2014

Second order characteristic based schemes for chemotaxis system

Saleh Albashrawi
Iowa State University

Second order characteristic based schemes for chemotaxis system

by

Saleh A. Albashrawi

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSPHY

Major: Applied Mathematics

Program of Study Committee:

Michael Smiley, Major Professor

Gary Lieberman

Jue Yan

James Rossmanith

Xuan Hien Nguyen

Iowa State University

Ames, Iowa

2014

DEDICATION

I dedicate this work to my family and friends, who have always supported me in every step that I have ever undertaken. In particular, I thank my wife (Ruqayyah) who has supported me and believed in me even when I had doubts myself. I could not have succeeded in my goals without her support.

I would also like to thank the faculty and staff at Iowa State University for their expert guidance and support during my tenure. In particular, I thank Professor Smiley for his great support and appreciated help.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION

It is well understood that organisms respond to chemical signals. This response is called taxis (from the Greek taxis "to arrange"). Typically the word taxis is preceded by a prefix that is determined by the type of stimulus that organisms in a given system respond to. The influence of the chemical signal might be to draw the organism closer to the source of the signal; we typically call this positive chemotaxis, and refer to the chemical as a chemoattractant. When an organism is driven away from the source of a chemical signal we refer to this as negative chemotaxis and in this case we call the chemical species a chemorepellent or a chemoinhibitor.

The significance of the chemotaxis is coming from its applications in biology and clinical pathology. The action of neutrophils is just one example of how the body uses chemotaxis to respond to an infection. A significant clinical target is the altered chemotactic ability of the extracellular or intracellular pathogens. Studying and knowing how to modify the chemotactic ability of theses cellular organisms by medical agents can decrease or prevent the ratio of infections or spreading of infectious diseases.

As the benefits and applications of an understanding of these phenomena are considerable, there is much interest in mathematically modeling them. The most known model for chemotaxis is the Keller-Segel model. A considerable focus of attention and research has been devoted to this model. Keller and Segel [1] suggested a system of the form:

$$u_t = \nabla \cdot (k_1(u,v)\nabla u - k_2(u,v)\nabla v),$$

$$v_t = k_c \Delta v - k_3(v)v + uf(v),$$

which is to hold for $x \in \Omega, t > 0$, for a bounded domain $\Omega \subset R^d$. Here $u(x,t)$ is the population density of an organism under study at a point x and time t, and $v(x,t)$ is the concentration of a chemotactic agent to which the organism responds, that is cells move toward or away from the high concentration region. Neumann boundary conditions were used: $\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = 0$ $(x \in \Omega)$, where $\frac{\partial}{\partial n}$ denotes the outer normal derivative along the boundary $\partial\Omega$ of Ω.

Some interesting mathematical questions arise in the context of the chemotaxis systems. The main issue is the existence and uniqueness of the solution. In particular, a concentration phenomenon may give rise to singularities at accumulation points. This phenomenon is known as the blow-up effect [9, 10, 11, 12]. In his article, Horstmann [2] gave a broad overview of work in this area with extensive bibliographies. The instability of the stationary solution due to large values of the chemotactic sensitivity function under some conditions on the reactive source term is another interesting phenomenon.

In the last few years, many numerical methods have been proposed to solve the chemotaxis systems. Smiley [4] proposed a first-order characteristic based scheme. The constructed scheme belongs to the finite volume method and it has good features such as stability and preserving properties. Some numerical results were obtained for the 2D case on a unit square domain. A. Chertock and A. Kurganov [6] derived a second order finite volume scheme that preserves positivity. The idea of the method is to construct an equivalent system by differentiating the second equation in the Keller-Segel system with respect to x and y and then write the induced system in form of convection-diffusion reaction system. The last form of the equation is solved by using a second-order Godunov-type central-upwind scheme. Numerical results for the 2D case were attached.

Among the finite element method, one can find a method proposed by Siato [13]. In the constructed scheme, upwind techniques were used to satisfy mass and positivity preserving properties . Epshteyn and Kurganov [14, 15] used the discontinuous Galerkin methods to construct a scheme that captured the blow up solution of the chemotaxis system. Their method was based on reformulating the original Keller-Segel system and put it in the form of a convection-diffusion-reaction system with a hyperbolic convective part. Then, interior penalty discontinuous Galerkin methods were introduced to solve the equivalent system. In their article, Budd, Carretero-González and Russell used the moving mesh method [16]. They focused on capturing the blow up solution and stated some numerical results but nothing said about the stability or qualitative properties.

One of the most important role to study the Keller-Segel system is the preserving properties, i.e mass preserving and positivity preserving [6, 13]. It is very important that the numerical scheme solving the Keller-Segel system satisfies these properties. Namely, the positivity preserving property plays important role in the stability of the method.

1.1 The main problem

This work is in the context of improvement of the method proposed in [4]. The method derived in [4] is first order and our goal is to raise it to be second order. We will be considering a simplified version of the Keller-Segel model where the second equation will be independent of the cell density u. Hence, our work will be devoted to deriving a numerical scheme that solve the first equation of the system which is also known as convection diffusion equation.

The combination of diffusion and convection is often found in nature e.g., heat transfer, weather prediction and atmospheric radioactivity propagation. It may also be treated as a simplified model of the system of the Navier-Stokes equations, which are representative equations in fluid dynamics. From a numerical point of view, it is challenging to come up with a numerical scheme solving these type of problems. Whereas the Laplace operator can be discretized by the standard-five point approximation, the convective term has to be discretized with care. The accuracy, numerical stability and boundedness of the approximate solution depends on the numerical scheme used for this term.

Characteristics (-based) methods have been developed for convection-diffusion problems [4, 27, 18, 19, 20]. Ewing and Wang [3], Morton [7] and G. Roos, M. Stynes and L. Tobiska [17] discussed many numerical methods proposed in this area with extensive bibliographies.

The modified method of characteristics (MMOC) was first formulated for advection-diffusion equations by Douglas and Russell [18]. In MMOC, the time derivative is combined with the advection to form a directional derivative along the characteristics. One drawback of this method is that it is not flexible to treat general boundary conditions. Also, it doesn't conserve mass locally. The method derived in [19] which was formulated to solve advection problems, is known to be mass conservative. The same property is satisfied by the characteristic mixed finite element method [20].

1.2 The semi-Lagrangian finite volume method

In this research we focus on a finite volume method that leads to a Eulerian-Lagrangian finite difference scheme. In this method the convection is treated by tracking the trajectories of the particles that pass through the fixed points of an Eulerian grid. These particles are traced back over a single time-step to determine their departure points at the previous time level. The values of the variables at the departure points are generally determined using some form of interpolation based on values at neighbouring grid points, where they are known. Semi-Lagrangian schemes are attractive, because they possess less restrictive stability requirements, and combine the advantages of fixed grids inherent in Eulerian methods.

To put our work in the proper perspective, we mention that the semi-Lagrangian finite volume methods were used for similar problems by many authors. T. N. Phillips and A. J. Williams [21] proposed a method for advection problem written in a conservative form that has been applied in [22, 23] for solving viscoelastic flow problems. More recent works include [4, 24, 25]. A two-grid finite volume element method, combined with the modified method of characteristics was given in [24]. In [25], a symmetric FVE scheme was proposed for nonlinear convection-diffusion problems using strategy of alternating direction.

