University of Nevada, Reno

A fast characteristic finite difference method for fractional advection-diffusion equations with non-linear reaction.

A thesis submitted in partial fulfillment of the
requirements for a degree of Master of Science in
Mathematics

By
Anthony LaFleur
Dr. Aleksey Telyakovskiy/ Thesis Advisor
August 2014

THE GRADUATE SCHOOL

We recommend that the thesis
prepared under our supervision by

ANTHONY LAFLEUR

entitled

A fast characteristic finite difference method for fractional advection-diffusion equations with non-linear reaction.

be accepted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

Aleksey S. Telyakovskiy, Ph.D., Advisor

Jeff W. Mortensen, Ph.D., Committee Member

Clay Cooper, Ph.D., Graduate School Representative

David W. Zeh, Ph.D., Dean, Graduate School

August, 2014

Abstract

Contaminant transport in porous media can be modeled with fractional differential equations. This approach results in early arrival of contaminants and heavy-tail distributions observed in field experiments. The implicit finite difference scheme with the shifted Grúnwald approximation discritizing the fractional advection-diffusion equation unconditionally stable. We add an additional non-linear, Lipschitz continuous term to account for reactions and we solve the advection-diffusion equation utilizing fast Toeplitz matrix-vector multiplication. We then extend the method to the two-dimensional case. Numerical results are provided to compare performance of the methods proposed.

Contents

1 Introduction **1**

1.1 History 1

1.2 Fractional Derivative Formulas 1

1.3 Numerical Solutions 2

1.4 Multidimensional Methods 5

1.5 Non-linear Systems 5

2 Preliminaries **6**

2.1 Lipschitz Continuity 6

2.2 Newton - Kantorovich 7

2.2.1 Newton's Method 7

2.2.2 Convergence of Newton's Method 8

3 Derivation of Scheme **9**

3.1 Shifted Grúnwald 10

3.2 Method of Characteristics 14

3.3 Finite Difference Scheme 18

4 Convergence of the Finite Difference Schemes **21**

4.1 Convergence of Implicit Scheme 21

4.1.1 Stability 22

4.1.2 Consistency 25

4.1.3 Convergence 26

4.2 Convergence of Explicit Scheme 29

4.2.1 Stability 29

4.2.2 Consistency 31

4.2.3 Convergence 32

5 Newton's Method **34**

5.1 Derivation of System 34
5.2 Convergence of Newton's Method 35

6 Fast Iterative Algorithm **38**
6.1 Explicit Algorithm 38
6.2 Implicit Algorithm 41

7 Extension to Two Spatial Dimensions **42**
7.1 Extension of the Method of Characteristics 43
7.2 Extension of Shifted Grúnwald Approximation 45
7.3 Two-Dimensional Finite Difference Schemes 46
7.3.1 Two-Dimensional Explicit Finite Difference Scheme 47
7.3.2 Two-Dimensional Implicit Finite Difference Scheme 47

8 Two-Dimensional ADI Methods **48**
8.1 ADI Method for the Explicit Finite Difference Scheme 49
8.1.1 Explicit ADI Method 49
8.1.2 Stability of Explicit ADI Method 50
8.1.3 Consistency of Explicit ADI Method 52
8.2 Two-Dimensional Implicit Finite Difference Scheme 54
8.2.1 Implicit ADI Method 54
8.2.2 Stability of Implicit ADI Method 56
8.2.3 Consistency of Implicit ADI Method 59

9 Numerical Experiments **59**
9.1 Slow Diffusion, Slow Reaction 60
9.2 Fast Diffusion, Fast Reaction 62
9.3 Two-Dimensional Numerical Experiment 64

10 Conclusion **66**

List of Figures

1 Convergence of Newton's Method 9
2 Grúnwald Weights 12
3 Characteristic Lines 15
4 2-D Characteristic Trace 44
5 Slow Diffusion, Slow Reaction Comparison Plot 61
6 Fast Diffusion, Fast Reaction Comparison Plot 63
7 Two Dimensional Numerical Solution 65

List of Tables

1 Example one, Slow Diffusion, Slow Reaction Comparison 60
2 Example one Explicit Error for 1024 spatial steps 62
3 Example one Implicit Error for 1024 spatial steps 62
4 Example Two, High Diffusion, High Reaction Comparison 63
5 Example two Explicit Error for 1024 spatial steps 64
6 Example two Implicit Error for 1024 spatial steps 64
7 Two Dimensional comparison with 128 spatial steps in x and y 66

1 Introduction

When discussing fractional derivatives, a question often asked is what is a fractional derivative? Most are familiar with the integer order differentiation taught in undergraduate calculus and there is often a clear sense of what the derivative represents in the physical sense; like velocity or acceleration of a particle. Therefore the concept of the fractional order seems a little alien. But in fact the fractional derivative is simply the generalization of the more common integer-order differentiation and has a lot of the same properties. Actual fractional order is a bit misleading since the order is not strictly limited to fractions and in fact can be any real number, or complex for that matter [4]. We will briefly discuss the history of fractional derivatives and provide a brief description of their application.

1.1 History

Fractional calculus is nearly as old as its more familiar integer counter part. In 1695 L'Hopital, replying to a letter from Leibniz, asked the question "What if the order will be 1/2?", referring to Leibniz's notation of $\frac{d^n}{dx^n}$. The question posed would not be answered by Leibniz, but would be a topic for other mathematicians including Euler, Liouville, Fourier, Abel, Riemann and many others [4, 20].

Even though the idea of fractional calculus is nearly as old as the integer-order forms, it has remained a relatively abstract mathematical area and until only recently, in the past 50 years or so, has it been shown to capture the characteristics of natural phenomena [20]. Some of these applications include, but not limited to, viscoelasticity of materials [4, 25], flow through porous media (or more generally Lévy motion) [2], heat transfer [17], hydrology [1], and many others.

1.2 Fractional Derivative Formulas

Unlike integer order differentiation, there are several forms for the fractional derivative. We will discuss a few of these to get a general familiarization. The most common fractional

derivative is the Riemann-Liouville form given as

$$D^{\alpha}_{RL}u(x) = \frac{1}{\Gamma(n-\alpha)}\frac{d^n}{dx^n}\int_a^x u(\xi)(x-\xi)^{n-\alpha-1}d\xi \tag{1}$$

for $x \in [a,b]$, where α is the differential order, $\Gamma(\cdot)$ the Gamma function, $n-1 < \alpha < n$ and $n = \lfloor \alpha \rfloor + 1$. The function, $u(x)$ must be $n-1$ times differentiable on $[a,b]$ and the $n-th$ derivative is integrable on $[a,b]$.

A common technique to solve fractional differential equations involves the use of the Laplace transform, which in the case of the Riemann-Liouville form, produces boundary conditions containing the limit values of the derivative at the lower terminal value, $x = a$. While may be mathematically solvable, there is no physical meaning [18]. In addition, the Riemann-Liouville fractional derivative of a constant is not zero. This leads us to an alternate form of the fractional derivative called the fractional derivative of Caputo type.

$$D^{\alpha}_{C}u(x) = \frac{1}{\Gamma(n-\alpha)}\int_a^x \frac{d^n u(\xi)}{d\xi^n}(x-\xi)^{n-\alpha-1}d\xi \tag{2}$$

where $n-1 < \alpha < n$ and $n = \lfloor \alpha \rfloor + 1$. The difference between Equation (1) and Equation (2) is the order of differentiation and integration. So while the Riemann-Liouville form requires differentiability $n-1$ times, the Caputo form will require differentiability n times. But there are some advantages to using the Caputo derivative, namely the fractional derivative of a constant is zero, and the boundary conditions of the Laplace transform impose boundary conditions involving integer order derivatives [18, 4].

1.3 Numerical Solutions

A large body of work has been done on the numerical solutions to fractional partial differential equations. However, we will be working with a specific equation, so we will present an overview related to this equation.

The fractional derivative given in Equation (1) has a discrete representation, called the Grúnwald-Letnikov formula, which will be discussed in greater detail later. This discretization often produces numerically unstable methods. For example in [11] both the explicit and implicit Euler methods were shown to be unconditionally unstable when using the Grúnwald-Letnikov approximation for the fractional derivative. However, [11] proposed a shifted approximation that was shown to be first order and stable.

Utilizing this first order approximation, Wang and Wang [24], demonstrated a method for solving an advection-dominated diffusion equation with fractional order space derivatives. This ADE modeled containment transport [2]. This model is based on Levy motion of particles in the media. This means that particles have a probability of "long walks," or moving much further in a given time unit than a Brownian model would predict. The fractional order was shown to model this phenomena.

An example of Brownian motion and the Fickian diffusion model is heat transfer through a solid material, i.e. conduction. This model is based on molecules transferring energy through collision to their neighbors, so the average distance that energy is transfered in a given unit of time can be modeled probabilistically and has a normal distribution. This leads to a finite variance in the distance that energy is transfered. An example of Levy motion would be flow through a porous media. Motion through the media is based on the pores, and if there is good alignment of the pores then a particle, say a contaminant, can travel much further than predicted by Brownian motion. This long excursion is called a long walk, and the variance is not necessarily finite [2].

An issue with the use of fractional order derivatives has been their non-local nature. The fact that the fractional order derivatives approximate the diffusion characteristics in [2] makes them computationally difficult to solve. The non-local nature of the fractional derivative implies that it will use points from the entire computational domain to compute the derivative at a single point. This generates dense matrices instead of the diagonal matrices generated

by similar integer-order partial differential equations.

However, Wang and Wang in [24] obtained some ways to increase the performance of these fractional partial differential equation models. In their work, Wang and Wang utilized a method of characteristics derived by Douglas and Russell, [7]. This method was developed for convection dominated diffusion problems, combining adapting the method of characteristics typically used in parabolic problems. The use of this method reduces the differential equation from an convection-diffusion equation to a diffusion equation along the characteristic curves, reducing the size of the problem by requiring fewer time steps.

In addition, Wang and Wang employed a novel way of multiplying the coefficient matrices generated by the fractional derivatives. They were able to reduce the matrix multiplication from a naive computational complexity of $O(n^2)$ to $O(n \lg(n))$. This matrix multiplication was used in the conjugate gradient method to solve the linear system generated by the finite difference scheme. The conjugate gradient method relies on a small condition number of the matrix being solved for reasonable convergence. Since the equation that Wang and Wang solved typically had very small diffusion constants, this was not a problem [24].

To overcome this restriction on the size of the diffusion constant, a couple of methods have been proposed. Lei and Sun proposed a circulant preconditioner in [9] to reduce the condition number of the coefficient matrix. Other methods have attempted to get away from the conjugate gradient method. Pang and Sun proposed a multigrid method in [16] to solve the linear system given by the finite difference scheme. However, this method has not been proven to converge. In [14], Moroney and Yang give a method using a Poisson preconditioner and backward differentiation that allows for the use of current ODE solver technologies, enabling the use of existing powerful ODE software packages.

1.4 Multidimensional Methods

All of the methods and papers discussed so far have been for single dimensional methods. To extend the current methods to multiple spatial dimensions we must first define a difference scheme for multiple spatial dimensions. In [13], Meerschaert et. al. developed a multi-dimensional method based on an extension of the shifted Grúnwald in multi-dimensions. They also adapted standard ADI methods for integer order partial differential equations to the fractional dispersion equation.

Utilizing the multi-dimensional shifted Grúnwald approximation in [13], Wang and Du in [23] develop an ADI method for the three-dimensional space-fractional diffusion equation and propose two separate ways to solve the resulting linear system.

1.5 Non-linear Systems

Most of the work done with regards to the fractional advection-diffusion equation has been focused on the linear 1-dimensional case with a few methods looking at various non-linear cases. We are not aware of any work that has been done with regards to the multi-dimensional space fractional advection-diffusion equation.

In this paper we will present first the one-dimensional case for the fractional advection-diffusion equation with non-linear reaction using the method of characteristics and fast matrix multiplication proposed by Wang and Wang. We will develop two methods for solving the resulting non-linear system and compare numerical results. We will then extend this method to two-dimensions based on the work in [13, 23] using the methods in [7], extended into multiple spatial dimensions and propose methods to solve the resulting finite difference scheme more efficiently.

2 Preliminaries

In this section we will present some of the preliminary ideas required to construct and conduct analysis of the numerical schemes. We will need Lipschitz continuity, which we will demand from the non-linear reaction term. We will also need results developed for iterative numerical solutions for non-linear systems.

2.1 Lipschitz Continuity

In order to reduce the class of non-linear functions we consider the Lipschitz continuous functions. First we will define Lipschitz continuity for general normed vector spaces [26].

Definition 2.1 (Lipschitz Continuity)**.** *Let (X, d_X) and (Y, d_Y) be normed vector spaces and $S : X \to Y$. S is said to be Lipschitz continuous if for some K we have $d_Y(S(x), S(y)) \leq K d_X(x, y)$ for all $x, y \in X$. K is called the Lipschitz constant for S.*

For the purpose of the work presented here we will consider functions of the following form $S : \mathbb{R}^3 \to \mathbb{R}$ and Lipschitz continuity will only need to be imposed in a single dimension of $\mathbb{R}^3$. We will be working with reaction terms of the form $S(u(x,t), x, t)$, and we will require Lipschitz continuity of the $u(x,t)$ argument only, so now we present a refined definition for the Lipschitz condition here [8].

Definition 2.2 (Lipschitz Condition)**.** *Let $f : \mathbb{R}^3 \to \mathbb{R}$. If f is continuous on $a \leq y \leq b$, $c \leq z \leq d$, $-\infty < x < \infty$ and satisfies the following inequality*

$$|f(x_1, y, z) - f(x_2, y, z)| \leq K|x_1 - x_2|$$

then f is Lipschitz continuous with respect to x and this inequality is known as the Lipschitz condition.

We will impose Lipschitz Continuity on the reaction term, $S(u(x,t), x, t)$, with respect to $u(x,t)$ and we will denote K as the associated Lipschitz Constant.

2.2 Newton - Kantorovich

In developing an implicit scheme we will derive a system of non-linear algebraic equations. This will require a non-linear solver. We will employ Newton's method to solve this system since it will enable us to adapt the linear methods used in the explicit scheme.

2.2.1 Newton's Method

Newton's method is a widely known method for solving nonlinear equations. We present it first for a single equation and then for a system of equations. When solving for the zero of a real valued function of a real valued variable the method is often called the Newton-Raphson iteration [8]. Newton's method starts with an initial guess of the solution x_0, then defines recursively the iteration as [8]

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

This can be expanded to more general functions over Banach Spaces [15].

Definition 2.3 (Newton's Method for Systems)**.** *Let X and Y be Banach spaces, $F : D \subset X \to Y$ and suppose that on an open convex set $D_0 \subset D$, F is Frechet differentiable and is Lipschitz Continuous by Definition 2.1, then the Newton Iterates are*

$$x_{n+1} = x_n - [Jx_n]^{-1} F x_n \tag{3}$$

where J denotes the Jacobian of F.

Computing the iterates as defined in Equation (3) involves finding the Jacobian, computing its inverse then multiplying by F. This procedure can get quite computationally expensive. It is preferred to solve the following linear system instead of Equation (3) [8]

$$\begin{aligned} [Jx_n]\Delta x_n &= -Fx_n \\ x_{n+1} &= x_n + \Delta x_n \end{aligned} \tag{4}$$

2.2.2 Convergence of Newton's Method

In order to guarantee the convergence of the implicit numerical scheme, we must ensure that the initial guess for the Newton iteration is within its radius of convergence. In [8] a brief argument is given for the convergence of the Newton-Raphson iteration. We will present the Newton-Kantorovich theorem from [15] to facilitate our exposition of the convergence criteria for the implicit numerical scheme developed later in this paper.

Theorem 2.1 (Newton-Kantorovich)**.** *Let X and Y be Banach Spaces and $F : D \subset X \to Y$. Suppose that on an open convex set $D_0 \subset D$, F is Frechet differentiable and satisfies the Lipschitz Continuity of Definition 2.1. For all $x_0 \in D_0$, assume that $J_0^{-1} = [F'(x_0)]^{-1}$ is defined on Y and that $h = \beta K \eta \leq \frac{1}{2}$ where $\|J_0^{-1}\| \leq \beta$ and $\|J_0^{-1} F x_0\| \leq \eta$. Set*

$$t^* = \frac{1}{\beta K}(1 - \sqrt{1 - 2h})$$

$$t^{**} = \frac{1}{\beta K}(1 + \sqrt{1 - 2h})$$

and suppose that $A = \{x | \|x - x_0\| \leq t^*\} \subset D_0$. Then the Newton iterates in Definition 2.3 are well defined, lie in A and converge to a solution x^* of $Fx = 0$ which is unique in $D_0 \bigcap \{x | \|x_0 - x\| < t^{**}\}$. Moreover, if $h < \frac{1}{2}$ the order of convergence is at least quadratic.

In Figure 1 we see an example of the sets described in Theorem 2.1 in one dimension. If the initial guess, x_0 is contained within the set A, then the Newton iterates will converge to the zero of the system. From Theorem 2.1 we get the following corollary for the convergence of the Jacobian System in Equation (4) and it is motivated by work in [15]

Corollary 2.1. *Let X and Y be Banach spaces and $F : D \subset X \to Y$ and $D_0 \subset D$ be an open and convex set such that F is Lipschitz Continuous on D_0. If*

$$K||J_0^{-1}|| \cdot ||J_0 F(x_0)|| \leq \frac{1}{2} \tag{5}$$

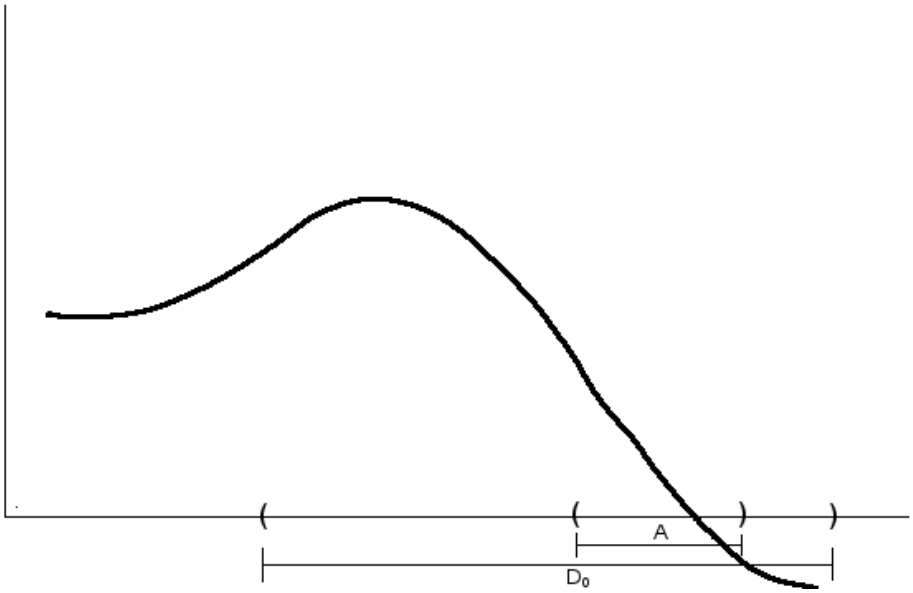

Figure 1: Example of the sets D_0 and A for some function

where K is the Lipschitz constant, then the Newton Iterates defined by the Jacobian System

$$[Jx_n]\Delta x_n = -Fx_n$$
$$x_{n+1} = x_n + \Delta x_n$$

are well defined and converge to a solution x^*.

