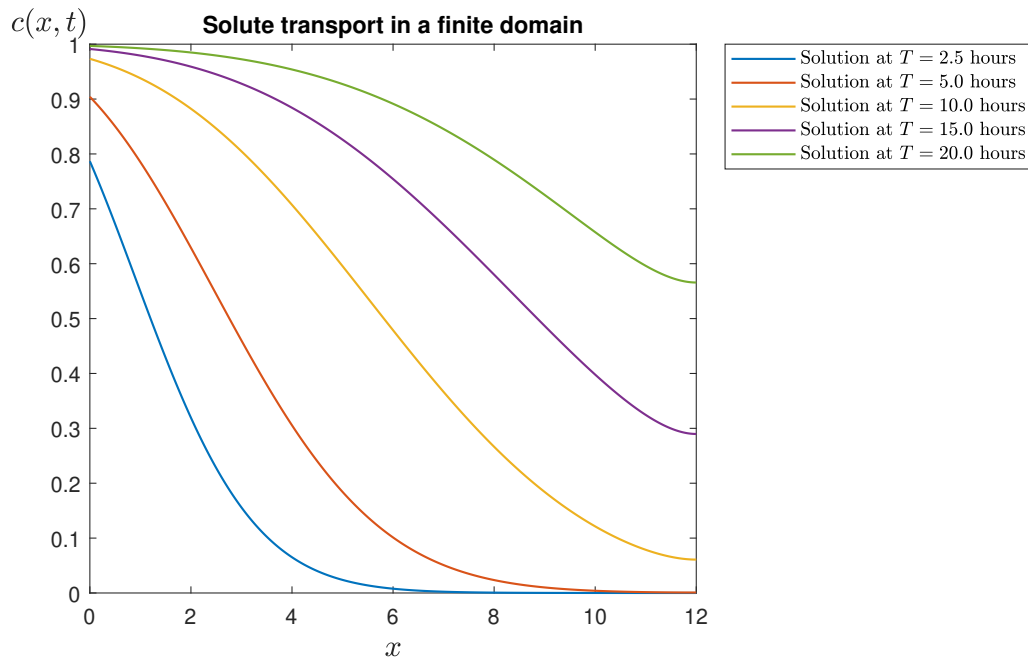


Figure 10: The solution of equation (32) (finite domain) with Cauchy type boundary conditions for times (hours) T - 2.5, 5, 10, 15, and 20.

The results of Figure 10 show that the RBF-FD method can handle cases with Cauchy boundary conditions. The major difference is how the first and last rows are represented in \mathbf{D} from equation (36). These rows were replaced respectively by:

$$\left[v - D \cdot D_x \quad -D \cdot D_x \quad \dots \quad -D \cdot D_x \right]_{1 \times N} \quad \text{and} \quad \left[D_x \quad D_x \quad \dots \quad D_x \right]_{1 \times N}$$

The term D_x represents the terms from the differentiation matrix for the first derivative, and D represents the diffusion constant. Similarly, the first and last entries of the right-hand side of the equation (36) were replaced with appropriate boundary concentration values from (39).

4.5 Conclusion

In this chapter, we introduced the concept of degradation and decay and then incorporated it into the transport equation (25). The RBF-FD method was then used to solve the new equation over a semi-infinite and finite domain numerically. The results showed that the infinite domain could be considered as a limiting case to the finite domain case. We also solved the equation using Dirichlet, Neumann, and Cauchy boundary conditions. Implementing these boundary conditions into the RBF-FD method only requires the changing of two rows of a matrix. The equation was solved for different λ values, with results showing that the change in concentration happens faster over time with a higher λ value.

5 Overview

Solving the one-dimensional transport equation (25) analytically can be simple, but when trying to solve it in higher dimensions, the analytical solution may not exist or is hard to solve. This thesis explores a numerical approach to solving the transport equation for one dimension. However, the derivation in chapter 2 and the methods used in chapters 3 and 4 can be easily adapted to higher dimensions. This alleviates the complexity that solving the equation analytically brings. In addition, the MATLAB program we wrote to approximate the solutions of the transport equation can be easily modified to accommodate various boundary conditions.

The work in the previous chapters shows that the RBF-FD method is a useful numerical tool for solving complex PDEs such as the transport equation. After a brief exploration into the history behind the RBF-FD method and deriving the transport equation (25), the RBF-FD method was applied to the basic transport equation. In this application, a transformation using the hyperbolic tangent function to map the semi-infinite domain to a finite one was introduced. The need to do a transformation arose due to the infinite domain specified by the boundary conditions. In the following convergence analysis, it was shown that if p is the degree of the appended polynomial to the RBF, the order of convergence is of $O(h^{p-2})$, where h is the fill distance.

In Chapter 4, the concept of degradation and decay was introduced into the transport equation. Using the work provided by Wexler [20], the RBF-FD method was used in the approach to solving the transport equation with linear decay and degradation present in both infinite and finite domains. The infinite domain problem was found to be relatively similar to

that of the problem discussed in Chapter 3. In addition, the new transport equation (32) was solved for problems using various boundary conditions to show that the RBF-FD method can easily handle any type of boundary condition and produce accurate results.