The difficulties of the method are coming from approximating the integral over the departure cells (i.e cells in the backward time). The sides of the departure cells are curves rather than straight lines. Therefore, this region should be approximated. In their paper, Peter H. Lauritzen, Ramachandran D. Nair and Paul A. Ullrich pointed to different strategies to approximate this region [8]. The one they focused on was based on tracking the intersection of the sides of the departure cells and arrival cells and then integrate over the induced shape. This idea produces irregular shapes. In [21], two methods were proposed to describe the pre-image of the arrival cells (cells at time t_{n+1}). In the first method, the departure point (i.e $x(t_n)$) was assumed to not lie in more than one grid cell away from their locations at time $t = t_{n+1}$. Hence the departure cell will be intersecting with one or more of the neighbouring cells. This restriction on the location of the departure points leads to smaller time step than one may like to use. The second method allows the departure cell to move to any position under one restriction that the departure points stay within the range of the nine neighbour cells. This method allows larger time step than the first method. The intersection between the departure cells and arrival cells is computed in the methods. Smiley [4] used an idea called pseudo characteristic. The principle behind the idea is to assume that the departure cell moves horizontally and then vertically. This will give regular rectangles as a result of intersecting the departure cells with the arrival ones. Therefore, finding the sum of these areas will lead to approximation of the integral in the backward time. In this thesis, we simply approximate the departure cells by connecting the vertices by straight segments and then convert the area integral into line integral. The vertices of the pre-image cells are the approximate solutions of the characteristic equations attached to the first equation of the system.

The cell average of the integrated function was used as an approximation in [21] whereas in [4] Smiley used piecewise constant functions, where the data points were the cell centres, to approximate the exact solution. The derived methods in [4, 21] were only first-order. In this work we used piecewise linear functions in 1D case and piecewise bilinear functions in the 2D case in which the values at the cell centres were given as data. The piecewise linear approximation function was defined on the dual grid cells that defined by the cell centres of the primary grid. The integral (2D case) in the backward time was computed by applying the Divergence Theorem to convert the area integral into line integral.

In their article, Todd F. Dupont and Yingjie Liu used an idea of backward forward error compensation and correction to improve the accuracy of a scheme [28]. The idea can be summarized as follows. One solves the transportation equation under study in forward time by a scheme for one time step then solve the equation using the same scheme in backward time for one time step. Comparing the two solutions gives some information about the error which can be used to improve the accuracy of the scheme used. Using this idea, they succeeded to improve the accuracy and get stable scheme. In our work, we used a similar principle but in a different way. We derive a scheme for the first equation of the chemotaxis system by solving the characteristic equation in backward time and another scheme using the solution of the characteristic equation in forward time. The final scheme is the average of the two schemes.

The schemes derived in this thesis are second order accurate and mass conservative. However, they are conditionally stable. Another concern is the positivity preserving property which we believe is satisfied by the schemes but we have not yet be able to prove it analytically. Also, the derived scheme in 2D case is complicated which makes it hard to implement and harder to analyse. Although the schemes are implicit that is a good feature in similar methods which allow big time steps to be used, it is not applicable in our schemes because of the restriction made to construct the approximating functions (i.e piecewise linear and piecewise bilinear).

The thesis is arranged as follows. In Chapter 2 we derive the second order scheme for one dimensional case followed by proving mass preserving property. An analysis of the consistency is given in Section 2.4 followed by discussion of the stability. Analogously, Chapter 3 contains the scheme and related analysis for a two dimensional system. Numerical experiments are attached in Chapter 4, where we devote Section 4.1 to confirming the theoretical fact stated in Chapter 2. In Subsection 4.1.1 we made some numerical experiment to discuss the positivity preserving property. Section 4.2 was devoted to show the fact already mentioned in Chapter 3. In addition, we did some numerical experiments to discuss the positivity preserving property of the one and two dimensional schemes. In these experiments, we let $v(x, y, t)$ to be known in advance to test certain cases such as when the cell density is clustering at the centre of the domain. There are two appendices attached to the thesis. In Appendix A, we discuss how to find eigenvalues and eigenvectors for tridiagonal matrices. MATLAB codes used in the argument made in Chapter 4 are given in Appendix B.

CHAPTER 2. A 1D SEMI-LAGRANGIAN SCHEME

2.1 One dimensional chemotaxis system

In this chapter we consider a simplified 1D version of Keller-Segel model where the second equation of our system doesn't depend on u. The considered 1D system of the chemotaxis is given by

$$u_t + (uv_x)_x = \kappa u_{xx} + g(x,t), \qquad 0 < x < 1,\ t > 0 \tag{2.1}$$

$$v_t = \sigma v_{xx} - \lambda v + f(x,t), \quad 0 < x < 1,\ t > 0 \tag{2.2}$$

$$u(x,0) = u_0,\ v(x,0) = v_0 \quad x \in \Omega \tag{2.3}$$

$$\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = 0 \qquad on \quad \partial\Omega \tag{2.4}$$

where $\Omega = (0,1)$ and $f(x,t)$ and $g(x,t)$ are a growth/decay terms. In this section we go through a first order scheme proposed by Smiley [4] to solve the above system and then proceed to derive a new scheme in the next section. We are looking for a numerical approximate solution of the system (2.1)-(2.4).

Let M denote the number of subdivisions of the domain, so that

$$\Delta x = \frac{1}{M}, \quad x_i = i\Delta x, \quad i = 0,1,2,...,M.$$

The cells I_i and cell centres $\bar{x}_i$ are defined by

$$I_i = [x_{i-1}, x_i], \qquad \bar{x}_i = \tfrac{x_i - x_{i-1}}{2} = x_{i-\frac{1}{2}},$$

for $i = 1,2,,...,M$. We will consider the cells I_i to be our control volumes whereas the data points are the cell centres $\bar{x}_i$.

It is straightforward to find an approximate solution to (2.2). So we solve (2.2) first and then discuss how to solve (2.1). Using a standard difference scheme, the implicit Euler method, the discrete system for v is given by

$$V_i^{n+1} = V_i^n + \Delta t\{\sigma\left(\frac{V_{i-1}^{n+1} - 2V_i^{n+1} + V_{i+1}^{n+1}}{\Delta x^2}\right) - \lambda V_i^{n+1} + f_i^{n+1}\},$$

where $V_i^n \cong v(\bar{x}_i, t_n)$. By setting $r_\sigma = \frac{\sigma \Delta t}{\Delta x^2}$, the system can be written in the form

$$-r_\sigma V_{i-1}^{n+1} + (1 + 2r_\sigma + \lambda\Delta t)V_i^{n+1} - r_\sigma V_{i+1}^{n+1} = V_i^n + \Delta t f_i^{n+1}, \quad (i = 2 : M-1).$$

The boundary conditions can be approximated by letting $V_0^{n+1} = V_1^{n+1}$ and $V_M^{n+1} = V_{M+1}^{n+1}$ so we have the special cases

$$\begin{aligned} (1 + r_\sigma + \lambda\Delta t)V_1^{n+1} - r_\sigma V_2^{n+1} &= V_1^n + \Delta t f_1^{n+1}, \qquad i = 1 \\ -r_\sigma V_{M-1}^{n+1} + (1 + r_\sigma + \lambda\Delta t)V_M^{n+1} &= V_M^n + \Delta t f_M^{n+1}, \qquad i = M. \end{aligned}$$