3 Derivation of Scheme

We consider a fractional advection-diffusion equation with non-linear reaction term

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(V(x,t)u - D_+(x,t)\frac{\partial^{\alpha-1}u}{\partial_+ x^{\alpha-1}} - D_-(x,t)\frac{\partial^{\alpha-1}u}{\partial_- x^{\alpha-1}}\right) = S(u,x,t) + f(x,t) \qquad (6)$$

where $V(x,t)$ is the mean advective velocity, $1 < \alpha < 2$, $D_+(x,t) = (1+\beta)D(x,t)/2$ and $D_-(x,t) = (1-\beta)D(x,t)/2$ with $0 < D_{min} \leq D(x,t) \leq D_{max} < +\infty$ being the diffusion

coefficients, $-1 \leq \beta \leq 1$ indicates the relative weight of forward versus backward transition probability: $f(x,t)$ is the sink-source term and $S(u,x,t)$ is a non-linear reaction term.

3.1 Shifted Grúnwald

Fractional order derivatives are present in Equation (6), and before we can develop an effective numerical solution, we must first develop some basic theory of numerical solution of equations with space-fractional derivatives. It should be noted that Equation (6) uses a Riemann-Liouville fractional derivative of order α, defined as follows [4, 11, 12]

$$\frac{d^\alpha f(r)}{dr^\alpha} = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dr^n} \int_L^r \frac{f(\xi)}{(r-\xi)^{\alpha+1-n}} \tag{7}$$

where n is an integer such that $n-1 < \alpha \leq n$, and over the finite domain $L < r < R$. In the case of $L = -\infty$, this is the Liouville form for the fractional derivative. In [11], Meerschaert and Tadjeran propose some methods for approximating (7). The first of these is the standard Grúnwald formula [4, 11]

$$\frac{d^\alpha c(r)}{dr^\alpha} = \frac{1}{\Gamma(-\alpha)} \lim_{M\to\infty} \frac{1}{h^\alpha} \sum_{k=0}^{M} \frac{\Gamma(k-\alpha)}{\Gamma(k+1)} c(r-kh)$$

where $c(r) \in C[L,R]$, $L, R \in \mathbb{R}$ and $L < r < R$, M is a positive integer, $h = (r-L)/M$ and $\Gamma(-)$ is the Gamma function. Propositions 2.1 and 2.3 of [11] establishes that this approximation, when used in the explicit or implicit Euler methods, is unstable. To remedy this, Meerschaert and Tadjeran proposed a shifted Grúnwald approximation to the fractional derivative given as [11]

$$\delta_\alpha f(x) = \frac{1}{\Gamma(-\alpha)} \frac{1}{h^\alpha} \sum_{k=0}^{\infty} \frac{\Gamma(k-\alpha)}{\Gamma(k+1)} f(x-(k-p)h) \tag{8}$$

where p is a non-negative integer and $L = -\infty$. The following Lemma is motivated by the

work in [11].

Lemma 3.1. *Let $Af(x) = d^\alpha f(x)/dx^\alpha$ be the Riemann-Liouville fractional derivative then the approximation $\delta_\alpha f(x)$ converges uniformly and has a truncation error of $O(h)$, namely*

$$\delta_\alpha f(x) = Af(x) + O(h)$$

Proof. See [11], Theorem 2.4. □

Meerschaert and Tadjeran, in [11], show that the error is also a function of the shift, p, and the smaller the shift the smaller the error. Thus Meerschaert and Tadjeran use a shift value of 1, we will also use a shift value of 1, shifting the approximation by 1 grid point, for our approximations [11, 12, 24].

Note that Equation (6) utilizes both left and right hand derivatives. The left hand and right hand derivatives are Riemann-Liouville fractional derivatives of order α [12, 4]. Equation (7) is for a left-sided derivative, and the derivatives used in Equation (6) are defined as

$$\frac{d^\alpha f(x)}{d_+ x^\alpha} = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dx^n} \int_L^x \frac{f(\xi)}{(x-\xi)^{\alpha+1-n}} d\xi \tag{9}$$

and

$$\frac{d^\alpha f(x)}{d_- x^\alpha} = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dx^n} \int_x^R \frac{f(\xi)}{(\xi-x)^{\alpha+1-n}} d\xi \tag{10}$$

on a finite domain $L < x < R$, with the case $1 \leq \alpha \leq 2$ and n is an integer such that $n-1 < \alpha \leq n$ [12]. Taking the finite sums in Equation (8) with $p = 1$ and substituting them in Equations (9) and (10) with $p = 1$ for a finite domain we'll have the following approximations for the left and right hand space-fractional derivatives [12] :

$$\frac{d^\alpha f(x)}{d_+ x^\alpha} \approx \frac{1}{\Gamma(-\alpha)} \frac{1}{h_+^\alpha} \sum_{k=0}^{N_+} \frac{\Gamma(k-\alpha)}{\Gamma(k+1)} f(x-(k-1)h_+) \tag{11}$$

$$\frac{d^\alpha f(x)}{d_- x^\alpha} \approx \frac{1}{\Gamma(-\alpha)} \frac{1}{h_-^\alpha} \sum_{k=0}^{N_-} \frac{\Gamma(k-\alpha)}{\Gamma(k+1)} f(x+(k-1)h_-) \tag{12}$$

where N_- and N_+ are positive integers, $h_- = (x-L)/N_-$ and $h_+ = (R-x)/N_+$. Figure 2

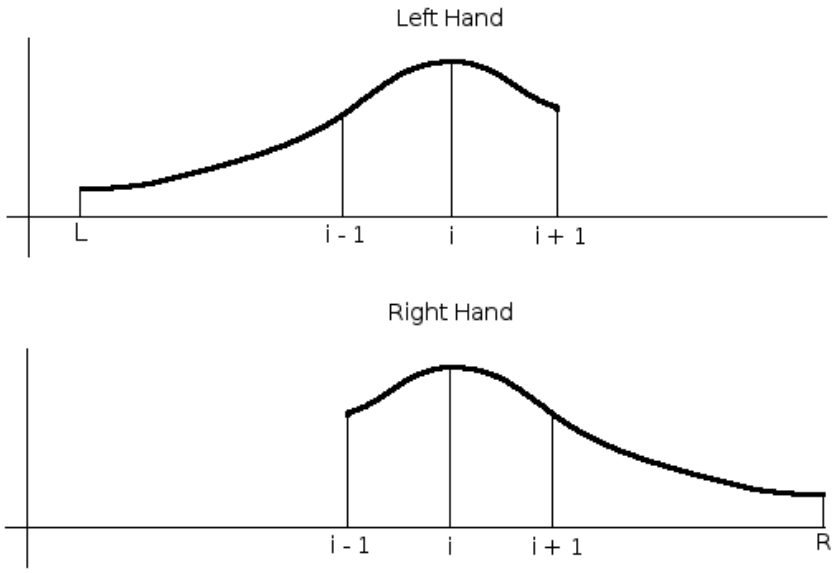

Figure 2: An example of both left hand and right hand shifted Grúnwald weights plotted along the same axis for some x-grid point i, on the interval $[L, R]$.

is an example of the shifted Grúnwald weights for a specific grid point x_i. Notice how the left-hand and the right-hand overlap by 1 index, due to the shift. The left-hand and right-hand are kind of mirrors of each other about the index i.

For ease of presentation, we will use the following notation for the left and right hand space-fractional derivatives in one dimension,

$$\delta_\alpha^+ f(x) = \frac{1}{\Gamma(-\alpha)} \frac{1}{h_+^\alpha} \sum_{k=0}^{N_+} \frac{\Gamma(k-\alpha)}{\Gamma(k+1)} f(x-(k-1)h_+) \tag{13}$$

$$\delta_{\alpha}^{-} f(x) = \frac{1}{\Gamma(-\alpha)} \frac{1}{h_{-}^{\alpha}} \sum_{k=0}^{N_{-}} \frac{\Gamma(k-\alpha)}{\Gamma(k+1)} f(x+(k-1)h_{-}) \tag{14}$$

We define the normalized Grúnwald weights as follows [11, 4]

$$g_k = \frac{\Gamma(k-\alpha)}{\Gamma(-\alpha)\Gamma(k+1)}$$

which can be written in the following way [12]

$$g_0 = 1, \quad g_k = (-1)^k \frac{(\alpha)(\alpha-1)\dots(\alpha-k+1)}{k!} \quad \text{for } k = 1, 2, 3, \dots$$

Motivated by the work in [11, 12, 4, 26, 24], we get the following lemma for the properties of the shifted Grúnwald approximation.

Lemma 3.2. *The normalized Grúnwald weights, g_k, have the following properties*

$$(1) \quad g_0 = 1, \quad g_k = (-1)^k \frac{(\alpha)(\alpha-1)\dots(\alpha-k-1)}{k!}$$

$$(2) \quad g_0 = 1, \quad g_1 = -\alpha, \quad g_k > 0 \text{ for } k = 2, 3, \dots$$

$$(3) \quad \sum_{k=0}^{\infty} g_k = 0, \text{ and } \sum_{k=0}^{n} g_k < 0, \quad \forall n \in \mathbb{N}$$

Proof. Properties (1) and (2) follow from the definition of the shifted Grúnwald weights [12, 4]. For property (3), it is well known that

$$(1+z)^{\alpha} = \sum_{k=0}^{\infty} \binom{\alpha}{k} z^k$$

for any complex $|z| \leq 1$ and any $\alpha > 0$, where

$$\binom{\alpha}{k} = \frac{(-1)^k \Gamma(k-\alpha)}{\Gamma(\alpha)\Gamma(k+1)}$$

Let $z = -1$, and we obtain the following result

$$(1 + (-1))^\alpha = \sum_{k=0}^{\infty} \frac{(-1)^k \Gamma(k-\alpha)}{\Gamma(\alpha)\Gamma(k+1)} (-1)^k$$

$$0 = \sum_{k=0}^{\infty} \frac{\Gamma(k-\alpha)}{\Gamma(\alpha)\Gamma(k+1)} = \sum_{k=0}^{\infty} g_k$$

By property (2), only g_1 is negative so any finite sum will be negative. □

3.2 Method of Characteristics

In order to develop a scheme for Equation (6) we will combine the method of characteristics with finite difference methods detailed above. The method of characteristics looks at solution behavior along characteristic lines. Consider the following equation

$$cu_y + u_x = 0 \tag{15}$$

with some initial condition, u_0. From Kincaid and Cheney [8] and Farlow [5], let a curve in the xy-plane be given as the graph of the function $y = y(x)$. Along this curve, solutions are of the form $u(x, y(x))$. For Equation (15) the following condition must be satisfied

$$0 = \frac{d}{dx} u(x, y(x)) = u_x + u_y \frac{dy}{dx}$$

so the curve we are seeking is a solution to the following differential equation

$$\frac{dy}{dx} = -\frac{u_x}{u_y}$$

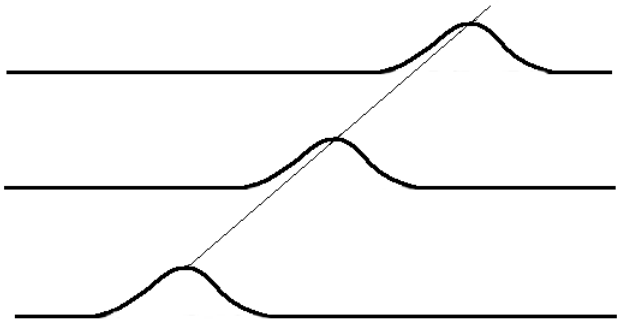

Figure 3: An example of a simple traveling wave equation at various time levels. The thin line connecting the solutions is the characteristic line.

so for Equation (15) we get the following

$$\frac{dy}{dx} = c$$

So this characteristic curve is a straight line, given by

$$y - y_0 = c(x - x_0)$$

and for this example the solution is constant along the characteristic lines, also known as the traveling wave solution, so the initial data u_0 is simply shifted along the domain based on the velocity of the wave, c. In Figure 3 multiple solutions for a few times are presented. A line depicting the characteristic curve is drawn. The solution along the curve is constant, the initial data is simply shifted along the domain according to the velocity of the wave.

The method of characteristics allows the problem for PDEs to be reduced to the problem for ODE's by examining the behavior of the function along the characteristic curves, reducing the complexity of the problem. The characteristic curves may not be lines, consider

the following [8, 5]

$$c(x,t)u_t + b(x,t)u_x = 0 \tag{16}$$

using the same procedure as used for Equation (15), we get that the velocity term

$$v = -\frac{c(x,t)}{b(x,t)}$$

and if $c(x,t), b(x,t)$ are constants then we get the same result as before, but if they are not constant then the initial data can get very distorted. For the development of our numerical method, we will want to look at the derivative in the direction of the characteristic curve. We define the directional derivative as follows (e.g. [22])

Definition 3.1. *The directional derivative of a function, $f : \mathbb{R}^n \to \mathbb{R}$, in the direction $u \in \mathbb{R}^n$ at x is defined as*

$$\nabla f \cdot \hat{u} = \lim_{h \to 0} \frac{f(x + h\hat{u})}{h}$$

where $\hat{u}$ is the unit vector in the direction of u and $x \in \mathbb{R}^n$.

For our purposes, the direction will be chosen in the direction of the characteristic curve for the given time and specified spacial position. We will now use the method of characteristics methodology (e.g.[7, 24]). For this we will consider an equation in the form

$$c(x,t)\frac{\partial u}{\partial t} + b(x,t)\frac{\partial u}{\partial x} - \frac{\partial}{\partial x}\left(a(x,t)\frac{\partial u}{\partial x}\right) = f(x,t) \tag{17}$$

Following [7] we set $\psi(x,t) = \sqrt{c^2(x,t) + b^2(x,t)}$. The time derivative and the advection terms in Equation (17) can be rewritten as [7]

$$\psi(x,t)\frac{du(x,t)}{dt} = c(x,t)\frac{\partial u(x,t)}{\partial t} + b(x,t)\frac{\partial u(x,t)}{\partial x}$$

where we define a new differential operator as

$$\frac{d}{dt} = \frac{c(x,t)}{\psi(x,y)}\frac{\partial}{\partial t} + \frac{b(x,t)}{\psi(x,y)}\frac{\partial}{\partial x}.$$

Using this, Equation (17) can be rewritten as

$$\psi(x,t)\frac{du(x,t)}{dt} - \frac{\partial}{\partial x}\left(a(x,t)\frac{\partial u}{\partial x}\right) = f(x,t) \tag{18}$$

Let N and M be positive integers, and $h = (R-L)/N$ and $\Delta t = T/M$, where h and Δt are the size of the spacial grid and the time step. Let $x_i = L + ih$ for $i = 0, 1, \ldots, N$ and $t^m = m\Delta t$ for $m = 0, 1, \ldots, M$. Now for any $x \in (L, R)$ at time step t^m, define

$$\bar{x} = x - b(x,t)(t^m - t)/c(x,t) \tag{19}$$

Then the characteristic derivative can be approximated by [7]

$$\psi(x,t)\frac{du(x,t)}{dt} \approx c(x,t)\frac{u(x,t^m) - u(\bar{x}, t^{m-1})}{\Delta t} \tag{20}$$

We should note that for the definition of $\bar{x}$, t is the time level at which $\bar{x}$ is being evaluated. So for $\bar{x}$ at the $m-1$ time level, one would use $t = t^{m-1}$. This motivates the following Lemma

Lemma 3.3. *The approximation given in Equation* (20) *approximates*

$$c(x,t)\frac{\partial u(x,t)}{\partial t} + b(x,t)\frac{\partial u(x,t)}{\partial x}$$

with a truncation error $O(\Delta t)$.

Proof. We want to show that

$$c(x,t^m)\frac{\partial u(x,t^m)}{\partial t} + b(x,t^m)\frac{\partial u(x,t^m)}{\partial x} - \frac{c(x,t^m)}{\Delta t}(u(x,t^m) - u(\bar{x},t^{m-1})) = O(\Delta t)$$

Using the definition of $\bar{x}$ given in Equation (19) with $t = t^{m-1}$, we note that $t^m - t^{m-1} = \Delta t$,

we obtain the Taylor's expansion of $u(\bar{x}, t^{m-1})$ and substitute into Equation (20) yielding

$$\frac{c(x,t^m)}{\Delta t}(u(x,t^m) - u(\bar{x},t^{m-1})) = \frac{c(x,t^m)}{\Delta t}[u(x,t^m) - u(x,t^m) + u_x(x,t^m)\frac{b(x,t^m)}{c(x,t^m)}\Delta t + u_t(x,t^m)\Delta t - u_{xx}(x,t^m)\frac{b^2(x,t^m)}{c^2(x,t^m)}(\Delta t)^2 + \ldots]$$

which reduces to

$$\begin{aligned}\frac{c(x,t^m)}{\Delta t}(u(x,t^m) - u(\bar{x},t^{m-1})) = b(x,t^m)u_x + c(x,t^m)u_t \\ -u_{xx}\frac{b^2(x,t^m)}{c(x,t^m)}\Delta t - u_{xt}b(x,t^m)\Delta t - u_{tt}\Delta t + \ldots\end{aligned} \tag{21}$$

Now subtracting Equation (21) from $c(x,t^m)\frac{\partial u(x,t^m)}{\partial t} + b(x,t^m)\frac{\partial u(x,t^m)}{\partial x}$ we get

$$c(x,t^m)\frac{\partial u(x,t^m)}{\partial t} + b(x,t^m)\frac{\partial u(x,t^m)}{\partial x} - \frac{c(x,t^m)}{\Delta t}(u(x,t^m) - u(\bar{x},t^{m-1})) = O(\Delta t)$$

as desired. □

3.3 Finite Difference Scheme

We will now develop a finite difference scheme for Equation (6), utilizing the shifted Grúnwald approximations and the method of characteristics developed previously. The divergence representation of the advection term in Equation (6) can be expanded to a non-divergence form [24, 7].

$$\frac{\partial}{\partial x}(V(x,t)u) = V(x,t)\frac{\partial u}{\partial x} + \frac{\partial V}{\partial x}u \tag{22}$$

From Wang and Wang, [24], the fractional diffusion term in Equation (6) cannot be expressed in the non-divergence form for variable diffusion coefficients $D_+(x,t)$ and $D_-(x,t)$. However, for constant coefficients V, D_-, D_+, Equation (6) reduces to the following form where

$$\frac{\partial u}{\partial t} + V\frac{\partial u}{\partial x} - D_+\frac{\partial^\alpha u}{\partial_+ x^\alpha} - D_-\frac{\partial^\alpha u}{\partial_- x^\alpha} = f(x,t) + S(u(x,t),x,t) \tag{23}$$

This equation, without the reaction term, was derived in [2], and utilized in [24]. It was shown to capture scale-dependent transport processes [2, 24]. As in [24], we will consider Equation (23) with variable coefficients. We should note this is not equivalent to Equation (6), but is a generalization of Equation (23) to variable coefficients. We will consider the following initial-boundary value problem for a transient space-fractional advection-diffusion equation in the non divergence form with an anomalous diffusion of order $1 < \alpha \leq 2$ [24].