There are limitations to the approach presented in this thesis. The first limitation is dealing with non-linear equations. RBF-FD method can be used to solve non-linear PDEs but requires methodologies to handle non-linearity. Another limitation can be shown through the Burger equation. This equation is similar to the transport equation, however the term v in equation 25) is not constant but replaced by the term c . While not impossible to implement the RBF-FD method for these types of equations, it falls outside of the scope of this thesis. It is an avenue worth pursuing for future research.

There are also several other avenues that were not explored in this thesis. A major concept tied to transport is that of sorption. In chapter 2; during the derivation of the transport equation; advection, diffusion, and dispersion could be formulated separately for both fluid and solid phases. However, the interaction between fluids and solids is an important process present in many environmental systems [5]. Sorption can be broadly categorized by the speed of the process into general types: fast and slow sorption [5]. While slow sorption can be solved relatively easily using the methods discussed in chapter 4, fast sorption requires the solving of a system of PDEs that govern transport for both fluids and solids. This requires some modification of the RBF-FD method and solving a coupled system of PDEs. Due to the time constraints, this fell outside the scope of the project and is something to look into for future work.

References

- [1] Kendall E. Atkinson, *Interpolation Theory*, Introduction to numerical analysis second edition, 1989, pp. 131-185.
- [2] Martin D. Buhmann, *Radial basis functions: theory and implementations*, Cambridge University Press, 2004.
- [3] H. S. Carslaw and J.C. Jaeger, *Conduction of heat in solids second edition*, Oxford at the Clarendon Press, 1959.
- [4] T.A. Driscoll and B. Fornberg, *Interpolation in the limit of increasingly flat radial basis functions*, Computers and Mathematics with Applications **43** (2002), 413-422.
vskip.2in
- [5] Holzbecher Ekkehard, *Environmental modeling using MATLAB*, Springer Berlin, Heidelberg, 2012.
- [6] Gregory E Fasshauer, *Meshfree approximation methods with matlab* (2007).
- [7] Bengt Fornberg and Natasha Flyer, *A primer on radial basis functions with applications to the geosciences*, Society for Industrial and Applied Mathematics, 2015.
- [8] ———, *Fast generation of 2-d node distributions for mesh-free pde discretizations*, Computers and Mathematics with Applications **69** (2015), 531-544.
- [9] B. Fornberg and G. Wright, *Stable computation of multiquadric interpolants for all values of the shape parameter*, Computers and Mathematics with Applications **48** (2004), 853-867.
- [10] B. Fornberg and J. Zuev, *The runge phenomenon and spatially variable shape parameters in RBF interpolation*, Computers and Mathematics with Applications **54** (2007), 379-398.
- [11] M.T. van Genuchten and W.J. Alves, *Analytical solutions of the one-dimensional convective-dispersive solute transport equation*, U.S. Department of Agriculture **Technical Bulletin no. 1661** (1982).
- [12] Gene H. Golub and Charles F. van Loan, *Vandermonde Systems and the FFT*, Matrix computations third edition, 1996, pp. 183-192.
- [13] Armin Iske, *Radial Basis Functions*, In: multiresolution methods in scattered data modelling, 2004, pp. 31-65.

- [14] Nadun Kulasekera Mudiyansele, *New numerical approximations of geological processes in heterogeneous systems using radial Basis Functions*, Doctorate Dissertation, Michigan Technological University, 2021.
- [15] C.A. Micchelli, *Interpolation of scattered data: distance matrices and conditionally positive definite functions*, *Constructive Approximation* **2** (1986), 11-22.
- [16] Lloyd N. Trefethen, *Spectral methods in MATLAB*, Society for Industrial and Applied Mathematics, 2000.
- [17] Akio Ogata and R.B. Banks, *A solution of the differential equation of longitudinal dispersion in porous media*, U.S. Geological Survey Professional Paper **411-A** (1961), A1-A9.
- [18] Peter J. Olver, *Introduction to partial differential equations*, Springer Cham, 2014.
- [19] Holger Wendland, *Scattered data approximation*, Cambridge Monographs on Applied and Computational Mathematics **17** (2005).
- [20] Eliezar J. Wexler, *Analytical solutions for one-, two-, and three-dimensional solute transport in groundwater systems with uniform flow*, U.S. Geological Survey **Open-File Report 89-56** (1989).
- [21] Wolfram, *Initial and boundary value problems* (2010), available at <https://reference.wolfram.com/language/tutorial/DSolveInitialAndBoundaryValueProblems.html>.

Vita

David Marhao completed his Master of Arts Mathematics with a College teaching concentration at Appalachian State University in 2023. David Marhao obtained his bachelor of Science in Mathematics with a Life Science concentration as well as a minor in Biology from the same university. He will be pursuing a teaching career after graduation, teaching math classes at either the community college or university level. He is also considering pursuing a Ph.D. in mathematics.