Now we proceed to solve equation (2.1). The main goal is to solve the equation (2.1) along the characteristic lines. At each time step, a discrete set of particles at grid points are tracked backward over a single time step along characteristics to their departure points. The characteristics associated with the equation (2.1) are solutions of ordinary equations

$$\frac{dx}{dt} = v_x(x(t), t), \qquad x(t_{n+1}) = x_i \tag{2.5}$$

where x_i is the arrival point. The characteristic curves determine regions

$$Q_i^{n+1} = \{(x,t) \in (0,1) \times (t_n, t_{n+1}) : x_{i-1}(t) < x < x_i(t), t \in (t_n, t_{n+1})\},$$

(Figure 2.1). These regions are coupled together to cover $(0,1) \times (t_n, t_{n+1})$. Integrating (2.1) over an interior region Q_i^{n+1} and using the Divergence Theorem we get

$$\begin{aligned} \iint\limits_{Q_i^{n+1}} \kappa u_{xx} = \iint_{Q_i^{n+1}} (u_t + (uv_x)_x)dxdt &= \int_{\partial Q_i^{n+1}} u < v_x, 1 > \cdot \vec{n} ds \\ &= \int\limits_{x_{i-1}}^{x_i} u(x, t_{n+1})dx - \int\limits_{x_{i-1}(t_n)}^{x_i(t_n)} u(x, t_n)dx \end{aligned} \tag{2.6}$$

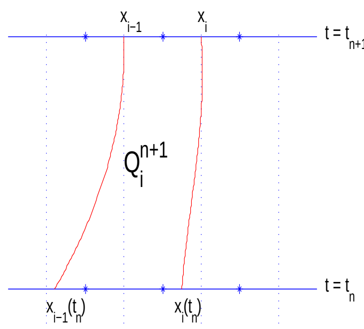

Figure 2.1 The region Q_i^{n+1} determined by the characteristics curves.

where $\vec{n}$ denotes the outward unit normal to the boundary ∂Q_i^{n+1} of Q_i^{n+1}. The right hand side of (2.6) is a consequence of the fact that the characteristic directions $\langle v_x, 1\rangle$ are tangent to characteristic curves represented by ∂Q_i^{n+1}. In the following, we are discussing how to approximate this integral identity.

We consider approximate solutions $U(x,t)$ that are piecewise constant functions with constant values on the cells. Thus

$$u(x,t) \approx U(x,t) = U_i^n, \qquad x_{i-1} < x < x_i,\ t = t_n.$$

Then

$$\frac{1}{\Delta x}\int\limits_{x_{i-1}}^{x_i} u(x,t_{n+1})dx \approx \frac{1}{\Delta x}\int\limits_{x_{i-1}}^{x_i} U(x,t_{n+1})dx = U_i^{n+1}.$$

For convenience we will write $\tilde{x}_i = x_i(t_n)$. Assuming $\tilde{x}_i \in [x_{i-1}, x_{i+1}]$ for each i, then the integral over $[\tilde{x}_{i-1}, \tilde{x}_i]$ will be broken into pieces depending on which cell $\tilde{x}_{i-1}$ and $\tilde{x}_i$ lie in. So

$$\begin{aligned}\int\limits_{\tilde{x}_{i-1}}^{\tilde{x}_i} U(x,t_n)dx &= \sum_{k=-1}^{1} |(x_{i+k-1}, x_{i+k}) \cap (\tilde{x}_{i-1}, \tilde{x}_i)| U_{i+k}^n \\ &= |(x_{i-2}, x_{i-1}) \cap (\tilde{x}_{i-1}, \tilde{x}_i)| U_{i-1}^n + |(x_{i-1}, x_i) \cap (\tilde{x}_{i-1}, \tilde{x}_i)| U_i^n \\ &+ |(x_i, x_{i+1}) \cap (\tilde{x}_{i-1}, \tilde{x}_i)| U_{i+1}^n.\end{aligned} \tag{2.7}$$

In order to approximate the above interval length, we use the identity

$$
\begin{aligned}
|(a,b)\cap(a+\Delta a, b+\Delta b)| &= \max\{0, \min\{b, b+\Delta b\} - \max\{a, a+\Delta a\}\} \\
&= \max\{0, b-a+\min\{0,\Delta b\} - \max\{0,\Delta a\}\}.
\end{aligned}
\tag{2.8}
$$

Integrating (2.5) over the subinterval $[t_n, t_{n+1}]$ and dividing by $\varDelta x$ to define the normalized distance we get

$$
\theta_i = \frac{\tilde{x}_i - x_i}{\varDelta x} = -\frac{1}{\varDelta x}\int_{t_n}^{t_{n+1}} v_x(x(s), s)ds \approx -\frac{\Delta t}{\Delta x} v_x(x(t_{n+1}), t_{n+1}), \tag{2.9}
$$

where the integral was approximated by the right end quadrature rule. Approximating $v_x(x(t_{n+1}), t_{n+1}) = v_x(x_i, t_{n+1})$ by $\frac{V_{i+1}^{n+1} - V_i^{n+1}}{\varDelta x}$, we then have

$$
\theta_i \approx \Theta_i = -\frac{\varDelta t}{\varDelta x^2}(V_{i+1}^{n+1} - V_i^{n+1}), \qquad i = 1 : M-1.
$$

By the definition (2.9), $\tilde{x}_i = x_i + \theta_i \Delta x = x_i + \Delta x + \theta_i \Delta x - \Delta x$, from which we get

$$
\tilde{x}_i = x_{i+k} + (\theta_i - k)\Delta x, \qquad k = -1, 0, 1.
$$

Hence, the interval lengths on the right side of (2.8), under the assumption $|\Theta_i| \leq \frac{1}{2}$ for all i, can be written as

$$
\begin{aligned}
|(x_{i-2}, x_{i-1}) \cap (\tilde{x}_{i-1}, \tilde{x}_i)| &\approx |(x_{i-2}, x_{i-1}) \cap (x_{i-2} + (\Theta_i + 1)\Delta x, x_{i-1} + (\Theta_i + 1)\Delta x)| \\
&= \max\{0, 1 + \min\{0, \Theta_i + 1\} - \max\{0, \Theta_{i-1} + 1\}\}\Delta x \\
&= -\min\{0, \Theta_{i-1}\}\Delta x, \\
|(x_{i-1}, x_i) \cap (\tilde{x}_{i-1}, \tilde{x}_i)| &\approx |(x_{i-1}, x_i) \cap (x_{i-1} + (\Theta_i + 1)\Delta x, x_i + (\Theta_{i-1} + 1)\Delta x)| \\
&= \max\{0, 1 + \min\{0, \Theta_i\} - \max\{0, \Theta_{i-1}\}\}\Delta x \\
&= (1 + \min\{0, \Theta_i\} - \max\{0, \Theta_{i-1}\})\Delta x, \\
|(x_i, x_{i+1}) \cap (\tilde{x}_{i-1}, \tilde{x}_i)| &\approx |(x_i, x_{i+1}) \cap (x_i + (\Theta_i - 1)\Delta x, x_i + (\Theta_i - 1)\Delta x)| \\
&= \max\{0, 1 + \min\{0, \Theta_i - 1\} - \max\{0, \Theta_{i-1} - 1\}\}\Delta x \\
&= \max\{0, \Theta_i\}\Delta x.
\end{aligned}
\tag{2.10}
$$