$$\begin{aligned} &\frac{\partial u}{\partial t} + V(x,t)\frac{\partial u}{\partial x} - D_+(x,t)\frac{\partial^\alpha u}{\partial_+ x^\alpha} - D_-(x,t)\frac{\partial^\alpha u}{\partial_- x^\alpha} = f(x,t) + S(u(x,t),x,t) \\ &\qquad a < x < b, 0 < t \leq T, \\ &u(x,0) = u_0(x), \quad a \leq x \leq b, \\ &u(a,t) = 0, \quad u(b,t) = 0, \quad 0 \leq t \leq T \end{aligned} \tag{24}$$

The term $V_x u$ on the right hand side of Equation (22) is included in the reaction term $S(u(x,t),x,t)$. This equation can be viewed as the generalization of Equation (23) for variable coefficients and is equivalent for constant coefficients. We observe that the first two terms in Equation (24) have the same form as in Equation (17). So applying the same process, where $c(x) = 1$ and $b(x) = V(x,t)$, utilizing the form in Equation (18) we get the following form for Equation (24) [24]

$$\psi(x,t)\frac{du(x,t)}{dt} - D_+(x,t)\frac{\partial^\alpha u}{\partial_+ x^\alpha} - D_-(x,t)\frac{\partial^\alpha u}{\partial_- x^\alpha} = f(x,t) + S(u(x,t),x,t) \tag{25}$$

Let $u_i^m = u(x_i,t^m)$, $D_{+,i}^m = D_+(x_i,t^m)$, $D_{-,i}^m = D_-(x_i,t^m)$, $S_i^m(u_i^m) = S(u(x_i,t^m),x_i,t^m)$, $f_i^m = f(x_i,t^m)$ and $\bar{u}_i^m = u(\bar{x}_i,t^m)$. Applying the shifted Grúnwald approximation for fractional derivatives given in Equations (13) and (14) and applying the approximation for derivative along the characteristic curve given in Equation (20) to Equation (25) gives the following finite difference scheme:

$$\frac{u_i^m - \bar{u_i}^{m-1}}{\Delta t} - \frac{D_{+,i}^m}{h^\alpha}\delta_\alpha^+ u^m - \frac{D_{-,i}^m}{h^\alpha}\delta_\alpha^- u^m - S_i^m(u_i^m) = f_i^m \tag{26}$$

Letting $u^m = [u_1^m, u_2^m, \ldots, u_{N-1}^m]^T$, $\bar{u}^m = [\bar{u}_1^m, \bar{u}_2^m, \ldots, \bar{u}_{N-1}^m]^T$, $S^m = [S(u_1^m, x_1, t^m)$, $S(u_2^m, x_2, t^m), \ldots, S(u_{N-1}^m, x_{N-1}, t^m)]^T$, $f^m = [f_1^m, f_2^m, \ldots, f_{N-1}^m]^T$, $A^m = [a_{i,j}^m]_{i,j=1}^{N-1}$, and I being the identity matrix of order $N-1$, then Equation (26) can be rewritten in matrix form

$$(I + A^m)u^m - \Delta t S^m = \bar{u}^{m-1} + \Delta t f^m \tag{27}$$

where the entries of the matrix A^m are given as

$$a_{i,j}^m = -\frac{\Delta t}{h^\alpha}\begin{cases} \left(D_{+,i}^m + D_{-,i}^m\right) g_1^{(\alpha)}, & j = i \\ \left(D_{+,i}^m g_2^\alpha + D_{-,i}^m g_0^{(\alpha)}\right), & j = i-1 \\ \left(D_{+,i}^m g_0^{(\alpha)} + D_{-,i}^m g_2^{(\alpha)}\right), & j = i+1 \\ D_{+,i}^m g_{i-j+1}^{(\alpha)}, & j < i-1 \\ D_{-,i}^m g_{j-i+1}^{(\alpha)}, & j > i+1 \end{cases} \tag{28}$$

This derivation utilizes an implicit evaluation of the reaction term $S(u(x,t), x, t)$, meaning that the $u(x,t)$ is evaluated at the current time step. This produces a system of non-linear equations that we will later solve with Newton's Method. However, it is also possible to express the reaction in an explicit manner, utilizing the previous time step along the characteristic trace for the value of $u(x,t)$. We can rewrite Equation (26) in the following form

$$\frac{u_i^m - \bar{u_i}^{m-1}}{\Delta t} - \frac{D_{+,i}^m}{h^\alpha}\delta_\alpha^+ u^m - \frac{D_{-,i}^m}{h^\alpha}\delta_\alpha^- u^m - S_i^m(\bar{u}_i^{m-1}) = f_i^m \tag{29}$$

Letting $\bar{S}^{m-1} = [S(\bar{u}_1^{m-1}, x_1, t^m)$, $S(\bar{u}_2^{m-1}, x_2, t^m), \ldots, S(\bar{u}_{N-1}^{m-1}, x_{N-1}, t^m)]^T$ we get the corresponding matrix for from Equation (29)

$$(I + A^m)u^m = \Delta t \bar{S}^{m-1} + \bar{u}^{m-1} + \Delta t f^m \tag{30}$$

This is a linear system, and we will solve this using standard methods for solving linear systems. We will refer to the scheme that utilizes the implicit representation of the reaction term as the implicit scheme and the scheme that uses the explicit representation as the explicit scheme for ease of reference.

4 Convergence of the Finite Difference Schemes

In this section, we analyze the stability of the finite difference schemes. We will start with the implicit scheme. The structures of the proofs provided below will follow similar structure of the proofs in [26].

4.1 Convergence of Implicit Scheme

First we provide a definition for the convergence of a finite difference scheme based on [19, 21],

Definition 4.1 (Convergence)**.** *Let u_Δ^m be the solution of a finite difference scheme for a space fractional partial differential equation at the t^m time level with spatial grid Δ and U_Δ^m be the exact solution of the space fractional partial differential equation. The finite difference scheme is said to converge at time t if as $(m+1)\Delta t \to t$*

$$\|U_\Delta^{m+1} - u_\Delta^{m+1}\| \to 0$$

as $\Delta \to 0$ and $\Delta t \to 0$.

Now we provide a theorem regarding the convergence of numerical solutions to non-linear partial differential equations that we will use as the basis of our proofs [19, 21, 10].

Theorem 4.1 (Convergence of a Numerical Method)**.** *Let U_Δ^m be the exact solution of a well-posed time-dependent problem and let u_Δ^m be the solution of a numerical method that is*

consistent with that time dependent problem with accuracy $(\Delta^q, (\Delta t)^r)$ *where* $q, r \geq 1$*, then if the method is stable we say its solution converges with a* (q, r) *order convergence rate:*

$$\|U_\Delta^m - u_\Delta^m\| \leq C(\Delta^q + (\Delta t)^r)$$

where C *is some positive constant.*

We have excluded for now the definitions of stability and consistency since we will address them independently later. We will now prove both stability and consistency of the numerical scheme given by Equation (26).

4.1.1 Stability

In order to prove stability of the implicit scheme we will first describe stability in the sense of a non-linear partial differential equation. The following definition is taken from [19].

Definition 4.2 (Stability of a Numerical Scheme)**.** *If we consider a numerical scheme as a mapping of discrete initial data to a numerical solution , say* $(W_\Delta^0, \{G_\Delta^0\}_{t^n \in [0,T]}) \to \{W_\Delta^n\}_{t^n \in [0,T]}$ *where* W_Δ^0 *is the initial condition and* G_Δ^0 *the forcing/reaction for some grid* Δ *, then the scheme is considered stable if for all sufficiently close pairs of admissible data,* (W_Δ^0, G_Δ^0) *and* (Y_Δ^0, F_Δ^0)*,*

$$(W_\Delta^0, \{G_\Delta^0\}_{t^n \in [0,T]}) \to \{W_\Delta^n\}_{t^n \in [0,T]}$$

and

$$(Y_\Delta^0, \{F_\Delta^0\}_{t^n \in [0,T]}) \to \{Y_\Delta^n\}_{t^n \in [0,T]}$$

the following estimate holds uniformly for sufficiently small spatial grid, Δ*, and time step,* Δt

$$\|W_\Delta^n - Y_\Delta^n\| \leq C \left[\|W_\Delta^0 - Y_\Delta^0\| + \sum_{m=0}^{n} |||G_\Delta^m - F_\Delta^m||| \right] \tag{31}$$

where $\|\cdot\|$ *and* $|||\cdot|||$ *represent two different norms.*

We will satisfy the definition of stability by showing that a small perturbation or smearing

of the initial data results in a similar distortion of the numerical solution. First we rewrite Equation (26) in the following way

$$u_i^m - \frac{\Delta t D_{+,i}^m}{h^\alpha}\delta_\alpha^+ u_i^m - \frac{\Delta t D_{-,i}^m}{h^\alpha}\delta_\alpha^- u_i^m = \Delta t S_i^m + \Delta t f_i^m + \bar{u}_i^{m-1} \tag{32}$$

Let v^0 and u^0 be sufficiently close sets of initial data to Equation (26). Let v^n and u^n be the solutions to Equation (26) at the nth time step for the initial data v^0 and u^0, respectively. We then define $\rho_i^m = v_i^m - u_i^m$, $\bar{\rho}_i^m = \bar{v}_i^m - \bar{u}_i^m$, and $\rho_{S,i}^m = S(v_i^m, x_i, t^m) - S(u_i^m, x_i, t^m)$. The corresponding error equation then becomes

$$\rho_i^m - \frac{\Delta t D_{+,i}^m}{h^\alpha}\delta_\alpha^+ \rho_i^m - \frac{\Delta t D_{-,i}^m}{h^\alpha}\delta_\alpha^- \rho_i^m = \Delta t \rho_{S,i}^m + \bar{\rho}_i^{m-1} \tag{33}$$

for $i = 1, 2, \ldots, N-1$; $m = 1, 2, \ldots, M-1$

Denoting $\|\rho^m\|_\infty = \max_{1\le i\le N-1} |\rho_i^m|$ we obtain the following theorem.

Theorem 4.2. *Suppose that ρ_i^m is the solution of Equation* (33) *and the non-linear reaction term $S(u(x,t), x, t)$ satisfies a Lipschitz condition, Definition 2.2, with respect to $u(x,t)$, then there is a positive constant C such that*

$$\|\rho^m\|_\infty \le C\|\rho^0\|_\infty, m = 1, 2, \ldots, M-1$$

for sufficiently small time step, Δt.

Proof. We will use a proof by induction. When $m = 1$, assume that $|\rho_l^1| = \max\{|\rho_1^1|, |\rho_2^1|, \ldots, |\rho_{N-1}^1|\}$. Using the properties of the shifted Grunwald approximation in Lemma 3.2, we have

$$0 \le -\frac{\Delta t}{h^\alpha}\left(D_{+,l}^1 \delta_\alpha^+ |\rho_l^1| + D_{-,l}^1 \delta_\alpha^- |\rho_l^1|\right)$$

From Equation (33) we get the following inequality

$$
\begin{aligned}
|\rho_l^1| &\leq |\rho_l^1| - \frac{\Delta t}{h^\alpha}\left(D_{+,l}^1 \delta_\alpha^+ |\rho_l^1| + D_{-,l}^1 \delta_\alpha^- |\rho_l^1|\right) \\
&\leq \left|\rho_l^1 - \frac{\Delta t}{h^\alpha}\left(D_{+,l}^1 \delta_\alpha^+ \rho_l^1 + D_{-,l}^1 \delta_\alpha^- \rho_l^1\right)\right| \\
&= \left|\Delta t \rho_{S,l}^1 + \bar{\rho}_l^0\right|
\end{aligned} \tag{34}
$$

Applying the Lipschitz condition for $S(u(x,t),x,t)$ to Inequality (34) gives

$$|\rho_l^1| \leq \Delta t K |\rho_l^1| + |\bar{\rho}_l^0|$$

resulting in the inequality

$$|\rho_l^1| \leq \frac{1}{1-\Delta t K}|\bar{\rho}_l^0| \tag{35}$$

for $1 - \Delta t K > 0$. From the definition of $\bar{u}_i^m$ given in [24], $\bar{\rho}_l^0$ can be rewritten as $\bar{\rho}_l^0 = \left(1 - Cr_l^1\right)\rho_{k+1}^0 + Cr_l^1 \rho_k^0$. Where $Cr_l^1 = V_l^1 \frac{\Delta t}{h}$ is the Courant number, $\lfloor Cr_l^1 \rfloor$ is its floor function, and $k = l - \lfloor Cr_l^1 \rfloor - 1$. Therefore we get the following,

$$
\begin{aligned}
-\left((1-Cr_l^1)|\rho_l^0| + Cr_l^1|\rho_l^0|\right) \leq \rho_{k+1}^0 + Cr_l^1\left(\rho_k^0 - \rho_{k+1}^0\right) \leq (1-Cr_l^1)|\rho_l^0| + Cr_l^1|\rho_l^0|, \\
|\rho_{k+1}^0 + Cr_l^1\left(\rho_k^0 - \rho_{k+1}^0\right)| \leq (1-Cr_l^1)|\rho_l^0| + Cr_l^1|\rho_l^0|, \\
|\bar{\rho_l^0}| \leq \|\rho^0\|_\infty
\end{aligned} \tag{36}
$$

Remark 1. *It should be noted here that a similar argument can be used if the Courant number is negative by simply reversing the inequalities.*

Now applying Inequality (36) to Equation (35) we get

$$|\rho_l^1| \leq \frac{1}{1-\Delta t K}\|\rho^0\|_\infty$$

Let $C = \frac{1}{1-\Delta t K}$, thus we obtain

$$\|\rho^1\|_\infty \leq C\|\rho^0\|_\infty$$

Now suppose that

$$\|\rho^n\|_\infty \leq C\|\rho^0\|_\infty, \quad n = 2, \ldots, m$$

Assume that $|\rho_l^{m+1}| = \max\{|\rho_1^{m+1}|, |\rho_2^{m+1}|, \ldots, |\rho_{N-1}^{m+1}|\}$, we get that

$$\begin{aligned}
|\rho_l^{m+1}| &\leq |\rho_l^{m+1}| - \frac{\Delta t}{h^\alpha}\left(D_{+,l}^{m+1}\delta_\alpha^+|\rho_l^{m+1}| + D_{-,l}^{m+1}\delta_\alpha^-|\rho_l^{m+1}|\right) \\
&\leq \left|\rho_l^{m+1} - \frac{\Delta t}{h^\alpha}\left(D_{+,l}^{m+1}\delta_\alpha^+\rho_l^{m+1} + D_{-,l}^{m+1}\delta_\alpha^-\rho_l^{m+1}\right)\right| \\
&= \left|\Delta t\rho_{S,l}^{m+1} + \bar{\rho}_l^m\right|
\end{aligned}$$

Applying the Lipschitz condition for $S(u(x,t),x,t)$ and using the same method as for Equation (35)

$$\begin{aligned}
|\rho_l^{m+1}| &\leq \frac{1}{1-\Delta tK}\|\rho^m\|_\infty \\
&\leq \frac{1}{1-\Delta tK}C\|\rho^0\|_\infty
\end{aligned}$$

Let $C_0 = \frac{1}{1-\Delta tK}C$. Therefore

$$\|\rho^{m+1}\|_\infty \leq C_0\|\rho^0\|_\infty \tag{37}$$

□

4.1.2 Consistency

In order to prove convergence of the implicit method we must prove that it is stable and consistent, here we will prove consistency using the following definition motivated by the work presented in [19, 21]

Definition 4.3 (Consistency)**.** *Let $u_\Delta^{m+1} = Qu_\Delta^m + \Delta tG_\Delta^m$ be a finite difference scheme for a space fractional partial differential equation and U_Δ^m be the exact solution to the space fractional partial differential equation at t^m. The finite difference scheme is consistent with the partial differential equation if U_Δ^m satisfies*

$$U_\Delta^{m+1} = QU_\Delta^m + \Delta tG_\Delta^m + \Delta t\tau$$

and $\|\tau\| \to 0$ as $\Delta \to 0$ and $\Delta t \to 0$.

From this definition we get the following theorem for consistency of the implicit scheme.

Theorem 4.3 (Consistency of Implicit Scheme)**.** *Let $u(x,t^m)$ denote the exact solution of Equation* (24) *then the finite difference scheme defined in Equation* (26) *is consistent with Equation* (24) *as defined in Definition 4.3 with order of accuracy $(h,\Delta t)$.*

Proof. Substituting $u(x,t^m)$ into Equation (26) we get

$$\frac{u(x,t^m)-\bar{u}(x,t^{m-1})}{\Delta t}-\frac{D^m_{+,i}}{h^\alpha}\delta^+_\alpha u(x,t^m)-\frac{D^m_{-,i}}{h^\alpha}\delta^-_\alpha u(x,t^m)-S^m(u(x,t^m))=f^m$$

Applying Lemmas 3.1 and 3.3 and rewriting yields,

$$\frac{\partial u(x,t^m)}{\partial t}+V(x,t^m)\frac{\partial u(x,t^m)}{\partial x}-D_+(x,t^m)\frac{\partial^\alpha u(x,t^m)}{\partial_+ x^\alpha}-D_-(x,t^m)\frac{\partial^\alpha u(x,t^m)}{\partial_- x^\alpha}$$
$$-f(x,t^m)-S(u(x,t^m),x,t^m)=O(h)+O(h)+O(\Delta t)$$

Therefore $\tau=O(h)+O(h)+O(\Delta t)$ which implies $\|\tau\|\to 0$ as $h\to 0$ and $\Delta t\to 0$. Hence Equation (26) is consistent with Equation (24) and has accuracy of $(h,\Delta t)$ as desired. □

4.1.3 Convergence

Theorem 4.1 provides an argument for convergence of the finite difference scheme based on stability and consistency of the scheme. However, this theorem is typically used for linear equations or non-linear equations with sufficiently smooth solutions [19]. Instead of imposing additional restrictions we are able to prove convergence of the finite difference scheme directly.

Let $u(x_i,t^m)$ be the exact solution of Equation (6) at the (x_i,t^m) grid point and let u_i^m be the numerical solution using Equation (26) at the (x_i,t^m) grid point. We will now define $\xi_i^m=u(x_i,t^m)-u_i^m$, $\bar{\xi}_i^m=\bar{u}(x_i,t^m)-\bar{u}_i^m$ and $\Omega^m=(\xi_1^m,\xi_2^m,\ldots,\xi_{N-1}^m)$. Subtracting u_i^m

from $u(x_i, t^m)$ using the form in Equation (32) we get

$$\begin{aligned} &\xi_i^m - \frac{\Delta t D_{+,i}^m}{h^\alpha} \sum_{k=0}^{i+1} g_k^{(\alpha)} \xi_{i-k-1}^m - \frac{\Delta t D_{-,i}^m}{h^\alpha} \sum_{k=0}^{N-i+1} g_k^{(\alpha)} \xi_{i+k-1}^m \\ &= \Delta t(S(u(x_i, t^m), x_i, t^m) - S(u_i^m, x_i, t^m)) + \bar{\xi}_i^{m-1} + \tau_i^m \end{aligned} \tag{38}$$

where τ_i^m is the truncation error for a given grid point, $i = 1, 2, \ldots, N-1$ and $m = 0, 1, \ldots, M$. Assuming $\|\Omega^m\|_\infty = \max(|\xi_1^m|, |\xi_2^m|, \ldots, |\xi_{N-1}^m|)$ then we get the following theorem

Theorem 4.4. *If the reaction term $S(u(x,t)x,t)$ satisfies the Lipschitz condition, Definition 2.2, with respect to $u(x,t)$ and ξ_i^m is the solution to Equation* (38) *then there exists a positive constant, C^* such that*

$$\|\Omega^m\|_\infty \leq C^*(\Delta t + h)$$

for sufficiently small time step, Δt.