By setting

$$a_i = \min\{0, \Theta_i\} \quad \text{and} \quad A_i = \max\{0, \Theta_i\}$$
$$a_0 = A_0 = a_M = A_M = 0$$

the equation (2.7) can be written as

$$\frac{1}{\Delta x}\int_{\tilde{x}_{i-1}}^{\tilde{x}_i} U(x, t_n)dx = -a_{i-1}U_{i-1}^n + (1 + a_i - A_{i-1})U_i^n + A_i U_{i+1}^n, \qquad 1 \le i \le M.$$

Finally, the diffusion term is approximated by a standard finite difference approximation. Using centred difference approximation of the derivative and right end quadrature rule we get

$$\frac{1}{\Delta x}\int_{t_n}^{t_{n+1}} \kappa u_{xx}(x_i(t), t)dt \approx \frac{\kappa \Delta t}{(\Delta x)^2}(U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i-1}^{n+1}).$$

From all of these approximations, the discrete system for U as $(i = 2 : M - 1)$ is given by

$$U_i^{n+1} = -a_{i-1}U_{i-1}^n + (1 + a_i - A_{i-1})U_i^n + A_i U_{i+1}^n + \frac{\kappa \Delta t}{(\Delta x)^2}(U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1}). \quad (2.11)$$

The system can be written in a compact form as

$$-r_\kappa U_{i-1}^{n+1} + (1 + 2r_\kappa)U_i^{n+1} - r_\kappa U_{i+1}^{n+1} = U_i^n + F_i^n - F_{i-1}^n, \qquad i = 2 : M - 1.$$
$$(1 + r)U_1^{n+1} - rU_2^{n+1} = U_1^n + F_1^n \ , \qquad i = 1$$
$$-rU_{M-1}^{n+1} + (1 + r)U_M^{n+1} = U_M^n - F_{M-1}^n, \qquad i = M$$

where $F_i^n = a_i U_i^n + A_i U_{i+1}^n$ $(i = 1 : M - 1)$, and $r_\kappa = \frac{\kappa \Delta t}{(\Delta x)^2}$.

The scheme (2.11) has good properties such as mass conserving and non-negativity preserving. However, it is first order and our purpose is to derive a second order scheme. Since the characteristic equations depend on the values of v, improving these values will give higher accuracy for the approximated departure points $\tilde{x}_i$. To achieve that we are going to use the Crank- Nicolson scheme to solve the v equation. Hence we have

$$V_i^{n+1} = V_i^n + \frac{1}{2}r_\sigma\{(V_{i-1}^{n+1} - 2V_i^{n+1} + V_{i+1}^{n+1}) + (V_{i-1}^n - 2V_i^n + V_{i+1}^n)\}$$
$$+ \frac{1}{2}\Delta t\{-\lambda(V_i^{n+1} + V_i^n) + (f_i^{n+1} + f_i^n)\}.$$

2.2 Derivation of the method

In the above scheme it was assumed that u is approximated by piecewise constant functions. To get higher order we are going to use piecewise linear functions as approximation of u on the grid cells; so we let

$$u(x,t_n) \approx U(x,t_n) = \begin{cases} U_{i-1}^n + \frac{x-\bar{x}_{i-1}}{\Delta x}(U_i^n - U_{i-1}^n), & \bar{x}_{i-1} < x < \bar{x}_i \\ U_i^n + \frac{x-\bar{x}_i}{\Delta x}(U_{i+1}^n - U_i^n), & \bar{x}_i < x < \bar{x}_{i+1} \end{cases}$$

where $U_i^n = u(\bar{x}_i, t_n)$.

Suppose that $\bar{x}_i \in [\tilde{x}_{i-1}, \tilde{x}_i]$ then

$$\int_{\tilde{x}_{i-1}}^{\tilde{x}_i} u(x,t_n)dx \approx \int_{\tilde{x}_{i-1}}^{\tilde{x}_i} U(x,t_n)dx = \int_{\tilde{x}_{i-1}}^{\bar{x}_i} U(x,t_n)dx + \int_{\bar{x}_i}^{\tilde{x}_i} U(x,t_n)dx.$$

In the following computations, we are setting $x_i(t) = x(t)$. Since $U(x,t_n)$ is a piecewise linear approximation, then by using the trapezoidal rule we obtain the following exact integrations

$$\begin{aligned} \int_{\tilde{x}_{i-1}}^{\bar{x}_i} U(x,t_n)dx &= \tfrac{1}{2}(\bar{x}_i - \tilde{x}_{i-1})[U(\tilde{x}_{i-1},t_n) + U(\bar{x}_i,t_n)] \\ &= \tfrac{1}{2}(\bar{x}_i - \tilde{x}_{i-1})[U_{i-1}^n + \frac{\tilde{x}_{i-1} - \bar{x}_{i-1}}{\Delta x}(U_i^n - U_{i-1}^n) + U_i^n], \\ \int_{\bar{x}_i}^{\tilde{x}_i} U(x,t_n)dx &= \tfrac{1}{2}(\tilde{x}_i - \bar{x}_i)[U(\bar{x}_i,t_n) + U(\tilde{x}_i,t_n)] \\ &= \tfrac{1}{2}(\tilde{x}_i - \bar{x}_i)[2U_i^n + \frac{\tilde{x}_i - \bar{x}_i}{\Delta x}(U_{i+1}^n - U_i^n)]. \end{aligned} \tag{2.12}$$

By the definition (2.9), we get $\frac{\tilde{x}_i - \bar{x}_i}{\Delta x} = \theta_i + \frac{1}{2} \approx \Theta_i + \frac{1}{2}$, from which we obtain

$$\bar{x}_i - \tilde{x}_{i-1} = x_{i-1} + \tfrac{1}{2}\Delta x - \tilde{x}_{i-1} \approx \Delta x(\tfrac{1}{2} - \Theta_{i-1}).$$

Similarly, the other coefficients in (2.12) can be written in terms of Θ_i and Θ_{i-1} . Hence, (2.12) can be simplified as (we dropped the superscript n for simplicity)

$$\int_{\tilde{x}_{i-1}}^{\tilde{x}_i} U(x,t_n)dx = \tfrac{1}{2}\Delta x(\tfrac{1}{2}-\Theta_{i-1})[(\tfrac{1}{2}-\Theta_{i-1})U_{i-1}+(\tfrac{3}{2}+\Theta_{i-1})U_i]$$

$$\int_{\tilde{x}_i}^{\tilde{x}_i} U(x,t_n)dx = \tfrac{1}{2}\Delta x(\tfrac{1}{2}+\Theta_i)[(\tfrac{3}{2}-\Theta_i)U_i+(\tfrac{1}{2}+\Theta_i)U_{i+1}].$$

By adding these two equations, we get

$$\begin{aligned}\int_{\tilde{x}_{i-1}}^{\tilde{x}_i} U(x,t_n)dx &= \tfrac{1}{2}\Delta x[(\tfrac{1}{2}-\Theta_{i-1})^2U_{i-1}+(\tfrac{1}{2}-\Theta_{i-1})(\tfrac{3}{2}+\Theta_{i-1})U_i\\ &+(\tfrac{1}{2}+\Theta_i)(\tfrac{3}{2}-\Theta_i)U_i+(\tfrac{1}{2}+\Theta_i)^2U_{i+1}]\\ &= \tfrac{1}{2}\Delta x[(\tfrac{1}{4}-\Theta_{i-1}+\Theta_{i-1}^2)U_{i-1}+(\tfrac{3}{4}-\Theta_{i-1}-\Theta_{i-1}^2)U_i\\ &+(\tfrac{3}{4}+\Theta_i-\Theta_i^2)U_i+(\tfrac{1}{4}+\Theta_i+\Theta_i^2)u_{i+1}]\\ &= \tfrac{1}{2}\Delta x[-\Theta_{i-1}(1-\Theta_{i-1})U_{i-1}-\Theta_{i-1}(1+\Theta_{i-1})U_i+\tfrac{1}{4}U_{i-1}\\ &+\tfrac{6}{4}U_i+\tfrac{1}{4}U_{i+1}+\Theta_i(1-\Theta_i)U_i+\Theta_i(1+\Theta_i)U_{i+1}].\end{aligned}$$

By setting

$$a_i = \Theta_i(1-\Theta_i), \quad A_i = \Theta_i(1+\Theta_i) \tag{2.13}$$

we get

$$\begin{aligned}\int_{\tilde{x}_{i-1}}^{\tilde{x}_i} u(x,t_n)dx \approx \int_{\tilde{x}_{i-1}}^{\tilde{x}_i} U(x,t_n)dx &= \frac{1}{2}\Delta x[(a_iU_i^n+A_iU_{i+1}^n)-(a_{i-1}U_{i-1}^n+A_{i-1}U_i^n)]\\ &+\frac{1}{8}\Delta x(U_{i-1}^n+6U_i^n+U_{i+1}^n).\end{aligned} \tag{2.14}$$

If the departure point $\tilde{x}_i$ coincides with the grid point($\tilde{x}_i = x_i$ and $\tilde{x}_{i-1} = x_{i-1}$), then $\Theta_i = \Theta_{i-1} = 0 \Rightarrow a_{i-1} = A_{i-1} = a_i = A_i = 0$, that implies:

$$\int_{\tilde{x}_{i-1}}^{\tilde{x}_i} u(x,t_n)dx \approx \int_{x_{i-1}}^{x_i} U(x,t_n)dx = \frac{1}{8}\Delta x(U_{i-1}^n+6U_i^n+U_{i+1}^n),$$

which gives a quadrature rule for a piecewise linear function over a fixed grid cell.

Remark 2.1: *We will be using a different approximation for θ_i from the one used in the previous section as we show. Integrating (2.9) over $[t_n, t_{n+1}]$ and using the trapezoidal rule to have*

$$\theta_i = \frac{x(t_n) - x_i}{\Delta x} = -\frac{1}{\Delta x}\int_{t_n}^{t_{n+1}} v_x(x(s), s)ds \approx -\frac{\Delta t}{2\Delta x}\big[v_x(x(t_n), t_n) + v_x(x(t_{n+1}), t_n)\big].$$

By approximating the derivative in the following way $v_x(x(t_n), t_n) \approx v_x(x_i, t_n)$ and writing $v_x(x_i, t_n) \approx \frac{V_i^n - V_{i-1}^n}{\Delta x}$, we obtain

$$\theta_i \approx \Theta_i = -\frac{1}{2}\frac{\Delta t}{(\Delta x)^2}\big[(V_i^{n+1} - V_{i-1}^{n+1}) + (V_i^n - V_{i-1}^n)\big]. \tag{2.15}$$

Finally, to complete the scheme the diffusion term must be approximated. Approximating the integral by the trapezoidal rule and applying a centred difference approximation for the derivative give

$$\iint_{Q_i^{n+1}} \kappa u_{xx}(x(t), t) \approx \frac{1}{2}\frac{\kappa\Delta t}{(\Delta x)^2}\big[(U_{i-1}^n - 2U_i^n + U_{i+1}^n) + (U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1})\big]. \tag{2.16}$$

Combining all of these approximations, the scheme can be written in a compact form as

$$\begin{aligned} \frac{1}{8}(U_{i-1}^{n+1} + 6U_i^{n+1} + U_{i+1}^{n+1}) &= \frac{1}{8}(U_{i-1}^n + 6U_i^n + U_{i+1}^n) + \frac{1}{2}(F_i^n - F_{i-1}^n) \\ &+ \frac{1}{2}r_\kappa[\delta_i^{n+1}(U) - \delta_{i-1}^{n+1}(U)] + \frac{1}{2}r_\kappa[\delta_i^n(U) - \delta_{i-1}^n(U)], \end{aligned} \tag{2.17}$$

where $r_\kappa = \frac{\kappa\Delta t}{(\Delta x)^2}$, $\delta_i^n(U) = U_{i+1}^n - U_i^n$, and $F_i^n = a_i U_i^n + A_i U_{i+1}^n$. This scheme is valid for the interior points (i.e. $i = 2, ..., M-1$). The special cases ($i = 1, M$) are given by

$$\begin{aligned} \frac{1}{8}(7U_1^{n+1} + U_2^{n+1}) &= \frac{1}{8}(7U_1^n + U_2^n) + \frac{1}{2}F_1^n + \frac{1}{2}r_\kappa[\delta_1^n(U) + \delta_1^{n+1}(U)], \\ \frac{1}{8}(7U_M^{n+1} + U_{M-1}^{n+1}) &= \frac{1}{8}(7U_M^n + U_{M-1}^n) - \frac{1}{2}F_{M-1}^n + \frac{1}{2}r_\kappa[\delta_{M-1}^n(U) + \delta_{M-1}^{n+1}(U)]. \end{aligned}$$

The scheme (2.17) lacks to symmetry. Also, our numerical experiment shows that the scheme is only first order. To recover the symmetry and hence achieve the second order accuracy we reverse the above process and look for the characteristics in the forward time . Then we take the average of the two schemes. So we are solving the differential equation

$$\frac{dx}{dt} = v_x(x(t), t), \qquad x(t_n) = x_i. \tag{2.18}$$

By the trapezoidal rule, we get

$$x(t_{n+1}) - x(t_n) = \int_{t_n}^{t_{n+1}} v_x(x(t), t)dt \approx \frac{1}{2}\Delta t[v_x(x(t_n), t_n) + v_x(x(t_{n+1}), t_{n+1})].$$

Setting $\tilde{\theta}_i = \frac{x(t_{n+1})-x_i}{\Delta x}$ and let $\tilde{\Theta}_i$ be its approximation, then we have that $\tilde{\Theta}_i = -\Theta_i$. Let $\tilde{F}_i$ be the flux in the forward time, then by the same argument we have

$$\tilde{F}_i = \tilde{a}_i U_i^{n+1} + \tilde{A}_i U_{i+1}^{n+1} = -A_i U_i^{n+1} - a_i U_{i+1}^{n+1},$$

where $\tilde{a}_i = \tilde{\Theta}_i(1 - \tilde{\Theta}_i) = -A_i$, and $\tilde{A}_i = \tilde{\Theta}_i(1 + \tilde{\Theta}_i) = -a_i$. So, the scheme in the forward time is given by

$$\begin{aligned} &\frac{1}{8}(U_{i-1}^{n+1} + 6U_i^{n+1} + U_{i+1}^{n+1}) + \frac{1}{2}(\tilde{F}_i^{n+1} - \tilde{F}_{i-1}^{n+1}) \\ &\quad= \frac{1}{8}(U_{i-1}^{n} + 6U_i^{n} + U_{i+1}^{n}) + \frac{1}{2}r_\kappa[\delta_i^{n+1}(U) - \delta_{i-1}^{n+1}(U)] + \frac{1}{2}r_\kappa[\delta_i^{n}(U) - \delta_{i-1}^{n}(U)]. \end{aligned} \tag{2.19}$$