Proof. This proof will be a very similar argument to that presented for stability of the method. We assume that $|\xi_l^1| = \max(|\xi_1^1|, |\xi_2^1|, \ldots, |\xi_{N-1}^1|)$, and by similar fashion to Equation (34) we get the following inequality

$$|\xi_l^1| \leq |\bar{\xi}_l^0 + \Delta t(S(u(x_l, t^1), x_l, t^1) - S(u_l^1, x_l, t^1)) + \tau_l^1|$$

Now applying the Lipschitz condition, Definition 2.2, and the triangle inequality we get

$$(1 - \Delta t K)|\xi_l^1| \leq |\bar{\xi}_l^0| + |\tau_l^1|$$

Let $|\tau_0^k| = \max_{1 \leq i \leq N-1, 1 \leq m \leq M}(|\tau_i^m|)$ and note $\|\Omega^0\| = 0$ since the 0th time level is the initial condition which is the same for both $u(x_i, t^m)$ and u_i^m. Therefore we get

$$|\xi_l^1| = \|\Omega^1\|_\infty \leq \frac{1}{1 - \Delta t K} |\tau_0^k|$$

for $1 - \Delta tK > 0$. Now suppose

$$\|\Omega^n\|_\infty \leq C|\tau_o^k|$$

for $n = 1, 2, \ldots, m$ and C is some constant. Similarly we assume $|\xi_l^{m+1}| = \max(|\xi_1^{m+1}|, |\xi_2^{m+1}|, \ldots, |\xi_{N-1}^{m+1}|)$ which gives

$$|\xi_l^{m+1}| \leq |\bar{\xi}_l^m + \Delta t(S(u(x_l, t^{m+1}), x_l, t^{m+1})) - S(u_l^{m+1}, x_l, t^{m+1}) + \tau_l^{m+1}|$$

Again applying the Lipschitz condition, Definition 2.2, and the triangle inequality we get

$$(1 - \Delta tK)|\xi_l^{m+1}| \leq |\bar{\xi}_l^m| + |\tau_0^k| \tag{39}$$

Using the same argument as in Equation (36) we get that

$$|\bar{\xi}_l^m| \leq \|\Omega^m\|_\infty$$

Applying this to Equation (39) yields

$$(1 - \Delta tK)|\xi_l^{m+1}| \leq C|\tau_0^k| + |\tau_0^k|$$

Let $C^* = \frac{1}{1-\Delta tK}(1 + C)$ which gives

$$\|\Omega^{m+1}\|_\infty \leq C^*|\tau_0^k|$$

From Lemma 3.1 and Lemma 3.3 we get a contribution of $O(h)$ and $O(\Delta t)$ to $|\tau_0^k|$. Since the reaction term is taken at the current time level, there is no additional associated error. Therefore we get that

$$\|\Omega^{m+1}\|_\infty \leq C^*(O(h) + O(\Delta t))$$

□

4.2 Convergence of Explicit Scheme

We will now prove the convergence of the explicit scheme given in Equation (29). The arguments are similar in structure to those presented for the implicit form and will utilize the same basic theorems and definitions as used earlier for the implicit form.

4.2.1 Stability

As with the implicit scheme we will start by proving stability of the method. We will satisfy the definition of stability Definition 4.2, but first we will rewrite Equation (29) in the following way

$$u_i^m - \frac{\Delta t D_{+,i}^m}{h^\alpha}\delta_\alpha^+ u_i^m - \frac{\Delta t D_{-,i}^m}{h^\alpha}\delta_\alpha^- u_i^m = \Delta t S_i^m(\bar{u}_i^{m-1}) + \Delta t f_i^m + \bar{u}_i^{m-1}$$

As in the implicit case, we will let u^0 and v^0 to be sufficiently close sets of initial data to Equation (29). Let u^n and v^n be the solutions to Equation (29) at the nth time level for initial data u^0, v^0 respectively. We then let $\rho_i^m = v_i^m - u_i^m$, $\bar{\rho} = \bar{v}_i^m - \bar{u}_i^m$, and $\bar{\rho}_{S,i}^{m-1} = S(\bar{v}_i^{m-1}, x_i, t^m) - S(\bar{u}_i^{m-1}, x_i, t^m)$. The corresponding error equation then becomes

$$\rho_i^m - \frac{\Delta t D_{+,i}^m}{h^\alpha}\delta_\alpha^+ \rho_i^m - \frac{\Delta t D_{-,i}^m}{h^\alpha}\delta_\alpha^- \rho_i^m = \Delta t \bar{\rho}_{S,i}^{m-1} + \bar{\rho}_i^{m-1} \tag{40}$$

for $i = 1, 2 \ldots, N-1$; $m = 1, 2, \ldots, M-1$. Assuming $\|\rho^m\|_\infty = \max_{1<i<N-1} |\rho_i^m|$ we obtain the following theorem.

Theorem 4.5. *Suppose that ρ_i^m is the solution of Equation* (40) *and the nonlinear reaction term $S(u(x,t), x, t)$ satisfies the Lipschitz Condition, Definition 2.2, with respect to $u(x,t)$, then there is a positive constant C such that*

$$\|\rho^m\|_\infty \leq C\|\rho^0\|_\infty, \quad m = 1, 2, \ldots, M-1$$

for sufficiently small time step, Δt.

Proof. As in the implicit case we will use a proof by induction. When $m = 1$, assume that

$|\rho_l^m| = \max\{|\rho_1^m|, |\rho_2^m|, \ldots, |\rho_{N-1}^m|\}$. We want to show that $|\rho_l^m| \leq C\|\rho^0\|_\infty$, where C is some positive constant. From Lemma 3.2 we get

$$0 \leq -\frac{\Delta t}{h^\alpha}\left(D^1_{+,l}\delta^+_\alpha|\rho^1_l| + D^1_{-,l}\delta^-_\alpha|\rho^1_l|\right)$$

From Equation (40) we get the following inequality

$$\begin{aligned}
|\rho_l^1| &\leq |\rho_l^1| - \frac{\Delta t}{h^\alpha}\left(D^1_{+,l}\delta^+_\alpha|\rho^1_{l-k+1}| + D^1_{-,l}\delta^-_\alpha|\rho^1_{l+k-1}|\right) \\
&\leq \left|\rho_l^1 - \frac{\Delta t}{h^\alpha}\left(D^1_{+,l}\delta^+_\alpha\rho^1_{l-k+1} + D^1_{-,l}\delta^-_\alpha\rho^1_{l+k-1}\right)\right| \\
&= \left|\Delta t\bar{\rho}^0_{S,l} + \bar{\rho}^0_l\right|
\end{aligned} \tag{41}$$

Applying the Lipschitz condition for $S(u(x,t),x,t)$ to Equation (41) above we get

$$|\rho_l^1| \leq \Delta tK|\bar{\rho}_l^0| + |\bar{\rho}_l^0| = (1+\Delta tK)|\bar{\rho}_l^0|$$

Applying the same argument as Equation (36) we get

$$|\rho_l^1| \leq (1+\Delta tK)|\bar{\rho}_l^0| \leq (1+\Delta tK)\|\rho^0\|_\infty$$

Let $C = (1+\Delta tK)$ which gives

$$\|\rho^1\|_\infty \leq C\|\rho^0\|_\infty$$

Now suppose that $\|\rho^n\|_\infty \leq C\|\rho^0\|_\infty$ for $n = 2, \ldots, m$. Assume $|\rho_l^{m+1}| = \max\{|\rho_1^{m+1}|, |\rho_2^{m+1}|, \ldots, |\rho_{N-1}^{m+1}|\}$, we get

$$\begin{aligned}
|\rho_l^{m+1}| &\leq |\rho_l^{m+1}| - \frac{\Delta t}{h^\alpha}\left(D^{m+1}_{+,l}\delta^+_\alpha|\rho^{m+1}_{l-k+1}| + D^{m+1}_{-,l}\delta^-_\alpha|\rho^{m+1}_{l+k-1}|\right) \\
&\leq \left|\rho_l^{m+1} - \frac{\Delta t}{h^\alpha}\left(D^{m+1}_{+,l}\delta^+_\alpha\rho^{m+1}_{l-k+1} + D^{m+1}_{-,l}\delta^-_\alpha\rho^{m+1}_{l+k-1}\right)\right| \\
&= \left|\Delta t\bar{\rho}^{m+1}_{S,l} + \bar{\rho}^m_l\right|
\end{aligned}$$

Applying Lipschitz condition and using the same method as for Equation (41) we get

$$\|\rho_l^{m+1}| \leq (1 + \Delta t K)\|\rho^m\|_\infty \leq (1 + \Delta t K)C\|\rho^0\|$$

Let $C_0 = (1 + \Delta t K)C$, so we get

$$\|\rho^{m+1}\|_\infty \leq C_0\|\rho^0\|_\infty$$

as desired. □

4.2.2 Consistency

We will prove consistency in a similar fashion to the implicit method. We get the following theorem,

Theorem 4.6 (Consistency of Explicit Method)**.** *Let $u(x, t^m)$ denote the exact solution of Equation* (24) *then the finite difference scheme defined in Equation* (29) *is consistent with Equation* (24) *as defined in Definition 4.3 with order of accuracy $(h, \Delta t)$.*

Proof. The proof is a similar to the argument presented for the implicit method, however, since the non-linear reaction term is being evaluated at the previous time step this will introduce an additional error term. Substituting the expansion of $u(\bar{x}, t^{m-1})$ from Lemma 3.3 into the reaction term, we will expand the following

$$S(u(x, t^m) - \Delta t \left(u_x(x, t^m)\frac{b(x, t^m)}{c(x, t^m)} + u_t(x, t^m) + \dots \right), x, t^m)$$

in the $u(x, t^m)$ variable. Using a Taylor's expansion we get

$$S(u(x, t^m), x, t^m) - \Delta t S_u(u(x, t^m), x, t^m) \left(u_x(x, t^m)\frac{b(x, t^m)}{c(x, t^m)} + u_t(x, t^m) + \dots \right) + \dots$$

which yields a truncation error of

$$S(u(\bar{x}, t^{m-1}), x, t^m) = S(u(x, t^m), x, t^m) + O(\Delta t)$$

Now substituting $u(x,t^m)$ into Equation (29), applying Lemmas 3.1 and 3.3 and the results above we get

$$\frac{\partial u(x,t^m)}{\partial t} + V(x,t^m)\frac{\partial u(x,t^m)}{\partial x} - D_+(x,t^m)\frac{\partial^\alpha u(x,t^m)}{\partial_+ x^\alpha} - D_-(x,t^m)\frac{\partial^\alpha u(x,t^m)}{\partial_- x^\alpha}$$
$$-f(x,t^m) - S(u(x,t^m),x,t^m) = O(h)+O(h) + O(\Delta t) + O(\Delta t)$$

Therefore $\tau = O(h) + O(h) + O(\Delta t) + O(\Delta t)$ which implies $\|\tau\| \to 0$ as $h \to 0$ and $\Delta t \to 0$. Hence Equation (29) is consistent with Equation (24) and has accuracy of $(h, \Delta t)$ as desired. □

4.2.3 Convergence

As with the implicit method, we can provide a direct proof for convergence of the finite difference scheme. Let $u(x_i,t^m)$ be the exact solution of Equation (6) at the (x_i,t^m) grid point and let u_i^m be the numerical solution using Equation (29) at the (x_i,t^m) grid point. We will now define $\xi_i^m = u(x_i,t^m) - u_i^m$, $\bar{\xi}_i^m = \bar{u}(x_i,t^m) - \bar{u}_i^m$ and $\Omega^m = (\xi_1^m, \xi_2^m, \ldots, \xi_{N-1}^m)$. Substituting $u(x_i,t^m)$ in Equation (32) for u_i^m and subtracting Equation (32) we get

$$\begin{aligned} &\xi_i^m - \frac{\Delta t D_{+,i}^m}{h^\alpha}\delta_\alpha^+ \xi_i - \frac{\Delta t D_{-,i}^m}{h^\alpha}\delta_\alpha^- \xi_i^m \\ &= \Delta t(S(\bar{u}(x_i,t^m),x_i,t^m) - S(\bar{u}_i^m,x_i,t^m)) + \bar{\xi}_i^{m-1} + \tau_i^m \end{aligned} \tag{42}$$

where τ_i^m is the truncation error for a given grid point, $i = 1,2,\ldots,N-1$ and $m = 0,1,\ldots,M-1$. Assuming $\|\Omega^m\|_\infty = \max(|\xi_1^m|, |\xi_2^m|, \ldots, |\xi_{N-1}^m|)$ then we get the following theorem

Theorem 4.7 (Convergence of Explicit Method)**.** *Let $S(u(x,t),x,t)$ in Equation (24) satisfy the Lipschitz condition, Definition 2.2, then the numerical scheme given in Equation (29) is convergent of order $((h)^1,(\Delta t)^1)$ where Δt is the time step and h is the spatial step of the numerical scheme.*

Proof. This proof will follow a very similar argument to that presented for stability of the method. We assume that $|\xi_l^1| = \max(|\xi_1^1|, |\xi_2^1|, \ldots, |\xi_{M-1}^1|)$, and by similar fashion to

Equation (34) we get the following inequality

$$|\xi_l^1| \leq |\bar{\xi}_l^0 + \Delta t(S(\bar{u}(x_l, t^1), x_l, t^1) - S(\bar{u}_l^1, x_l, t^1)) + \tau_l^1|$$

Now applying the Lipschitz condition, Definition 2.2, and the triangle inequality

$$|\xi_l^1| \leq (1 + \Delta t K)|\bar{\xi}_l^0| + |\tau_l^1|$$

Let $|T_0^k| = \max_{1 \leq i \leq M-1, 1 \leq m \leq N}(|\tau_i^m|)$ and note $\|\Omega^0\| = 0$ since the 0th time level is the initial condition which is the same for both $u(x_i, t^m)$ and u_i^m. Therefore we get

$$|\xi_l^1| = \|\Omega^1\|_\infty \leq |\tau_0^k|$$

Now suppose

$$\|\Omega^n\|_\infty \leq C|\tau_o^k| \tag{43}$$

for $n = 1, 2, \ldots, m$ and C is some constant. Similarly we assume $|\xi_l^{m+1}| = \max(|\xi_1^{m+1}|, |\xi_2^{m+1}|, \ldots, |\xi_{N-1}^{m+1}|)$ which gives

$$|\xi_l^{m+1}| \leq |\bar{\xi}_l^m + \Delta t(S(\bar{u}(x_l, t^{m+1}), x_l, t^{m+1}) - S(\bar{u}_l^m, \bar{x}_l, t^{m+1})) + \tau_l^{m+1}|$$

Again applying the Lipschitz condition, Definition 2.2, and the triangle inequality we get

$$|\xi_l^{m+1}| \leq (1 + \Delta t K)|\bar{\xi}_l^m| + |\tau_0^k| \tag{44}$$

Using the same argument as in Equation (36) we get that

$$|\bar{\xi}_l^m| \leq \|\Omega^m\|_\infty$$

Applying this with Equation (43) and Equation (44) we arrive at

$$\|\Omega^{m+1}\| \leq (2 + C)|\tau_0^k|$$

From Lemma 3.1 and Lemma 3.3 we get a contribution of $O(\Delta x)$ and $O(\Delta t)$ to $|\tau_0^k|$. The explicit approximation of the reaction term gives an additional first order temporal error which can easily be verified by Taylor expansion. Let $C^* = (2 + C)$, thus we get

$$\|\Omega^{m+1}\| \leq C^*(O(h) + O(\Delta t) + O(\Delta t))$$

□

5 Newton's Method

The implicit scheme given in Equation (26) contains a non-linear reaction term yielding a non-linear algebraic system. It is possible to take the non-linear term in the previous time step, making the scheme linear and solvable by current dense matrix techniques, however the benefit of utilizing the characteristic method is the larger time step that can be taken, reducing the overall size of the problem. An additional temporal error term is added in the explicit scheme by taking the reaction at the previous time step, potentially requiring smaller time steps. To avoid this additional error we take the reaction at the current time level, avoiding this additional error term. We then solve this non-linear system with Newton's method.

5.1 Derivation of System

Here we will derive the linear system to be solved with Newton's method. From Definition 2.3 Newton's method for systems is given as

$$J_F(x_n)\Delta x_n = -F(x_n)$$
$$x_{n+1} = x_n + \Delta x_n$$

where $J_f(x_n)$ is the Jacobian of $F(x)$, and $F(x) = 0$ is the function to be solved. So for the implicit scheme given in Equation (26), we look at the equivalent matrix equation given in Equation (27). We rewrite it in the following fashion

$$F(u^m) = (I + A^m)u^m - \Delta t S^m - \bar{u}^{m-1} - \Delta t f^m = 0 \tag{45}$$

which gives the following Jacobian

$$J_F(u^m) = I + A^m - \Delta t (S^m)^*(u^m) \tag{46}$$

where $S^*(u^m)$ is defined as follows

$$(S^m)^*(u^m) = \begin{pmatrix} \frac{\partial}{\partial u_1^\alpha} S(u_1^m) & 0 & \dots & 0 \\ 0 & \frac{\partial}{\partial u_2^\alpha} S(u_2^m) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{\partial}{\partial u_{N-1}^\alpha} S(u_{N-1}^m) \end{pmatrix}$$

Rewriting in terms of x_n, where $x_0 = \bar{u}^{m-1}$ will be used as the initial guess to start the iteration, the linear system becomes

$$(I + A^m - \Delta t (S^m)^*(x_n))(x_{n+1} - x_n) = -F(x_n) \tag{47}$$

This linear system can then be solved by methods for dense linear systems which we will present later in this paper.

5.2 Convergence of Newton's Method

In this section we will analyze the convergence of Newton's method using Theorem 2.1. This gives conditions for an initial guess to guarantee convergence of Newton's method. The idea is that the starting guess is the solution from the previous time level, so that

solution can be made arbitrarily close to the solution at the current time level by making the time step smaller. Here we will derive conditions for Δt to guarantee convergence of Newton's Method.

Theorem 5.1. *Let $S(u(x,t),x,t)$ be Lipschitz continuous, $f(x,t)$ be bounded, and $u(x,t)$ be continuous then there exists a Δt such that the Newton iterates to Equation* (47) *converge by Theorem 2.1. We define the Newton iterate by Definition 2.3 .*

Proof. From Corollary 2.1, we have the following condition for convergence

$$\|J_F^{-1}\|\|J_F F\|K \leq \|J_F^{-1}\|\|J_F\|\|F\|K_F \leq 1/2$$

Where F is the function in Equation (45) evaluated at the initial guess. J_F is the Jacobian defined in Equation (46) evaluated at the initial guess, and J_F^{-1} is the inverse of the Jacobian evaluated at the initial guess. K_F is the constant from the Lipschitz condition for F.

We use the Gershgorin Circle Theorem to estimate $\|J_F\|$ and $\|J_F^{-1}\|$ norms. The diagonals of J_F are

$$diag(J_F(u_i^m)) = \{1 + \frac{\Delta t}{h^\alpha}\left(D_{+,i}^m + D_{-,i}^m\right)\alpha - \Delta t S'(u_i^m);\ \text{for } i = 1,2,\ldots,N-1\} \tag{48}$$

And from the properties of the Shifted Grunwald approximation given in Lemma 3.2 and the fact that $S^*(u^m)$ is a diagonal matrix, we know that the sum of the off diagonal row entries of A^m is less than the diagonal entry. Therefore the Gershgorin disks are located at $1 + \frac{\Delta t}{h^\alpha}\left(D_{+,i}^m + D_{-,i}^m\right)\alpha - \Delta t S'(u_i^m)$ with radius of $\frac{\Delta t}{h^\alpha}\left(D_{+,i}^m + D_{-,i}^m\right)\alpha$. This argument is given in greater detail in both [24, 11]. This gives the following approximation for $\|J_F(u^m)\|$ and $\|J_F^{-1}(u^m)\|$

$$\begin{aligned}\|J_F(u^m)\| &\leq \max\{|1 + \frac{2\Delta t}{h^\alpha}\left(D_{+,i}^m + D_{-,i}^m\right)\alpha - \Delta t (S^m)^*(u_i^m)|; i = 1,2,3,\ldots,N-1\}\\ \|J_F^{-1}(u^m)\| &\leq \frac{1}{\min\{|1 - \Delta t (S^m)^*(u_i^m)|; i = 1,2,3,\ldots,N-1\}}\end{aligned}$$

Lipschitz continuity of $S(u(x,t),x,t)$ implies that the derivative is bounded. Substituting, we get the following

$$\begin{aligned} \|J_F(u^m)\| \leq & |1 + \frac{2\Delta t}{h^\alpha}\left(D^m_{+,i} + D^m_{-,i}\right)\alpha + \Delta t K| \\ \|J_F^{-1}(u^m)\| \leq & \frac{1}{|1-\Delta t K|} \end{aligned} \tag{49}$$

We now want to get a bound for $\|F(\bar{u}^{m-1})\|$. Substituting $\bar{u}^{m-1}$ into Equation (45) we get

$$F(\bar{u}^{m-1}) = (I + A^m)\bar{u}^{m-1} - \Delta t (S^m)^*(\bar{u}^{m-1}) - \bar{u}^{m-1} - \Delta t f^m.$$

Reducing and applying the triangle inequality,

$$\|F(\bar{u}^{m-1})\| = \|A^m\bar{u}^{m-1} - \Delta t(S^m)^*(\bar{u}^{m-1}) - \Delta t f^m\| \leq \|A^m\|\|\bar{u}^{m-1}\| + \Delta t\|(S^m)^*(\bar{u}^{m-1})\| + \Delta t\|f^m\|$$

Using the results above for the norm of A^m, Lipschitz continuity of $S(u(x,t),x,t)$ and boundedness of f^m, we get the following result

$$\|F(\bar{u}^{m-1})\| \leq |\frac{2\Delta t}{h^\alpha}(D_{+,i} + D_{-,i})|\|\bar{u}^{m-1}\| + \Delta t K + \Delta t\|f^m\|$$

We know that $\|\bar{u}^{m-1}\| \leq \|u^{m-1}\|$, and since f^m is bounded, $\|f^m\|$ is finite. Combining all the norms, we get the following sufficient condition for convergence of Newton's Method.

$$\frac{|1 + \frac{2\Delta t}{h^\alpha}(D^m_{+,i} + D_{-,i})\alpha + \Delta t K|}{|1 - \Delta t K|}\left(\frac{2\Delta t}{h^\alpha}(D_{+,i} + D_{-,i})|\|\bar{u}^{m-1}\| + \Delta t K + \Delta t\|f^m\|\right) K_F \leq \frac{1}{2} \tag{50}$$

It is easy to see that the limit of the left hand side of Inequality (50) goes to zero as $\Delta t \to 0$. Therefore, there exists a positive Δt such that the initial condition for Newton's method, $\bar{u}^{m-1}$ lies within the set A from Theorem 2.1. □

Remark 2. *In order to get an overall estimate for the maximum Δt that can guarantee convergence of Newton's method, one would need to know $u(x,t)$, which may not be readily available. However, it is possible to get a bound for Δt for each time step since Equation* (50)

only uses data from the previous time step. Therefore it may be possible to implement a more optimized selection for Δt in a variable time step scheme.

6 Fast Iterative Algorithm

A typical problem with the use of fractional derivatives is the non-local nature of the differentiation [24, 18, 4] which typically produces dense matrices in numerical schemes [24]. In order to solve Equation (26) and Equation (29) we will use an iterative solver. In both cases we will employ the Conjugate Gradient method, modifying an efficient algorithm proposed in [24], and in the implicit we will use Newton's method to solve a non-linear system. We will start with the explicit method since it requires the least modification.

6.1 Explicit Algorithm

We are concerned with solving the system in Equation (30). We note that the matrix $I+A^m$ is nonsingular, strictly diagonally dominant [24, 12], however it is non-symmetric. There are a number of methods to solve this system, such as the conjugate gradient squared (CGS) method, generalized minimal residual (GMRES), multigrid, and many others [24, 6, 16]

Since Equation (6) is advection dominated and the advective term gets absorbed into the characteristic tracking term the condition number of $I + A^m$ should be small [24]. Thus we will use the algorithm proposed in [24], the conjugate gradient normal residual (CGNR) method. We will then be solving the following system

$$(I + A^m)^T(I + A^m)u^m = (I + A^m)^T\bar{u}^{m-1} + \Delta t(I + A^m)^T\bar{S}^{m-1} + \Delta t(I + A^m)^T f^m$$

If we let

$$B^m = (I + A^m)^T(I + A^m)$$

$$b^m = (I + A^m)^T\bar{u}^{m-1} + \Delta t(I + A^m)^T\bar{S}^{m-1} + \Delta t(I + A^m)^T f^m$$

then we are solving the system $B^m u^m = b^m$. The iterative algorithm is given in Algorithm 1.

Data: $u^n = u_0$
Result: Solution to $B^m u^m = b^m$
Step 1;
initialization;
$y = b^m - B^m u^m$, $\rho = y^T y$;
Step 2;
if $k = 1$ **then**
  $p = y$
else
  $\beta = \rho/\bar{\rho}$, $p = y + \beta p$
end
$w = B^m p$
$\alpha = \rho/p^T w$, $u^m = u^m + \alpha p$
$y = y - \alpha w$, $\bar{\rho} = \rho$, $\rho = y^T y$;
Step 3;
if $\|\rho\|_2 > \epsilon$ **then**
  Go to Step 2
end

Algorithm 1: Conjugate Gradient Method

The largest cost in Algorithm 1 is the matrix vector multiplication $B^m u^m$. Because A^m is dense, this multiplication if done naively is a $O(n^2)$ operation, making the aforementioned algorithm a potentially $O(n^3)$ operation. However because A^m is a Toeplitz matrix, it is possible to exploit this structure. In [24] a fast Toeplitz matrix multiplication is used to reduce the $B^m u^m$ multiplication from $O(n^2)$ to $O(n \cdot log(n))$ operational complexity. This is possible since A^m can be presented as a Fourier decomposition [24, 6, 3].

First we must explain the construction of the algorithm. It is well known that a circulant matrix, C_n can be decomposed in the following way [24, 6, 3].

$$C_n = F_n^{-1} \text{diag}(F_n c) F_n$$

where c is a first column of C_n and F_n is the $n \times n$ discrete Fourier transform matrix. It is clear from Equation (28) that A^m can be decomposed into two Toeplitz matrices, A_L^m and

A_R^m, one representing the left hand diffusion and the other the right hand diffusion [24]. A $N-1 \times N-1$ Toeplitz matrix, T_{N-1} can be embedded into a $(2N-2) \times (2N-2)$ circulant matrix in the following way [24, 3]

$$C_{2N-2} = \begin{bmatrix} T_{N-1} & B_{N-1} \\ B_{N-1} & T_{N-1} \end{bmatrix}$$

where B_{N-1} is defined as

$$B_{N-1} = \begin{bmatrix} 0 & t_{2-N} & \dots & t_{-2} & t_{-1} \\ t_{N-2} & 0 & t_{2-N} & \dots & t_{-2} \\ \vdots & t_{N-2} & 0 & \ddots & \vdots \\ t_2 & \vdots & \ddots & \ddots & t_{2-N} \\ t_1 & t_2 & \dots & t_{N-2} & 0 \end{bmatrix}$$

Embedding our Toeplitz matrices A_L^m and A_R^m, we get the following circulant matrices.

$$C_{2N-2,L} = \begin{bmatrix} A_L^m & B_L^m \\ B_L^m & A_L^m \end{bmatrix}, C_{2N-2,R} = \begin{bmatrix} A_R^m & B_R^m \\ B_R^m & A_R^m \end{bmatrix},$$

So the matrix-vector products $A_L^m u^m$ and $A_R^m u^m$ can be obtained as the first half of the matrix-vector multiplication of $C_{2N-2,L} u^m$ and $C_{2N-2,R} u^m$ respectively [24]. We thus get Algorithm 2 for the fast matrix-vector multiplication. Applying Algorithm 2 in Algo-

Data: A^m, u^m
Result: Solution to $A^m u^m$
Step 1 $w_{2N-2} = F_{2N-2} u_{2N-2} = FFT(u_{2N-2})$
Step 2 $v_{2N-2,L} = F_{2N-2} c_{2N-2,L} = FFT(c_{2N-2,L})$
$v_{2N-2,R} = F_{2N-2} c_{2N-2,R} = FFT(c_{2N-2}, R)$
Step 3 $z_{2N-2,L} = w_{2N-2} \dot{v}_{2N-2,L} = [w_1 v_{1,L}, \dots, w_{2N-2} v_{2N-2,L}]^T$
$z_{2N-2,R} = w_{2N-2} \dot{v}_{2N-2,R} = [w_1 v_{1,R}, \dots, w_{2N-2} v_{2N-2,R}]^T$
Step 4 $y_{2N-2,L} = F_{2N-2}^{-1} z_{2N-2,L} = IFFT(z_{2N-2,L})$
$y_{2N-2,R} = F_{2N-2}^{-1} z_{2N-2,R} = IFFT(z_{2N-2,R})$
Step 5 Define y_L and y_R to be the first half of $y_{2N-2,L}$ and $y_{2N-2,R}$ respectively
Step 6 $A^m u = -\frac{\Delta t}{h^\alpha}(d_+^m \cdot y_l + d_-^m \cdot y_R)$

Algorithm 2: Fast Matrix Multiplication

rithm 1 for all matrix-vector multiplications reduces the operational complexity from $O(n^3)$ to $O(n^2 log(n))$ to solve the system defined in Equation (30).

6.2 Implicit Algorithm

In order to solve Equation (26), which is a system of non-linear equations, we apply Newton's method for systems, Equation (47). This involves an additional loop in the algorithm; one for Newton's method in addition to the one for solving the linear Jacobian system. Therefore we will apply both Algorithm 1 and Algorithm 2 to solve this system using Algorithm 3. Note that there is an additional matrix-vector multiplication per time level caused by the

Data: $u^n = u_0$
Result: Solution to Equation (26)
Step 1;
initialization;
$x_0 = u_0$, $k = 1$
Step 2;
while $m < N$ **do**
  initialize b^m
  if $k = 1$ **then**
    Solve $(I + A^m - \Delta t S^*(x_0))\delta = b^m$
  end
  while $\|\delta\|_2 < \epsilon$ **do**
    $x_n = x_{n-1} + \delta$
    Solve $(I + A^m - \Delta t S^*(x_n))\delta = b^m$
  end
end

Algorithm 3: Newtons Method

initialization of Newton's method. In terms of operational complexity, the implicit method is $O(mn^2 log(n))$ where m is based on the number of Newton iterations required to solve the non-linear system. This number is dependent on the initial guess and the stopping criteria (ϵ). By the sequential criterion of continuous functions we get that as $\Delta t \to 0$, $u_i^{m-1} \to u_i^m$. Therefore it is possible to select Δt small enough so that it will be within the error bounds of Algorithm 3. This implies that the algorithm will only execute one loop, therefore Algorithm 1 once. So its operational complexity will approach that of the explicit method as $\Delta t \to 0$.

7 Extension to Two Spatial Dimensions

The methods discussed so far have been for one space dimension. We will now present an extension of the one dimensional methods to two dimensions by using an extension of the Method of Characteristics [7] and the Shifted Grunwald approximation [13]. In order to solve the finite difference schemes, we will employ an Alternating Direction Implicit (ADI) method to reduce the computational time [13, 23]. We will consider an extension of Equation (24) to two dimensions, presented here

$$\begin{aligned}
&\frac{\partial u}{\partial t} + V(x,y,t)\frac{\partial u}{\partial x} + W(x,y,t)\frac{\partial u}{\partial y} - D_{+,x}(x,t)\frac{\partial^\alpha u}{\partial_+ x^\alpha} - D_{-,x}(x,t)\frac{\partial^\alpha u}{\partial_- x^\alpha} \\
&-D_{+,y}(y,t)\frac{\partial^\gamma u}{\partial_+ y^\gamma} - D_{-,y}(y,t)\frac{\partial^\gamma u}{\partial_- y^\gamma} = f(x,y,t) + S(u(x,y,t),x,y,t) \\
&0 < t \leq T, \\
&u(x,y,0) = u_0(x,y), \quad (x,y) \in [X_1, X_2] \times [Y_1, Y_2] \\
&u(X_1,y,t) = 0, \quad u(X_2,y,t) = 0, t \in [0,T], y \in [Y_1, Y_2] \\
&u(x,Y_1,t) = 0, \quad u(x,Y_2,t) = 0, t \in [0,T], x \in [X_1, X_2]
\end{aligned} \tag{51}$$

Where $1 < \alpha \leq 2$, $1 < \gamma \leq 2$ are orders of fractional derivatives, $X_1 < x < X_2$ and $Y_1 < y < Y_2$ is a finite rectangular domain and $D_{+,x}(x,t), D_{-,x}(x,t), D_{+,y}(y,t), D_{-,y}(y,t) > 0$. Assume that Equation (51) has a unique and sufficiently smooth solution, $u(x,y,t)$.

As with the one-dimensional case, we can include $V_x(x,y,t)u$ and $W_y(x,y,t)$ from the divergence form in the reaction term $S(u(x,y,t),x,y,t)$ and the reaction term is Lipschitz continuous with respect to $u(x,y,t)$. We will define a uniform grid, Δ, with step size of h in both x and y for ease of analysis. It is possible to have different step sizes for x and y, or even non-uniform grids, however, there are additional restrictions that must be imposed with respect to the interpolation that can be used in the Method of Characteristics [7].

7.1 Extension of the Method of Characteristics

We will start developing a finite difference method by first presenting an extension of the Method of Characteristics. The derivation presented earlier, as well as the work in [7], is not dependent on the dimension of the equation being used. Therefore it should be straight forward to extend the method to multiple spacial dimensions.

We set $\psi(x,y,t) = \sqrt{1+V^2(x,y,t)+W^2(x,y,t)}$ and the time and advective terms in Equation (51) can be rewritten as

$$\psi(x,y,t)\frac{du(x,y,t)}{dt} = \frac{\partial u}{\partial t} + V(x,y,t)\frac{\partial u}{\partial x} + W(x,y,t)\frac{\partial u}{\partial y}$$

where, as in the one-dimensional case, we define a new differential operator in the direction of the characteristic curve as

$$\frac{d}{dt} = \frac{1}{\psi(x,y,t)}\frac{\partial}{\partial t} + \frac{V(x,y,t)}{\psi(x,y,t)}\frac{\partial}{\partial x} + \frac{W(x,y,t)}{\psi(x,y,t)}\frac{\partial}{\partial y}$$

and we get the finite difference approximation

$$\psi(x,y,t)\frac{du(x,y,t)}{dt} \approx \frac{u^m - \bar{u}^{m-1}}{\Delta t} \tag{52}$$

where $t^m = m\Delta t$, $u^m = u(x,y,t^m)$ and $\bar{u}^{m-1} = u(\bar{x},\bar{y},t^{m-1})$. We define $\bar{x}$ and $\bar{y}$ in a similar fashion as the one dimensional case.

$$\bar{x} = x - V(x,y,t^m)(t^m - t) \tag{53}$$

$$\bar{y} = y - W(x,y,t^m)(t^m - t) \tag{54}$$

Where x, y are the current grid locations and t is the time level desired. For our purposes $t = t^{m-1}$. In Figure 4 we see an example of how a characteristic trace in two dimensions may look. If you look at the trace from either the $x-t$ or $y-t$ planes, it will look like the

characteristic trace in the one-dimensional case. We now present the following lemma for

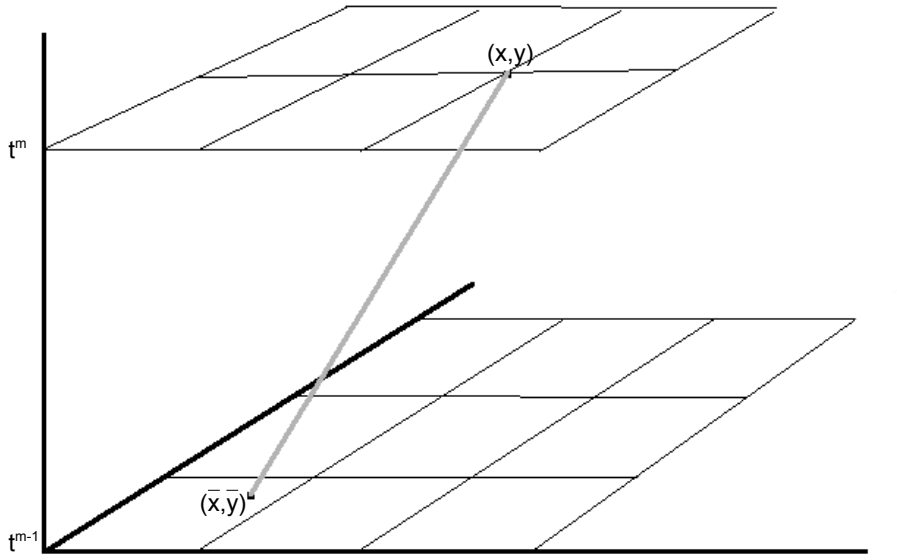

Figure 4: An example of characteristic trace from time level t^m to t^{m-1} in two spacial dimensions.

the truncation error of the characteristic method in the multi-dimensional case.