By taking the average of these two schemes (2.17) and (2.19) we have

$$\begin{aligned} &\frac{1}{8}(U_{i-1}^{n+1} + 6U_i^{n+1} + U_{i+1}^{n+1}) + \frac{1}{4}(\tilde{F}_i^{n+1} - \tilde{F}_{i-1}^{n+1}) - \frac{1}{2}r_\kappa[\delta_i^{n+1}(U) - \delta_{i-1}^{n+1}(U)] \\ &\quad= \frac{1}{8}(U_{i-1}^{n} + 6U_i^{n} + U_{i+1}^{n}) + \frac{1}{4}(F_i^{n} - F_{i-1}^{n}) + \frac{1}{2}r_\kappa[\delta_i^{n}(U) - \delta_{i-1}^{n}(U)]. \end{aligned} \tag{2.20}$$

If we adopt the convention that $F_0^n = F_M^n = \tilde{F}_0^{n+1} = \tilde{F}_M^{n+1} = 0$ and $\delta_0^n(U) = \delta_M^n(U) = \delta_0^{n+1}(U) = \delta_M^{n+1}(U) = 0$, then this equation holds for $i = 1, M$.

2.3 Mass preserving property

It is very important for a numerical method that it satisfies the solution properties of the pde i.e mass preserving property and non-negative preserving property. So, we show that in absence of sources or sinks the scheme (2.20), under the assumption that all the values are positive, is mass preserving by showing that the locally conservative form of the scheme leads to telescoping sums. Recall that the boundary conditions are homogeneous Neumann which indicates that the boundary points will be approximated in a different way (i.e piecewise constant). Hence, we have $U_0 = U_1$ and $U_{M+1} = U_M$ for all time levels.

Lemma 2.1: If $\{U_i^{n+1}\}$ is the solution of the scheme (2.20), given the data $\{U_i^n\}_{i=1}^M$, then

$$\sum_{i=1}^{M} U_i^{n+1} = \sum_{i=1}^{M} U_i^n. \tag{2.21}$$

Proof: We get this from the conservative form of the scheme. We sum the terms on the right side of (2.20) to get

$$\begin{aligned}
&\sum_{i=1}^{M}\frac{1}{8}(U_{i-1}^n + 6U_i^n + U_{i+1}^n) + \sum_{i=1}^{M}\frac{1}{4}(F_i^n - F_{i-1}^n) + \sum_{i=1}^{M}\frac{1}{2}r(\delta_i^n(U) - \delta_{i-1}^n(U)) \\
&\quad = \sum_{i=1}^{M}\frac{1}{8}(U_{i-1}^n + 6U_i^n + U_{i+1}^n) + \frac{1}{4}(F_M^n - F_0^n) + \frac{1}{2}r(\delta_M^n(U) - \delta_0^n(U)) \\
&\quad = \sum_{i=1}^{M}\frac{1}{8}(U_{i-1}^n + 6U_i^n + U_{i+1}^n) = \sum_{i=1}^{M} U_i^n.
\end{aligned}$$

Similarly we get the left hand side. So

$$\sum_{i=1}^{M} U_i^{n+1} = \sum_{i=1}^{M} U_i^n \qquad \square$$

2.4 Consistency

In this section we are going to discuss the consistency of the scheme. Let $u(x,t)$ be a smooth solution of (2.1) and let $u_i^n = u(\bar{x}_i, t_n)$ denote its values at the space-time nodal points $(\bar{x}_i, t_n)$. The local truncation error τ_i^{n+1} is defined by

$$\tau_i^{n+1} = \frac{1}{8k}\left[(u_{i-1}^{n+1} + 6u_i^{n+1} + u_{i+1}^{n+1}) - (u_{i-1}^{n} + 6u_i^{n} + u_{i+1}^{n})\right]$$
$$+ \frac{1}{4k}\left[(\tilde{F}_i^{n+1} - \tilde{F}_{i-1}^{n+1}) - (F_i^n - F_{i-1}^n)\right] - \frac{\kappa}{2h^2}\left[(\delta_i^n(u) - \delta_{i-1}^n(u) + (\delta_i^{n+1}(u) - \delta_{i-1}^{n+1}(u))\right] \tag{2.22}$$

where $h = \Delta x$, $k = \Delta t$.

In the right side of following expansions we are writing $u = u_i^n = u(\bar{x}_i, t_n)$. Similarly, all the derivatives are evaluated at the point $(\bar{x}_i, t_n)$ and we are writing $u_x = u_x(\bar{x}_i, t_n)$. By the Taylor expansion about the point $(\bar{x}_i, t_n)$, we have

$$\begin{aligned}\frac{1}{8k}(u_{i-1}^n + 6u_i^n + u_{i+1}^n) &= \frac{1}{8k}[u - hu_x + \frac{1}{2}h^2u_{xx} - \frac{1}{6}h^3u_{xxx}\\ &\quad + 6u + u + hu_x + \frac{1}{2}h^2u_{xx} + \frac{1}{6}h^3u_{xxx} + O(h^4)]\\ &= \frac{1}{8k}[8u + h^2u_{xx} + O(h^4)].\end{aligned} \tag{2.23}$$

Expanding the quadrature term at time t_{n+1} gives

$$\begin{aligned}&\frac{1}{8k}(u_{i-1}^{n+1} + 6u_i^{n+1} + u_{i+1}^{n+1})\\ &\quad= \frac{1}{8k}[u - hu_x + ku_t + \frac{1}{2}h^2u_{xx} - hku_{xt} + \frac{1}{2}k^2u_{tt} - \frac{1}{6}h^3u_{xxx} + \frac{1}{2}h^2ku_{xxt} - \frac{1}{2}hk^2u_{xtt}\\ &\quad+ \frac{1}{6}k^3u_{ttt} + 6u + 6ku_t + 3k^2u_{tt} + k^3u_{ttt}\\ &\quad+ u + hu_x + ku_t + \frac{1}{2}h^2u_{xx} + hku_{xt} + \frac{1}{2}k^2u_{tt} + \frac{1}{6}h^3u_{xxx} + \frac{1}{2}h^2ku_{xxt} + \frac{1}{2}hk^2u_{xtt}\\ &\quad+ \frac{1}{6}k^3u_{ttt} + O(h^4) + O(k^4)]\\ &\quad= \frac{1}{8k}[8u + 8ku_t + h^2u_{xx} + 4k^2u_{tt} + h^2ku_{xxt} + \frac{8}{6}k^3u_{ttt} + O(h^4) + O(k^4)]\end{aligned} \tag{2.24}$$