Lemma 7.1 (Two Dimensional Characteristic Error)**.** *The approximation given in Equation* (52) *approximates*

$$\frac{\partial u}{\partial t} + V(x,y,t)\frac{\partial u}{\partial x} + W(x,y,t)\frac{\partial u}{\partial y}$$

with a truncation error $O(\Delta t)$*.*

Proof. We want to show that

$$\frac{\partial u(x,y,t^m)}{\partial t} + V(x,y,t^m)\frac{\partial u(x,y,t^m)}{\partial x} + W(x,y,t^m)\frac{\partial u(x,y,t^m)}{\partial y} - \frac{1}{\Delta t}(u(x,y,t^m) - u(\bar{x},\bar{y},t^{m-1})) = O(\Delta t)$$

Using the definitions of $\bar{x}$ and $\bar{y}$ given in Equations (53) and (54) with $t = t^{m-1}$, and noting

that $t^m - t^{m-1} = \Delta t$, we get the Taylor's expansion of Equation (52)

$$\frac{1}{\Delta t}(u(x,y,t^m) - u(\bar{x},\bar{y},t^{m-1})) = \frac{1}{\Delta t}[u(x,y,t^m) - u(x,y,t^m) + u_x(x,y,t^m)V(x,t^m)\Delta t$$
$$+ u_y(x,y,t^m)W(x,y,t^m)\Delta t + u_t(x,y,t^m)\Delta t - u_{xx}(x,t^m)V^2(x,y,t^m)(\Delta t)^2 + \ldots]$$

which reduces to

$$\begin{aligned} \frac{1}{\Delta t}(u(x,y,t^m) - u(\bar{x},\bar{y},t^{m-1})) &= u_t + V(x,y,t^m)u_x + W(x,y,t^m)u_y \\ -u_{xx}V^2(x,y,t^m)\Delta t + u_{xt}V(xy,,t^m)\Delta t - u_{xy}V(xy,,t^m)W(x,y,t^m)\Delta t + \ldots & \end{aligned} \tag{55}$$

Now subtracting Equation (55) from $\frac{\partial u(x,y,t^m)}{\partial t} + V(x,t^m)\frac{\partial u(x,y,t^m)}{\partial x} + W(y,t^m)\frac{\partial u(x,y,t^m)}{\partial y}$ we obtain

$$\frac{\partial u(x,y,t^m)}{\partial t} + V(x,y,t^m)\frac{\partial u(x,y,t^m)}{\partial x} + W(x,y,t^m)\frac{\partial u(x,y,t^m)}{\partial y} - \frac{1}{\Delta t}(u(x,y,t^m) - u(\bar{x},\bar{y},t^{m-1})) = O(\Delta t)$$

as desired. □

7.2 Extension of Shifted Grúnwald Approximation

We will now present the extension of the one-dimensional Shifted Grúnwald presented earlier. For ease of the analysis we are using a uniform grid in both x and y, with a step size of h. So we will define $N_{x,-} = (x - X_1)/h$, $N_{x,+} = (X_2 - x)/h$, $N_{y,-} = (y - Y_1)/h$ and $N_{y,+} = (Y_2 - y)/h$. We should note that with this derivation the N constants are not guaranteed to be integers. Therefore we will pick our square grid so that the N constants are integers. Note, this is only for ease of the presentation and comparison of operational complexity. We would normally define the grid based on how many steps, the N constants, we wished to use. Thus, the shifted Grúnwald approximation is [13]

$$\frac{\partial^\alpha u(x,y,t)}{\partial_+ x^\alpha} = \frac{1}{\Gamma(-\alpha)}\lim_{h\to 0}\frac{1}{h^\alpha}\sum_{k=0}^{N_{x,+}}\frac{\Gamma(k-\alpha)}{\Gamma(k+1)}u(x+(k-1)h,y,t)$$

Note that the Grúnwald weights, g_k are still dependent only on the order of the differentiation and the number of partitions. Therefore Lemma 3.2 applies to the multidimensional case as well. We will adopt the following notation for the shifted Grúnwald based on the single variable case:

$$\delta^{-}_{\alpha,x}u(x,y,t) = \frac{1}{\Gamma(-\alpha)}\frac{1}{h^{\alpha}}\sum_{k=0}^{N_{x,-}}\frac{\Gamma(k-\alpha)}{\Gamma(k+1)}u(x-(k-1)h,y,t) \tag{56}$$

$$\delta^{+}_{\alpha,x}u(x,y,t) = \frac{1}{\Gamma(-\alpha)}\frac{1}{h^{\alpha}}\sum_{k=0}^{N_{x,+}}\frac{\Gamma(k-\alpha)}{\Gamma(k+1)}u(x+(k-1)h,y,t) \tag{57}$$

$$\delta^{-}_{\gamma,y}u(x,y,t) = \frac{1}{\Gamma(-\gamma)}\frac{1}{h^{\gamma}}\sum_{k=0}^{N_{y,-}}\frac{\Gamma(k-\gamma)}{\Gamma(k+1)}u(x,y-(k-1)h,t) \tag{58}$$

$$\delta^{+}_{\gamma,y}u(x,y,t) = \frac{1}{\Gamma(-\gamma)}\frac{1}{h^{\gamma}}\sum_{k=0}^{N_{y,+}}\frac{\Gamma(k-\gamma)}{\Gamma(k+1)}u(x,y+(k-1)h,t) \tag{59}$$

These approximations have a first order truncation error in their respective spacial dimension. The proof for this is nearly the same as in the one-dimensional case, so we will use Lemma 3.1 (see [11] Theorem 2.7 for a proof of the one-dimensional case and [13] Theorem 3.3 for two-dimensional). We are now ready to develop our two-dimensional finite difference schemes.

7.3 Two-Dimensional Finite Difference Schemes

As with the one-dimensional case we will propose two methods; one with the reaction taken at the current time level and one with the reaction taken at the previous time level along the characteristic trace. Again we will have a non-linear system and a linear system respectively. This will motivate two different solution techniques. We will start with the explicit scheme first.

7.3.1 Two-Dimensional Explicit Finite Difference Scheme

First we will index elements of our grid, Δ. Let $x_i = X_1 + ih$ and $y_j = Y_1 + jh$. We will continue to use the same temporal notation as before, $t^m = m\Delta t$. We will also use the following notation for $u(x, y, t)$, let $u^m_{i,j} = u(x_i, y_j, t^m)$. Now applying Method of Characteristics and the shifted Grunwald, we get the following finite difference scheme

$$\begin{aligned}\frac{u^m - \bar{u}^{m-1}}{\Delta t} - \frac{D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x}u^m - \frac{D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}u^m - \frac{D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y}u^m \\ - \frac{D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}u^m - S^m(\bar{u}^{m-1}) = f^m\end{aligned} \tag{60}$$

Which can be rewritten in the following form

$$\left(1 - \frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} - \frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x} - \frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} - \frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)u^m = \bar{u}^{m-1} + \Delta t S^m(\bar{u}^{m-1}) + \Delta t f^m \tag{61}$$

The above two-dimensional finite difference scheme is stable and consistent. Both can be proven in nearly the same fashion as the one-dimensional case. There is only a slight modification to account for the additional derivative term in the y-direction for the stability proof. Consistency involves a little more modification, but only to the truncation error of the reaction term. However, the expansion of $u(\bar{x}, \bar{y}, t)$ was given in Lemma 7.1 and replaces the $\bar{u}$ term in the one dimensional case, making the proof of truncation error essentially the same proof as the one dimensional case.

This method involves solving a non-sparse linear system for $(N-1) \times (N-1)$ unknowns, essentially N^2, unknowns. Assuming $\Delta t \approx h$, a naive iterative solver implementation, such as CGNR, will have an operational complexity of $O(N(N^2)^2)$ or $O(N^5)$, making this very computationally intensive to solve especially for a finer Δ.

7.3.2 Two-Dimensional Implicit Finite Difference Scheme

Using the same grid indexing, applying Method of Characteristics, shifted Grúnwald and taking the reaction term at the current time level we get the following finite difference

scheme,

$$\frac{u^m - \bar{u}^{m-1}}{\Delta t} - \frac{D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x}u^m - \frac{D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}u^m - \frac{D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y}u^m - \frac{D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}u^m - S^m(u^m) = f^m \tag{62}$$

which can be rewritten as,

$$\left(I - \frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} - \frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x} - \frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} - \frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,x}\right)u^m - \Delta t S^m(u^m) = \bar{u}^{m-1} + \Delta t f^m \tag{63}$$

Again, this finite difference method is stable and consistent and it can be proven with small modifications to the proofs in the one-dimensional case. For this method there is even less modification than for the explicit scheme. One additional issue is the convergence criteria determined from the one-dimensional case, it happens to be same for the multi-dimensional case. The result is dependent on only the non-linear reaction term and the time step, both independent of the number of spacial dimensions in the system, so the results will be the same for both the one and two dimensional cases.

As in the one-dimensional case, this is a system of non-linear equations. So we will employ a non-linear solver namely Newton's method to solve this system. However, after constructing the Jacobian system, we are left with a non-sparse linear system with essentially N^2 unknowns. This is computationally intensive to solve, as with the explicit method.

8 Two-Dimensional ADI Methods

Above we presented two finite difference methods for the two-dimensional space-fractional advection-diffusion equation given in Equation (51). Both methods were determined to be computationally intensive since they solved non-sparse linear systems with $N \times N$ unknowns giving a computational complexity of $O(N^5)$ for a naive implementation. We propose a modification to the Alternating Direction Implicit (ADI) method proposed in [13] for the explicit method and a similar algorithm for solving the Jacobian system in the implicit scheme. We will begin with the explicit method since it requires the least amount of modification.

8.1 ADI Method for the Explicit Finite Difference Scheme

We will present here the explicit ADI method and its proof for convergence. Convergence of the ADI method relies heavily on the behavior of the mixed derivatives [13]. We will first present the method then prove that it is stable and consistent thus showing that it is convergent.

8.1.1 Explicit ADI Method

The idea of the ADI method is to solve the problem a direction at a time. We will use an adaptation of the ADI method developed by Meerschaert and Tadjeran, [13]. We will apply it to our two-dimensional ADE, whereas their method is for a two-dimensional dispersion equation. The method developed in [13] is a modification of common integer-order ADI methods, a detailed description can be found in [21]. We will decompose the operators in Equation (61) into an operator in x and an operator in y directions.

$$\begin{aligned}\left(1-\left(\frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x}+\frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)\left(1-\left(\frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y}+\frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right)u^m \\ = \bar{u}^{m-1}+\Delta t S^m(\bar{u}^{m-1})+\Delta t f^m\end{aligned} \tag{64}$$

This is not the same as Equation (61), in fact this will add additional $(\Delta t)^2$ terms to the finite difference scheme, introducing an additional $O(\Delta t \cdot h)$ perturbation error, which we will later prove. We will solve Equation (64) in two steps, first in the x-direction for the intermediate variable $u^{m-\frac{1}{2}}$

$$\left(1-\left(\frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x}+\frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)u_j^{m-\frac{1}{2}}=\bar{u}^{m-1}+\Delta t S^m(\bar{u}_j^{m-1})+\Delta t f^m \tag{65}$$

for each fixed j. Then, using the intermediate step solution, $u^{m-\frac{1}{2}}$, from Equation (65) we solve in the y-direction for the final solution u^m,

$$\left(1-\left(\frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y}+\frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right)u_i^m=u_i^{m-\frac{1}{2}} \tag{66}$$

for each fixed i.

For each step, we are solving a non-sparse linear system with N unknowns N times. Assuming a naive implementation, each linear system has an operational complexity of $O(N^2)$, solved N times, for each step. So the final operational complexity for a single time step is $O(N^3)$. Assuming a similar temporal step size, we get a final operational complexity of $O(N^4)$, a significant improvement over the direct approach. As we will see later, we can reduce this even farther by utilizing some fast algorithms developed for the one-dimensional case.

8.1.2 Stability of Explicit ADI Method

To prove stability, we must show that a small perturbation, or smearing, of the initial data produces a corresponding change in the numerical solution. To do this, we will show that the operator norms of the operators in Equations (65) and (66) are greater and bounded away from one, so their inverse operators have norms less than one. The following form for Equation (64) will be useful,

$$A^m B^m u^m = \bar{u}^{m-1} + \Delta t S^m(\bar{u}^{m-1}) + \Delta t f^m \tag{67}$$

where B^m represents the $\left(1 - \left(\frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} + \frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right)$ operator and A^m represents the $\left(1 - \left(\frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} + \frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)$ operator. We can also rewrite Equation (65) in the following way,

$$A^m_j u_j^{m-\frac{1}{2}} = \bar{u}^{m-1}_j + \Delta t S^m_j(\bar{u}^{m-1}_j) + \Delta t f^m_j \tag{68}$$

where A_j^m is the $(N-1)\times(N-1)$ coefficient matrix at a fixed y-index, j. $A_j^m = [a_{i,k}^m]$ has the following structure for $i = 1, \ldots, N_x - 1$ and $k = 1, \ldots, N_x - 1$

$$a_{i,k}^m = -\frac{\Delta t}{h^\alpha}\begin{cases} 1 - \left(D_{+,i}^m + D_{-,i}^m\right) g_1^{(\alpha)}, & k = i \\ \left(D_{+,i}^m g_2^{(\alpha)} + D_{-,i}^m g_0^{(\alpha)}\right), & k = i-1 \\ \left(D_{+,i}^m g_0^{(\alpha)} + D_{-,i}^m g_2^{(\alpha)}\right), & k = i+1 \\ D_{+,i}^m g_{i-j+1}^{(\alpha)}, & k < i-1 \\ D_{-,i}^m g_{j-i+1}^{(\alpha)}, & k > i+1 \end{cases} \tag{69}$$

The y-direction can be rewritten in a very similar fashion, the structure of the coefficient matrix is the same as in the x-direction. We now get the following lemma.

Lemma 8.1. *Each one dimensional system defined by Equations* (65) *and* (66) *is unconditionally stable for* $1 < \alpha \leq 2$ *and* $1 < \gamma \leq 2$.

Proof. This proof will follow very closely the argument presented in Theorem 3.2 [13]. Using Equation (68), we have to show that the operator norm of A_j^m is larger than 1 then any small perturbation in the initial data will be dampened through the calculation. We will now apply the Gershgorin circle theorem to the matrix in Equation (69) and use the properties of the Grunwald weights, Lemma 3.2. Since $g_1^{(\alpha)} = -\alpha$, and $\sum_{k=0}^{\infty} g_k^{(\alpha)} = 0$, we get that the eigenvalues of A_j^m have a magnitude strictly larger than 1. Therefore the eigenvalues of the inverse operator, $[A_j^m]^{-1}$ have eigenvalues with magnitude strictly less than 1. Thus $\|[A_j^m]^{-1}\| < 1$. This argument is given in greater detail in [24, 13] and was used previously for the convergence of Newton's method in the one-dimensional case. So the one-dimensional system defined in Equation (65) is unconditionally stable. We can use the same argument for the y-direction. □

Now we want to show that if the operators A^m and B^m commute, then the scheme is unconditionally stable. Commutativity of these two operators is a common assumption for the classic ADI-Euler methods [13]. We note that A^m is a block-diagonal matrix, meaning $A^m = diag(A_1, \ldots, A_k)$ and so the eigenvalues of the matrix A^m are in the union of the Gershgorin disks, and so the operator norm of A^m has a magnitude strictly greater than 1,

and B^m follows a similar argument. Therefore, the norms of the inverse operators of both B^m and A^m have magnitude strictly less than one. This gives us the following theorem

Theorem 8.1 (Stability of Explicit ADI Method)**.** *The ADI method, defined in Equation* (64) *is unconditionally stable for* $1 < \alpha \leq 2$ *and* $1 < \gamma \leq 2$.

Proof. Let $u(x, y, 0)$ and $w(x, y, 0)$ be sufficiently close sets of initial data $\rho^m = u(x, y, t^m) - w(x, y, t^m)$, and $\rho_S^{m-1} = S(u(x, y, t^{m-1}), x, y, t^m) - S(v(x, y, t^{m-1}), x, y, t^m)$. So we get

$$\rho^1 = [B^0]^{-1}[A^0]^{-1}(\rho^0 + \Delta t \rho_S^0)$$

and now for $m = 2$, we get

$$\rho^2 = [B^1]^{-1}[A^1]^{-1}[B^0]^{-1}[A^0]^{-1}(\rho^0 + \Delta t \rho_S^0) + [B^1]^{-1}[A^1]^{-1}(\Delta t \rho_S^1)$$

This process repeats in this fashion until we get to the mth time level. We get that

$$\rho^m = \prod_{k=0}^{m-1} [B^k]^{-1}[A^k]^{-1}\rho^0 + \sum_{k=0}^{m-1} \Delta t \rho_S^k \prod_{i=k}^{m-1} [B^i]^{-1}[A^i]^{-1}$$

Taking the norm of both sides and applying the triangle inequality,

$$\|\rho^m\| \leq \| \prod_{k=0}^{m-1} [B^k]^{-1}[A^k]^{-1}\rho^0\| + \| \sum_{k=0}^{m-1} \Delta t \rho_S^k \prod_{i=k}^{m-1} [B^i]^{-1}[A^i]^{-1}\|$$

and since the operator norms of $[A^m]^{-1}$ and $[B^m]^{-1}$ are less than one, we get the following inequality.

$$\|\rho^m\| \leq \|\rho^0\| + \sum_{k=0}^{m-1} \|\|\Delta t \rho_S^k\|\|$$

Thus, this method is stable by Definition 4.2. □

8.1.3 Consistency of Explicit ADI Method

In order to prove consistency of the explicit ADI method, we have to first determine the truncation error associated with the mixed derivatives. We define the mixed derivative

below [13].

$$\delta^{-}_{\alpha,x}\delta^{-}_{\gamma,y}f(x,y) = \frac{1}{h^{\gamma}}\frac{1}{h^{\alpha}}\sum_{n=0}^{N_x}\sum_{m=0}^{N_y} g_n^{(\alpha)} g_m^{(\gamma)} f(x-(n-1)h, y-(m-1)h) \tag{70}$$

All other combinations of mixed derivatives (both left and right hand) can be expressed in a similar fashion and have been omitted. From [13], Theorem 3.1, we get the following lemma.

Lemma 8.2 (Mixed Fractional Derivative Truncation Error)**.** *The numerical approximation for the mixed fractional derivative in Equation* (70) *is first order consistent, i.e.*

$$\frac{\partial^{\alpha}}{\partial_{-}x^{\alpha}}\frac{\partial^{\gamma}}{\partial_{-}y^{\gamma}}f(x,y) = \delta^{-}_{\alpha,x}\delta^{-}_{\gamma,y}f(x,y) + O(h)$$

uniformly in $(x,y) \in \mathbb{R}^2$.

We should note that Lemma 8.2 applies for all combinations of left and right hand mixed derivatives. Using this lemma, the consistency analysis for the explicit ADI method is a small modification to the consistency analysis for the one-dimensional case.

Theorem 8.2 (Consistency of Explicit ADI Method)**.** *Let* $u(x,y,t^m)$ *denote the exact solution of Equation* (51) *then the ADI method defined in Equation* (67) *is consistent with Equation* (51) *as defined in Definition 4.3 and has order an of accuracy* $O(h,\Delta t)$.

Proof. Substituting $u(x,y,t^m)$ into Equation (67)

$$A^m B^m u^m = \bar{u}^{m-1} + \Delta t S^m(\bar{u}^{m-1}) + \Delta t f^m$$

Using a similar technique as for the explicit one-dimensional case, we get a first order truncation error from the non-linear reaction term. Applying Lemmas 8.2, 3.1 and 3.3 we get

$$\begin{aligned}
&\frac{\partial u}{\partial t} + V(x,y,t)\frac{\partial u}{\partial x} + W(x,y,t)\frac{\partial u}{\partial y} - D_{+,x}(x,t)\frac{\partial^{\alpha} u}{\partial_{+}x^{\alpha}} - D_{-,x}(x,t)\frac{\partial^{\alpha} u}{\partial_{-}x^{\alpha}} \\
&-D_{+,y}(y,t)\frac{\partial^{\alpha} u}{\partial_{+}y^{\gamma}} - D_{-,y}(y,t)\frac{\partial^{\alpha} u}{\partial_{-}y^{\gamma}} - f(x,y,t) - S(u(x,y,t),x,y,t) \\
&= O(h) + O(\Delta t) + O(h\Delta t)
\end{aligned}$$

□

So we get an additional $O(h\Delta t)$ perturbation error, which is small compared to the other terms. We have shown that the Explicit ADI method is consistent and stable, therefore by Theorem 4.1, we get that the Explicit ADI method is convergent.

8.2 Two-Dimensional Implicit Finite Difference Scheme

As in the one-dimensional case, the implicit representation of the non-linear reaction term yields a non-linear system of equations. We again apply Newton's method to solve this system. As with the explicit scheme, the resulting Jacobian system is a non-sparse linear system with $(N-1) \times (N-1)$ unknowns which is computationally intensive to solve. Therefore, we propose a modification of the ADI method to solve this system.

8.2.1 Implicit ADI Method

Applying Newton's method to Equation (63), we obtain the Jacobian system to be solved for each Newton iterate

$$(I - \frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} - \frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x} - \frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} - \frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,x} - \Delta t S^*(x_n))(\Delta x_n) = -F(x_n)$$

where J_F is

$$J_F = I - \frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} - \frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x} - \frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} - \frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,x} - \Delta t S^*(x_n)$$

we will define $F(x_n)$ later on as we analyze the stability of the method. We should note that $F(x_n)$, which we will refer to as the equivalent system, is not the same as Equation (63). As with the explicit case, there are additional $(\Delta t)^2$ terms added to the finite difference scheme, which will introduce additional perturbation error. The idea is to solve this system in the x-direction, the y-direction, then the non-linear reaction as with the ADI method proposed

for the explicit method. We decompose the Jacobian system in the following manner.

$$\begin{aligned}(I-(\frac{\Delta tD^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x}+\frac{\Delta tD^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}))(I-(\frac{\Delta tD^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y}+\frac{\Delta tD^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}))\\ \cdot(I-\Delta tS^*(x_n))(\Delta t)=-F(x_n)\end{aligned} \tag{71}$$

We solve Equation (71) in three steps. First in the x-direction for the intermediate solution $x_j^{\frac{1}{3}}$ for each fixed y grid index, j, using the following equation

$$\left(I-\left(\frac{\Delta tD^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x}+\frac{\Delta tD^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)x_j^{\frac{1}{3}}=-F(x_j)$$

Then, using the intermediate solution $x^{\frac{1}{3}}$ found above, we find the second intermediate solution, $x_i^{\frac{2}{3}}$ by solving in the y-direction for each fixed x grid index, i, using the following

$$\left(I-\left(\frac{\Delta tD^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y}+\frac{\Delta tD^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right)x_i^{\frac{2}{3}}=x_i^{\frac{1}{3}}$$

Finally we solve for Δx_n using the intermediate solution $x^{\frac{2}{3}}$ by solving the following system

$$(I-\Delta tS^*(x_n))(\Delta x_n)=x^{\frac{2}{3}}$$

In the first and second steps, we are solving a non-spare linear system with $N-1$ unknowns $N-1$ times. Assuming a naive implementation, the operational complexity of solving a dense linear system with $N-1$ unknowns is $O(N^2)$. We are solving N such systems in the first step. Therefore the operational complexity of the first step is $O(N^3)$. We repeat this same process for the second step since the solving technique is the same. Therefore the operational complexity of completing the first and second steps is $O(N^3)$.

For the last step, we should note that the system is a diagonal system, therefore it can be computed in linear time. We have $N-1\times N-1$ unknowns, therefore the operational complexity of the last step is $O(N^2)$. Therefore, to complete one Newton iteration is $O(N^3)$. The Newton iterations may need to be completed multiple times, say M, to get a solution

sufficiently close, within some tolerance, ϵ. Assuming a similar temporal step size as for the grid size $\Delta t \approx h$, the operational complexity of the implicit ADI method is then $O(MN^4)$; assuming a similar number of iterations for Newton's method to converge.

8.2.2 Stability of Implicit ADI Method

Our ADI method is actually being applied in the Newton iterations, thus the system we need to analyze for the stability is the system Newton's method is actually solving for. Taking Equation (71) and distributing the fractional derivative operators into the $(I - \Delta t S^*(x_n))$ term we get

$$\begin{aligned} J_F = &\left(I - \left(\tfrac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} + \tfrac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)\left(I - \left(\tfrac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} + \tfrac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right) \\ &-\Delta t S^*(x_n)\left(I - \left(\tfrac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} + \tfrac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)\left(I - \left(\tfrac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} + \tfrac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right) \end{aligned}$$

We made perturbations to the J_F term in the Newton iterations. We have no reason to suspect that the terms independent of u in the original system are modified by this perturbation. So, in determining the equivalent system we assume that terms such as f^m remain the same between the original system and our modified system. Using this new J_F term, derive our equivalent system here

$$\begin{aligned} &\left(I - \left(\tfrac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} + \tfrac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)\left(I - \left(\tfrac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} + \tfrac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right)u^m \\ &-\left(I - \left(\tfrac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} + \tfrac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}\right)\right)\left(I - \left(\tfrac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} + \tfrac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,y}\right)\right)\Delta t S^m(u^m) \\ &-\bar{u}^{m-1} - \Delta t f^m = 0 \end{aligned} \tag{72}$$

This is also our $F(x_n)$ term on the right hand side of Equation (71). For ease of presentation we let $\delta_{x,\alpha} = \frac{\Delta t D^m_{+,x}}{h^\alpha}\delta^+_{\alpha,x} + \frac{\Delta t D^m_{-,x}}{h^\alpha}\delta^-_{\alpha,x}$ and $\delta_{y,\gamma} = \frac{\Delta t D^m_{+,y}}{h^\gamma}\delta^+_{\gamma,y} + \frac{\Delta t D^m_{-,y}}{h^\gamma}\delta^-_{\gamma,x}$. Substituting and expanding we get the following

$$\begin{aligned} (I - \delta_{x,\alpha} - \delta_{y,\gamma} + \delta_{x,\alpha}\delta_{y,\gamma})u^m - \Delta t S^m(u^m) + \Delta t \delta_{x,\alpha} S^m(u^m) + \Delta t \delta_{y,\gamma} S(u^m) \\ - \Delta t \delta_{x,\alpha}\delta_{y,\gamma} S(u^m) = \bar{u}^{m-1} + \Delta t f^m \end{aligned}$$

Let v^0 and u^0 be sufficiently close sets of initial data to Equation (51). Let v^m and u^m be the numerical solutions to Equation (51) at the mth time level. Now we define $\rho^m_{i,j} = v^m_{i,j} - u^m_{i,j}$, $\bar{\rho}^m_{i,j} = \bar{v}^m_{i,j} - \bar{u}^m_{i,j}$ and $\rho^m_{S,i,j} = S(v^m_{i,j}, x_i, y_j, t^m) - S(u^m_{i,j}, x_i, y_j, t^m)$. We then rewrite our equivalent system as

$$\begin{aligned}
&\rho^m_{i,j} - \tfrac{\Delta t D_{+,i}}{h^\alpha}\delta^+_{x,\alpha}\rho^m_{i,j} - \tfrac{\Delta t D_{-,i}}{h^\alpha}\delta^-_{x,\alpha}\rho^m_{i,j} - \tfrac{\Delta t D_{+,j}}{h^\gamma}\delta^+_{y,\gamma}\rho^m_{i,j} - \tfrac{\Delta t D_{-,j}}{h^\gamma}\delta^-_{y,\gamma}\rho^m_{i,j} + \delta_{x,\alpha}\delta_{y,\gamma}\rho^m_{i,j} \\
&= \left(-\tfrac{\Delta t D_{+,i}}{h^\alpha}\delta^+_{x,\alpha} - \tfrac{\Delta t D_{-,i}}{h^\alpha}\delta^-_{x,\alpha} - \tfrac{\Delta t D_{+,j}}{h^\gamma}\delta^+_{y,\gamma} - \tfrac{\Delta t D_{-,j}}{h^\gamma}\delta^-_{y,\gamma}\right)\Delta t\rho^m_{S,i,j} \\
&+\Delta t\delta_{x,\alpha}\delta_{y,\gamma}\rho^m_{S,i,j} + \Delta t\rho^m_{S,i,j} + \bar{\rho}^{m-1}_{i,j}
\end{aligned} \tag{73}$$

for $i = 1, 2, \ldots, N-1; j = 1, 2, \ldots, N-1; m = 1, 2, \ldots, M$.

Theorem 8.3. *Suppose that $\rho^m_{i,j}$ is the solution of Equation* (73) *and the nonlinear reaction term $S(u(x,y,t),x,y,t)$ satisfies Lipschitz condition with respect to $u(x,y,t)$ then there is a positive constant C such that*

$$\|\rho^m\| \le C\|\rho^0\|_\infty, \quad m = 1, 2, \ldots, M,$$

for sufficiently small time step, Δt.

Proof. We will use a proof by induction. When $m = 1$, assume that $|\rho^1_l| = \max\{|\rho^1_{1,1}|, |\rho^1_{2,1}|, \ldots, |\rho^1_{1,2}|, |\rho^1_{2,2}|, \ldots, |\rho^1_{N_x-1,N_y-1}|\}$. By the properties of the Grunwald weights, Lemma 3.2, we have

$$0 \le \delta_{x,\alpha}\delta_{y,\gamma}\rho^1_l$$

From Equation (73) we get

$$\begin{aligned}
|\rho^1_l| &\le |\rho^1_l| - \frac{\Delta t D^m_{+,l}}{h^\alpha}\delta^+_{x,\alpha}|\rho^1_l| - \frac{\Delta t D^m_{-,l}}{h^\alpha}\delta^-_{x,\alpha}|\rho^1_l| - \frac{\Delta t D^m_{+,l}}{h^\gamma}\delta^+_{y,\gamma}|\rho^1_l| - \frac{\Delta t D^m_{-,l}}{h^\gamma}\delta^-_{y,\gamma}|\rho^1_l| - (-\delta_{x,\alpha}\delta_{y,\gamma}|\rho^1_l|) \\
&\le |\rho^1_l - \frac{\Delta t D^m_{+,l}}{h^\alpha}\delta^+_{x,\alpha}\rho^1_l - \frac{\Delta t D^m_{-,l}}{h^\alpha}\delta^-_{x,\alpha}\rho^1_l - \frac{\Delta t D^m_{+,l}}{h^\gamma}\delta^+_{y,\gamma}\rho^1_l - \frac{\Delta t D^m_{-,l}}{h^\gamma}\delta^-_{y,\gamma}\rho^1_l - (-\delta_{x,\alpha}\delta_{y,\gamma}\rho^1_l)| \\
&= |\left(-\frac{\Delta t D^m_{+,l}}{h^\alpha}\delta^+_{x,\alpha} - \frac{\Delta t D^m_{-,i}}{h^\alpha}\delta^-_{x,\alpha} - \frac{\Delta t D^m_{+,j}}{h^\gamma}\delta^+_{y,\gamma} - \frac{\Delta t D^m_{-,j}}{h^\gamma}\delta^-_{y,\gamma}\right)\Delta t\rho^1_{S,l} \\
&+ \Delta t\delta_{x,\alpha}\delta_{y,\gamma}\rho^1_{S,l} + \Delta t\rho^1_{S,l} + \bar{\rho}^0_l|
\end{aligned}$$

Applying Lipschitz continuity, using properties of the Grúnwald weights and expanding the mixed derivative term, we get the following

$$|\rho_l^1| \leq \frac{2\alpha(\Delta t)^2 D_{+,l}}{h^\alpha} K|\rho_l^1| + \frac{2\alpha(\Delta t)^2 D_{-,l}}{h^\alpha} K|\rho_l^1| + \frac{2\gamma(\Delta t)^2 D_{-,l}}{h^\gamma} K|\rho_l^1| + \frac{2\gamma(\Delta t)^2 D_{+,l}}{h^\gamma} K|\rho_l^1|$$
$$+ \frac{2\alpha(\Delta t)^2 D_{+,l}}{h^\alpha} \frac{2\gamma(\Delta t)^2 D_{-,l}}{h^\gamma} K|\rho_l^1| + \frac{2\alpha(\Delta t)^2 D_{+,l}}{h^\alpha} \frac{2\gamma(\Delta t)^2 D_{+,l}}{h^\gamma} K|\rho_l^1| + \frac{2\alpha(\Delta t)^2 D_{-,l}}{h^\alpha} \frac{2\gamma(\Delta t)^2 D_{+,l}}{h^\gamma} K|\rho_l^1|$$
$$+ \frac{2\alpha(\Delta t)^2 D_{-,l}}{h^\alpha} \frac{2\gamma(\Delta t)^2 D_{-,l}}{h^\gamma} K|\rho_l^1| + \Delta t K|\rho_l^1| + |\bar{\rho}_l^0|$$

We get these $\frac{2\alpha(\Delta t)^2 D_{+,l}}{h^\alpha} K|\rho_l^1|$ terms from the application of the fractional differential operators to the non-linear reaction term. We are evaluating the $S(u(x,y,t),x,y,t)$ term at a maximum value of $|\rho^1|$. However, the fractional operator evaluates at every point at that fixed x or y. Thus, $S(u(x,y,t),x,y,t)$ is not necessarily positive. However, by Lipschitz continuity, we know that $S(u(x,y,t),x,y,t)$ evaluated at every ρ^1 is bounded by $|S(\rho_l^1,x,y,t)|$. Hence the maximum the differential operator applied to $S(u(x,y,t),x,y,t)$ can be is $\frac{2\alpha(\Delta t)^2 D_{+,l}}{h^\alpha} K|\rho_l^1|$ by the properties of the shifted Grunwald and Lipschitz condition. We move all $|\rho_l^1|$ terms from the right hand side to the left hand size and factor and divide. We will not write the terms of the constant C here for ease of exposition. From this we get

$$|\rho_l^1| \leq \frac{1}{C}|\bar{\rho}_l^0|$$

Note that here we get a convergence criteria, the constant C must be positive in order for this inequality to hold. Using the same argument as Equation (36) we get

$$|\rho_l^1| \leq \frac{1}{C}|\bar{\rho}_l^0| \leq \frac{1}{C}\|\rho^0\|_\infty$$

Now suppose that

$$\|\rho^m\|_\infty \leq \frac{1}{C}\|\rho^0\|_\infty$$

we apply the same methodology used for the case of $m = 1$, and we get

$$\|\rho^m\|_\infty \leq \frac{1}{C^*}\frac{1}{C}\|\rho^0\|_\infty$$

as desired. □

8.2.3 Consistency of Implicit ADI Method

The consistency of the implicit ADI method follows almost the same argument as in the explicit case. The only difference between the two is the introduction of the fractional differential operators applied to the non-linear reaction term. In this case, there is no additional perturbation error associated with the non-linear reaction since it is evaluated at the current time level. Therefore, the perturbation error comes from the fractional differential operators and mixed operators, whose truncation error has been discussed earlier.

Theorem 8.4 (Consistency of Implicit ADI Method)**.** *Let $u(x, y, t^m)$ denote the exact solution of Equation* (51) *then the ADI method defined in Equation* (71) *for the finite difference scheme, Equation* (62)*, is consistent with Equation* (51) *as defined in Definition 4.3 and has accuracy of $O(h, \Delta t)$.*

Proof. Substituting $u(x, y, t^m)$ into the equivalent finite difference scheme shown in Equation (72) and applying Lemmas 8.2, 3.1 and 3.3 we get

$$\begin{aligned}
&\frac{\partial u}{\partial t} + V(x,y,t)\frac{\partial u}{\partial x} + W(x,y,t)\frac{\partial u}{\partial y} - D_{+,x}(x,t)\frac{\partial^\alpha u}{\partial_+ x^\alpha} - D_{-,x}(x,t)\frac{\partial^\alpha u}{\partial_- x^\alpha} \\
&-D_{+,y}(y,t)\frac{\partial^\gamma u}{\partial_+ y^\gamma} - D_{-,y}(y,t)\frac{\partial^\gamma u}{\partial_- y^\gamma} - f(x,y,t) - S(u(x,y,t),x,y,t) \\
&= O(h) + O(\Delta t) + O(h\Delta t)
\end{aligned}$$

as desired. □

We should note that from the differential operator applied to the non-linear reaction we get an additional $O(h\Delta t)$, while from the mixed derivative operator applied to the non-linear reaction we get an additional $O(h(\Delta t)^2)$.

9 Numerical Experiments

In this section we conduct numerical experiments to investigate the performance of the methods detailed. We use two examples, one with a slow diffusion and slow reaction and

one with fast diffusion and fast reaction. A simple reaction term $cu^2(1-u)$ was chosen, where c is some constant. We chose this reaction based on commonly used reaction terms in similar one-dimensional equations. In the context of our equation, this may be modeling either a chemical reaction occurring in porous media, or in the case of the logistic equation in the fast diffusion fast reaction experiment, perhaps a biological containment such as a bacteria in an aquifer. We have presented the analysis in this paper in terms of an abstract reaction term $S(u(x,t),x,t)$ to allow for more diverse selection of reactions based on various problems that may be proposed. We provide examples based on common integer-order problems here.

9.1 Slow Diffusion, Slow Reaction

In this experiment we look at the convergence of the implicit and explicit methods to each other. We take an initial condition used by [24] given here

$$u(x,t) = \frac{1}{7\pi}\int_0^\infty e^{-2D|cos(\frac{\pi x}{2})|(t+20.5)\xi^\alpha} cos(\xi(x-Vt-.5V))d\xi \tag{74}$$

evaluated at $t=0$. The conditions for the experiment are chosen as follows: $\alpha = 1.8$, $(a,b) = (0,3)$, $[0,T] = [0,1]$, $V(x,t) = 1$, $D_+(x,t) = D_-(x,t) = .01$, $f(x,t) = 0$ and $S(u(x,t),x,t) = .1u^2(x,t)(1-u(x,t))$. We look at the difference between the explicit and implicit methods, comparing their computational times and how quickly they converge to each other. In Figure 5 we can see the two solutions plotted against each other.