The expansions of the diffusion terms are given by

$$\begin{aligned}\frac{\kappa}{2h^2}(\delta_i^n(u) - \delta_{i-1}^n(u)) &= \frac{\kappa}{2h^2}[u - hu_x + \frac{1}{2}h^2u_{xx} - \frac{1}{6}h^3u_{xxx} + \frac{1}{24}h^4u_{xxxx}\\ &\quad - 2u + u + hu_x + \frac{1}{2}h^2u_{xx} + \frac{1}{6}h^3u_{xxx} + \frac{1}{24}h^4u_{xxxx} + O(h^5)]\\ &= \frac{\kappa}{2h^2}[h^2u_{xx} + \frac{1}{12}h^4u_{xxxx} + O(h^5)],\end{aligned} \tag{2.25}$$

and

$$
\begin{aligned}
&\frac{\kappa}{2h^2}(\delta_i^{n+1}(u) - \delta_{i-1}^{n+1}(u)) \\
&\qquad = \frac{\kappa}{2h^2}[u - hu_x + ku_t + \frac{1}{2}h^2u_{xx} - hku_{xt} + \frac{1}{2}k^2u_{tt} - \frac{1}{6}h^3u_{xxx} + \frac{1}{2}h^2ku_{xxt} \\
&\qquad - \frac{1}{2}hk^2u_{xtt} + \frac{1}{6}k^3u_{ttt} + \frac{1}{24}h^4u_{xxxx} - \frac{1}{6}h^3ku_{xxxt} + \frac{1}{4}h^2k^2u_{xxtt} - \frac{1}{6}hk^3u_{xttt} \\
&\qquad + \frac{1}{24}k^4u_{tttt} - 2u - 2ku_t - k^2u_{tt} - \frac{1}{3}k^3u_{ttt} - \frac{1}{12}k^4u_{tttt} \\
&\qquad + u + hu_x + ku_t + \frac{1}{2}h^2u_{xx} + hku_{xt} + \frac{1}{2}k^2u_{tt} + \frac{1}{6}h^3u_{xxx} + \frac{1}{2}h^2ku_{xxt} + \frac{1}{2}hk^2u_{xtt} \\
&\qquad + \frac{1}{6}k^3u_{ttt} + \frac{1}{24}h^4u_{xxxx} + \frac{1}{6}h^3ku_{xxxt} + \frac{1}{4}h^2k^2u_{xxtt} + \frac{1}{6}hk^3u_{xttt} + \frac{1}{24}k^4u_{tttt} \\
&\qquad + O(h^5) + O(k^5)] \\
&\qquad = \frac{\kappa}{2h^2}[h^2u_{xx} + h^2ku_{xxt} + \frac{1}{12}h^4u_{xxxx} + \frac{1}{2}h^2k^2u_{xxtt} + O(h^5) + O(k^5)].
\end{aligned} \tag{2.26}
$$

The expansions of the fluxes are more involved. These expansions will be given by

$$
\begin{aligned}
&\frac{1}{4k}(F_i^n - F_{i-1}^n) = \frac{1}{4k}[-a_{i-1}u_{i-1}^n + (a_i - A_{i-1})u_i^n + A_iu_{i+1}^n] \\
&\quad = \frac{1}{4k}\big[- a_{i-1}(u - hu_x + \frac{1}{2}h^2u_{xx} - \frac{1}{6}h^3u_{xxx}) + (a_i - A_{i-1})u \\
&\quad + A_i(u + hu_x + \frac{1}{2}h^2u_{xx} + \frac{1}{6}h^3u_{xxx}) + O(h^4)\big] \\
&\quad = \frac{1}{4k}\big[(-a_{i-1} + a_i - A_{i-1} + A_i)u + (a_{i-1} + A_i)hu_x + \frac{1}{2}(-a_{i-1} + A_i)h^2u_{xx} \\
&\quad + \frac{1}{6}(a_{i-1} + A_i)h^3u_{xxx} + O(h^4)\big],
\end{aligned} \tag{2.27}
$$

and

$$
\begin{aligned}
\frac{1}{4k}(\tilde{F}_i^{n+1} - \tilde{F}_{i-1}^{n+1}) &= \frac{1}{4k}(A_{i-1}u_{i-1}^{n+1} + (a_{i-1} - A_i)u_i^{n+1} - a_i u_{i+1}^{n+1}) \\
&= \frac{1}{4k}\Big[A_{i-1}(u - hu_x + ku_t + \frac{1}{2}h^2u_{xx} - hku_{xt} + \frac{1}{2}k^2u_{tt} - \frac{1}{6}h^3u_{xxx} + \frac{1}{2}h^2ku_{xxt} \\
&- \frac{1}{2}hk^2u_{xtt} + \frac{1}{6}k^3u_{ttt}) + (a_{i-1} - A_i)(u + ku_t + \frac{1}{2}k^2u_{tt} \\
&+ \frac{1}{6}k^3u_{ttt}) - a_i(u + hu_x + ku_t + \frac{1}{2}h^2u_{xx} + hku_{xt} + \frac{1}{2}k^2u_{tt} + \frac{1}{6}h^3u_{xxx} + \frac{1}{2}h^2ku_{xxt} \\
&+ \frac{1}{2}hk^2u_{xtt} + \frac{1}{6}k^3u_{ttt}) + O(h^4) + O(k^4)\Big] \\
&= \frac{1}{4k}[(A_{i-1} + a_{i-1} - A_i - a_i)u - (A_{i-1} + a_i)hu_x + (A_{i-1} + a_{i-1} - A_i - a_i)ku_t \\
&+ \frac{1}{2}(A_{i-1} - a_i)h^2u_{xx} + \frac{1}{2}(A_{i-1} + a_{i-1} - A_i - a_i)k^2u_{tt} - (A_{i-1} + a_i)hku_{xt} \\
&- \frac{1}{6}(A_{i-1} + a_i)h^3u_{xxx} + \frac{1}{2}(A_{i-1} - a_i)h^2ku_{xxt} \\
&- \frac{1}{2}(A_{i-1} + a_i)hk^2u_{xtt} + \frac{1}{6}(A_{i-1} + a_{i-1} - A_i - a_i)k^3u_{ttt} \\
&+ O(h^4) + O(k^4)].
\end{aligned}
\tag{2.28}
$$

Subtracting (2.23) from (2.24) gives

$$
\begin{aligned}
\frac{1}{8k}\big[(u_{i-1}^{n+1} + 6u_i^{n+1} + u_{i+1}^{n+1}) - (u_{i-1}^n + 6u_i^n + u_{i+1}^n)\big] &= u_t + \frac{1}{2}ku_{tt} + \frac{h^2}{8}u_{xxt} + \frac{1}{6}k^2u_{ttt} \\
&+ O(h^3) + O(k^3).
\end{aligned}
\tag{2.29}
$$

Now, the addition of (2.25) and (2.26) gives

$$
\begin{aligned}
&\frac{\kappa}{2h^2}\big[(\delta_i^n(u) - \delta_{i-1}^n(u)) + (\delta_i^{n+1}(u) - \delta_{i-1}^{n+1}(u))\big] \\
&= \frac{\kappa}{2h^2}\{[h^2u_{xx} + \frac{1}{12}h^4u_{xxxx} + O(h^5)] + [h^2u_{xx} + h^2ku_{xxt} + \frac{1}{12}h^4u_{xxxx} + \frac{1}{2}h^2k^2u_{xxtt} \\
&+ O(h^5) + O(k^5)]\} = \kappa(u_{xx} + \frac{1}{2}ku_{xxt} + \frac{1}{12}h^2u_{xxxx} + \frac{1}{4}k^2u_{xxtt}) + O(h^3) + O(k^3).
\end{aligned}
\tag{2.30}
$$