M	N	L^1	L^2	L^∞	Explicit (s)	Implicit (s)
32	256	1.27E-003	1.10E-003	1.84E-003	.516	1.19
256	1024	1.61E-004	1.39E-004	2.32E-004	15.9	32.6
1024	1024	3.93E-005	3.29E-005	5.33E-005	62.3	98.1
1024	2048	4.13E-005	3.50E-005	5.77E-005	130	306
2048	2048	2.28E-005	1.81E-005	2.80E-005	255	522

Table 1: Difference between Implicit and Explicit methods for various grids with corresponding CPU times for example one runs

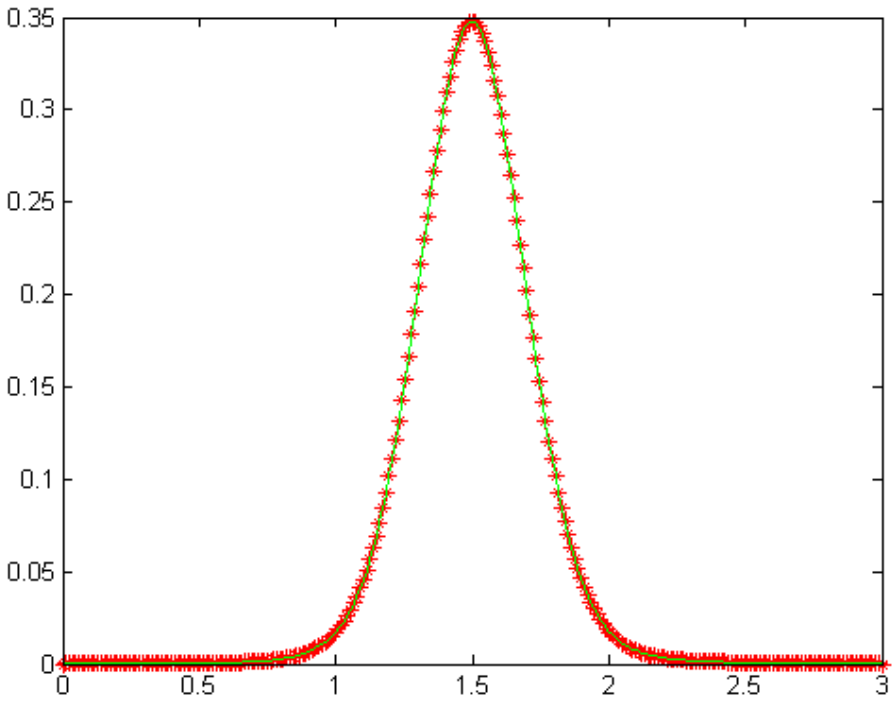

Figure 5: The solutions for implicit and explicit methods for the slow diffusion slow reaction problem at 1024 time steps and 256 spatial steps. The solid green line is the explicit solution and the stars are the implicit solution

In Table 1 we have provided the results on various grids. It is clear that as we take a finer grid the two methods are converging to the same solution. Looking at the error as we increase the number of either time or spatial steps, we see that the error appears to be decreasing in a linear fashion. This is consistent with our theoretical analysis. We are interested in convergence rates of the individual methods. It is not obvious from our comparison test which is converging faster, or the computational time required.

The trade off between the explicit and implicit methods is error vs computational time. Our theoretical analysis indicates the explicit method is less computationally complex, but has an additional $O(\Delta t)$ error term. In order to explore this trade off, we take a high resolution grid and compute both solutions on that grid. We take a sufficiently fine enough

grid that the error between the methods is less than 10^{-5} and use this solution as a base line comparison between the methods. We then select a fine spatial grid and examine the convergence to the base line solution as we take finer temporal grids. Tables 2 and 3 show the results of runs with a fixed spatial grid of 1024 steps and variable temporal grids.

M	N	L^1	L^2	L^∞	Time (s)
16	1024	4.01E-003	5.12E-003	1.20E-002	1.04
32	1024	3.00E-003	3.74E-003	8.63E-003	2.13
64	1024	2.42E-003	2.94E-003	6.68E-003	4.08
128	1024	1.91E-003	2.28E-003	5.14E-003	7.76

Table 2: Difference between explicit and high resolution solutions with 1024 spatial steps and with CPU for example one

M	N	L^1	L^2	L^∞	Time (s)
16	1024	3.84E-003	4.93E-003	1.14E-002	2.48
32	1024	2.96E-003	3.71E-003	8.50E-003	4.46
64	1024	2.41E-003	2.93E-003	6.65E-003	8.94
128	1024	1.91E-003	2.28E-003	5.13E-003	17.8

Table 3: Difference between implicit and high resolution solutions with 1024 spatial steps and with CPU for example one

We see additional error in the explicit method, as our theoretical analysis indicates. However, the error is small compared to the increased computational time for the implicit method. At least for this problem, the explicit method is a better choice of methods.

9.2 Fast Diffusion, Fast Reaction

For this experiment we use the same initial condition, but with different data. The setup for the experiment uses the initial condition defined by Equation (74) with the following data: $\alpha = 1.8$, $(a, b) = (0, 3)$, $[0, T] = [0, 1]$, $V(x, t) = 1$, $D_+(x, t) = D_-(x, t) = .1$, $f(x, t) = 0$ and $S(u(x, t), x, t) = u(x, t)(1 - u(x, t))$.

Figure 6 shows the difference between the two methods. We can see from Table 4 and

Figure 6 that the implicit and explicit are converging to a common solution, as in experiment one.

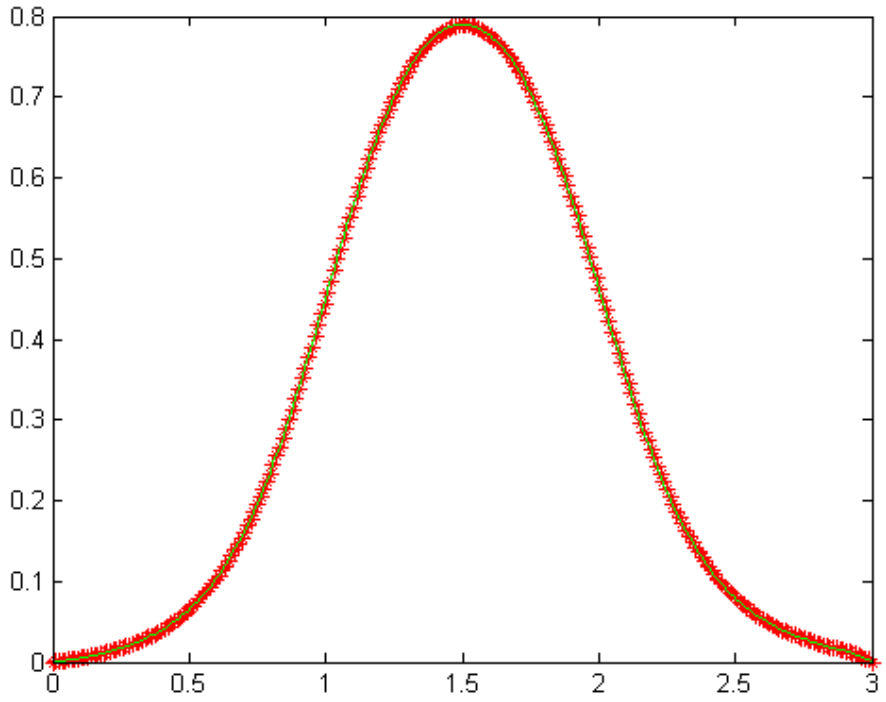

Figure 6: The solutions for implicit and explicit methods for the fast diffusion fast reaction problem at 1024 time steps and 256 spatial steps. The solid green line is the explicit solution and the stars are the implicit solution

We are interested in how quickly each method is converging to this common solution

M	N	L^1	L^2	L^∞	Explicit (s)	Implicit (s)
32	256	1.29E-002	8.50E-003	7.92E-003	.749	2.46
256	1024	1.67E-003	1.10E-003	1.02E-003	22.9	53.9
1024	1024	4.19E-004	2.75E-004	2.55E-004	84.8	192
1024	2048	4.20E-004	2.76E-004	2.56E-004	206	484
2048	2048	2.09E-004	1.37E-004	1.27E-004	374	974

Table 4: Comparison between implicit and explicit methods for various grid sizes with CPU time for example two

solution. We take a sufficiently fine grid so that the difference between the two solutions measured in the L^∞ norm is less than 10^{-5} and let this be our baseline solution. Our the-

oretical analysis indicates that the difference between the methods will be in the temporal grid. So we take a fine spatial grid and examine the error between each method and the base line solution as we take finer temporal grids.

In Tables 5 and 6 we see the performance difference between the methods. We note that

M	N	L^1	L^2	L^∞	Time (s)
16	1024	3.65E-002	2.33E-002	2.04E-002	1.54
32	1024	1.30E-002	8.26E-003	7.51E-003	2.89
64	1024	4.56E-003	2.93E-003	2.71E-003	5.69
128	1024	1.84E-003	1.22E-003	1.13E-003	11.4

Table 5: Difference between explicit and high resolution solutions with 1024 spatial steps and with CPU for example two

M	N	L^1	L^2	L^∞	Time (s)
16	1024	1.87E-002	1.50E-002	1.84E-002	7.88
32	1024	9.63E-003	7.74E-003	9.47E-003	11.4
64	1024	4.89E-003	3.94E-003	4.82E-003	17.2
128	1024	2.48E-003	2.00E-003	2.45E-003	29.8

Table 6: Difference between implicit and high resolution solutions with 1024 spatial steps and with CPU for example two

the implicit method initially has less error than the explicit method, as our theoretical analysis indicates. However, the cost expense of running the implicit method is not worth the reduced error. In this case, the explicit method can be run at 128 temporal steps for the same computation time as 32 temporal steps in the implicit method for about 1/4 of the error.

9.3 Two-Dimensional Numerical Experiment

In this experiment we look at the convergence of the implicit and explicit ADI methods to each other. We will take the following initial condition,

$$u(x,y,0) = \begin{cases} 1 & (x,y) \in [.5,1] \times [.5,1] \\ 0 & (x,y) \notin [.5,1] \times [.5,1] \end{cases}$$

and set parameters for the experiment as follows: $\alpha = 1.8$, $\gamma = 1.6$, $[X_1, X_2] = [0, 3]$, $[Y_1, Y_2] = [0, 3]$, $[0, T] = [0, 1]$, $V(x, y, t) = 1$, $W(x, y, t) = 1$, $D_{x,+} = 0.001$, $D_{x,-} = .001$, $D_{y,+} = 0.001$, $D_{y,-} = 0.001$, $f(x, y, t) = 0$ and $S(u(x, y, t), x, y, t) = 0.1u(x, y, t)(1 - u(x, y, t))$. As with the one dimensional case we will look at the difference between the explicit and implicit methods and compare their computational times. In Figure 7 we see a typical numerical solution. For the tests run, we use the same number of spatial steps in both the x and the y directions.

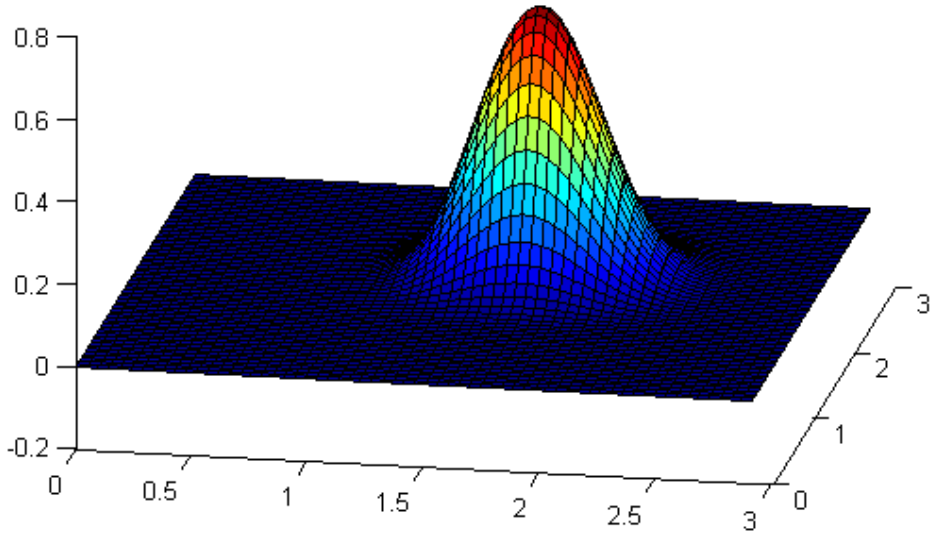

Figure 7: Explicit ADI two dimensional solution with 128 x 128 spatial grid and 128 time steps.

From Table 7 we see that the two solutions are converging to a common solution. We should note the computation times. The ADI methods detailed in this paper are ideal methods for parallelization, since at each step each $N \times N$ system can be solved independently of one other. In order to reduce some of the computational times we implemented a basic parallel code for both the implicit and explicit methods. Times with an asterisk next

M	N	L^1	L^2	L^∞	Explicit* (s)	Explicit (s)	Implicit* (s)	Implicit (s)
64	128	1.34-002	1.67E-002	4.16E-002	6.71	361	71.2	665
128	128	1.00-002	1.17E-002	2.94E-002	13.2	N/A	148	N/A
256	128	9.07-003	1.03E-002	2.80E-002	26.4	824	292	2340
512	128	8.67-003	9.77E-003	2.72E-002	54.4	N/A	477	N/A
1024	128	8.49-003	9.52E-003	2.26E-002	109	3297	1195	9560

Table 7: Difference between explicit and implicit ADI methods for various time steps with 128 steps in x and y. Times with * are computation time for a parallel code implementation.

to them are for the parallel code. We note significant improvement in computation times, especially for the explicit method. However, even without the improvements, the explicit method has much better performance than the implicit method. It is unlikely that any additional error in the explicit method will be offset by the additional computation time of the implicit method.

10 Conclusion

We have shown that current methods for the numerical solutions of space fractional advection-diffusion equations can be extended to equations with non-linear reaction. Using the shifted-Grúnwald approximation with method of characteristics in the implicit Euler method, we present two techniques to solve the non-linear equation. In one case we focused on perturbation error reduction and the other on computational complexity.

In our numerical experiments we have shown that the non-linear reaction term affects the comparative performance of the methods. Both methods converge linearly with space and time, however, in both cases the explicit method had better convergence in terms of computational time. Though there may be cases where the additional error in the explicit method makes the implicit method more favorable, these are likely special cases.

The methods developed for one-dimensional problems can also be extended to two dimensions. Utilizing ADI methods commonly employed in the integer derivative problems we can use the fast iterative algorithms developed for the one-dimensional case. As with

the one-dimensional methods, we proposed two methods; one focusing on reduced error the other on computational complexity. We also demonstrate that the ADI methods are good candidates for parallel code. The improvements are shown to be significant over the serial implementations. However, the additional computational cost of the implicit method make its implementation prohibitive, even with parallel implementation. Therefore, we determine that for the two dimensional case the explicit ADI method is superior to the implicit ADI method.

References

[1] David A. Benson, Mark M. Meerschaert, and Jordan Revielle. Fractional calculus in hydrologic modeling: A numerical perspective. *Advances in Water Resources*, 51:479–497, JAN 2013.

[2] David A. Benson, Stephen W. Wheatcraft, and Mark M. Meerschaert. The fractional-order governing equation of levy motion. *Water Resources Research*, 36(6):1413–1423, 2000.

[3] Raymond H. Chan and Michael K. Ng. Conjugate gradient methods for Toeplitz systems. *SIAM Rev.*, 38(3):427–482, 1996.

[4] Kai Diethelm. *The analysis of fractional differential equations*, volume 2004 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 2010.

[5] Stanley J. Farlow. *Partial differential equations for scientists and engineers*. Dover Publications, Inc., New York, 1993. Revised reprint of the 1982 original.

[6] Gene H. Golub and Charles F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, fourth edition, 2013.

[7] Jim Douglas Jr and Thomas F. Russell. Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference proceedures. *SIAM Journal on Numerical Analysis*, 19(5):871 – 885, 1982.

[8] David Kincaid and Ward Cheney. *Numerical analysis*. Brooks/Cole Publishing Co., Pacific Grove, CA, second edition, 1996.

[9] Siu-Long Lei and Hai-Wei Sun. A circulant preconditioner for fractional diffusion equations. *Journal of Computational Physics*, 242:715–725, JUN 1 2013.

[10] Randall J. LeVeque. *Numerical methods for conservation laws*. Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel, second edition, 1992.

[11] Mark M. Meerschaert and Charles Tadjeran. Finite difference approximations for fractional advection dispersion flow equations. *Journal of Computational and Applied Mathematics*, 172(1):65 – 77, 2004.

[12] Mark M. Meerschaert and Charles Tadjeran. Finite difference approximations for two-sided space-fractional partial differential equations. *Applied Numerical Mathematics*, 56(1):80 – 90, 2006.

[13] MM Meerschaert, HP Scheffler, and C Tadjeran. Finite difference methods for two-dimensional fractional dispersion equation. *Journal of Computational Physics*, 211(1):249–261, JAN 1 2006.

[14] Timothy Moroney and Qianqian Yang. Efficient solution of two-sided nonlinear space-fractional diffusion equations using fast Poisson preconditioners. *Journal of Computational Physics*, 246:304–317, AUG 1 2013.

[15] James M. Ortega. The Newton-Kantorovich theorem. *Amer. Math. Monthly*, 75:658–660, 1968.

[16] Hong-Kui Pang and Hai-Wei Sun. Multigrid method for fractional diffusion equations. *J. Comput. Phys.*, 231(2):693–703, 2012.

[17] Dominik Sierociuk, Andrzej Dzielinski, Grzegorz Sarwas, Ivo Petras, Igor Podlubny, and Tomas Skovranek. Modelling heat transfer in heterogeneous media using fractional calculus. *Philosophical Transactions of the Royal Society A-Mathematical Physical and Engineering Sciences*, 371(1990, SI), MAY 13 2013.

[18] Ercilia Sousa. How to approximate the fractional derivative of order $1 < \alpha \leq 2$. *Internat. J. Bifur. Chaos Appl. Sci. Engrg.*, 22(4):1250075, 13, 2012.

[19] Eitan Tadmor. A review of numerical methods for nonlinear partial differential equations. *Bull. Amer. Math. Soc. (N.S.)*, 49(4):507–554, 2012.

[20] J. A. Tenreiro Machado, Alexandra M. S. F. Galhano, and Juan J. Trujillo. On development of fractional calculus during the last fifty years. *Scientometrics*, 98(1):577–582, JAN 2014.

[21] J. W. Thomas. *Numerical partial differential equations: finite difference methods*, volume 22 of *Texts in Applied Mathematics*. Springer-Verlag, New York, 1995.

[22] William R. Wade. *An Introduction to Analysis*. Prentice Hall, Inc., New Jersey, 3rd edition, 2004.

[23] Hong Wang and Ning Du. Fast alternating-direction finite difference methods for three-dimensional space-fractional diffusion equations. *Journal of Computational Physics*, 258:305–318, FEB 1 2014.

[24] Kaixin Wang and Hong Wang. A fast characteristic finite difference method for fractional advection diffusion equations. *Advances in Water Resources*, 34(7):810 – 816, 2011.

[25] Andrew W. Wharmby and Ronald L. Bagley. Generalization of a theoretical basis for the application of fractional calculus to viscoelasticity. *Journal of Rheology*, 57(5):1429–1440, SEP 2013.

[26] Qianqian Yang, Fawang Liu, and Ian Turner. Stability and convergence of an effective numerical method for the time-space fractional Fokker-Planck equation with a nonlinear source term. *Int. J. Differ. Equ.*, pages Art. ID 464321, 22, 2010.