Subtracting (2.30) from (2.29) leads to

$$\begin{aligned}
&\frac{1}{8k}\left[(u_{i-1}^{n+1}+6u_i^{n+1}+u_{i+1}^{n+1})-(u_{i-1}^{n}+6u_i^{n}+u_{i+1}^{n})\right]\\
&\qquad\qquad -\frac{\kappa}{2h^2}\left[(\delta_i^n(u)-\delta_{i-1}^n(u))+(\delta_i^{n+1}(u)-\delta_{i-1}^{n+1}(u))\right]\\
&\quad = u_t-\kappa u_{xx}+\frac{1}{2}k\frac{\partial}{\partial t}[u_t-\kappa u_{xx}]+\frac{h^2}{8}u_{xxt}-\frac{\kappa}{12}h^2u_{xxxx}+\frac{1}{6}k^2u_{ttt}-\frac{\kappa}{4}k^2u_{xxtt}\\
&\quad + O(h^3)+O(k^{3)}).
\end{aligned} \tag{2.31}$$

Next we combine the flux terms. So, by subtracting (2.27) from (2.28) we get

$$\begin{aligned}
&\frac{1}{4k}[(\tilde{F}_i^{n+1}-\tilde{F}_{i-1}^{n+1})-(\tilde{F}_i^{n}-\tilde{F}_{i-1}^{n})]\\
&\quad=\frac{1}{4k}[2C_1u+C_2hu_x+C_1ku_t+\frac{1}{2}C_1h^2u_{xx}+C_3hku_{xt}+\frac{1}{2}k^2C_1u_{tt}+\frac{1}{6}C_2h^3u_{xxx}\\
&\quad+\frac{1}{2}C_4h^2ku_{xxt}+\frac{1}{2}C_3hk^2u_{xtt}+\frac{1}{6}C_1k^3u_{ttt}+O(h^4)+O(k^4)],
\end{aligned} \tag{2.32}$$

where

$$C_1=A_{i-1}-A_i+a_{i-1}-a_i, \qquad C_2=-(A_{i-1}+A_i+a_{i-1}+a_i),$$

$$C_3=-(A_{i-1}+a_i), \qquad C_4=A_{i-1}-a_i.$$

Since the above constants involve Θ_{i-1} and Θ_i, we compute the Taylor expansions for Θ_{i-1} and Θ_i about the point $(\bar{x}_i,t_n)$. Let $v(x,t))$ be a smooth solution of (2.2) and let $v_i^n=v(\bar{x}_i,t_n)$ be its values at the points $(\bar{x}_i,t_n)$, then

$$\begin{aligned}
\Theta_{i-1}&=-\frac{k}{2h^2}[(v_i^{n+1}-v_{i-1}^{n+1})+(v_i^n-v_{i-1}^n)]\\
&=-\frac{k}{2h^2}[2hv_x-h^2v_{xx}+hkv_{xt}+\frac{1}{3}h^3v_{xxx}-\frac{1}{2}h^2kv_{xxt}+\frac{1}{2}hk^2v_{xtt}-\frac{1}{12}h^4v_{xxxx}\\
&+\frac{1}{6}h^3kv_{xxxt}-\frac{1}{4}h^2k^2v_{xxtt}+\frac{1}{6}hk^3v_{xttt}+O(h^5)+O(k^5)],
\end{aligned}$$

$$\begin{aligned}
\Theta_{i}&=-\frac{k}{2h^2}[(v_{i+1}^{n+1}-v_{i}^{n+1})+(v_{i+1}^n-v_{i}^n)]\\
&=-\frac{k}{2h^2}[2hv_x+h^2v_{xx}+hkv_{xt}+\frac{1}{3}h^3v_{xxx}+\frac{1}{2}h^2kv_{xxt}+\frac{1}{2}hk^2v_{xtt}+\frac{1}{12}h^4v_{xxxx}\\
&+\frac{1}{6}h^3kv_{xxxt}+\frac{1}{4}h^2k^2v_{xxtt}+\frac{1}{6}hk^3v_{xttt}+O(h^5)+O(k^5)].
\end{aligned}$$

Using these two equations, the expansions of the above constants are given by

$$\begin{aligned}
C_1 &= A_{i-1} - A_i + a_{i-1} - a_i \\
&= \Theta_{i-1} + \Theta_{i-1}^2 - \Theta_i - \Theta_i^2 + \Theta_{i-1} - \Theta_{i-1}^2 - \Theta_i + \Theta_i^2 \\
&= 2(\Theta_{i-1} - \Theta_i) \\
&= 2k[v_{xx} + \tfrac{1}{2}kv_{xxt} + \tfrac{1}{12}h^2 v_{xxxx} + \tfrac{1}{4}k^2 v_{xxtt} + O(h^3) + O(k^3)],
\end{aligned}$$

$$\begin{aligned}
C_2 &= -(A_{i-1} + A_i + a_{i-1} + a_i) \\
&= -2(\Theta_{i-1} + \Theta_i) \\
&= \tfrac{k}{h}[4v_x + 2kv_{xt} + \tfrac{2}{3}h^2 v_{xxx} + k^2 v_{xtt} + O(h^3) + O(k^3)],
\end{aligned}$$

$$\begin{aligned}
C_3 &= -(A_{i-1} + a_i) = -[(\Theta_{i-1} + \Theta_i)(1 + \Theta_{i-1} - \Theta_i)] \\
&= \tfrac{1}{2}C_2 + \tfrac{1}{4}C_1 C_2 \\
&= \tfrac{1}{2}\tfrac{k}{h}[4v_x + 2kv_{xt} + \tfrac{2}{3}h^2 v_{xxx} + k^2 v_{xtt} + O(h^3) + O(k^3)] \\
&+ \tfrac{1}{2}\tfrac{k^2}{h}[v_{xx} + \tfrac{1}{2}kv_{xxt} + \tfrac{1}{12}h^2 v_{xxxx} + \tfrac{1}{4}k^2 v_{xxtt} + O(h^3) + O(k^3)] \\
&\cdot [4v_x + 2kv_{xt} + \tfrac{2}{3}h^2 v_{xxx} + k^2 v_{xtt} + O(h^3) + O(k^3)] \\
&= 2\tfrac{k}{h}v_x + \tfrac{k^2}{h}v_{xt} + 2\tfrac{k^2}{h}v_x v_{xx} + O(h^2) + O(k^2),
\end{aligned}$$

$$\begin{aligned}
C_4 &= A_{i-1} - a_i \\
&= \Theta_{i-1}^2 + \Theta_i^2 + \Theta_{i-1} - \Theta_i = [\Theta_{i-1}(\Theta_{i-1} - \Theta_i) + \Theta_i(\Theta_i + \Theta_{i-1})] + (\Theta_{i-1} - \Theta_i) \\
&= (\Theta_{i-1} - \Theta_i)(1 + \Theta_{i-1}) + \Theta_i(\Theta_{i-1} + \Theta_i) = \tfrac{1}{2}C_1 + \tfrac{1}{2}C_1\Theta_{i-1} - \tfrac{1}{2}C_2\Theta_i \\
&= O(k).
\end{aligned}$$

For simplicity we are going to expand the right side of (2.32) term by term . So, we have

$$
\begin{aligned}
2C_1u &= 4k[v_{xx} + \frac{1}{2}kv_{xxt} + \frac{1}{12}h^2v_{xxxx} + \frac{1}{4}k^2v_{xxtt} + O(h^3) + O(k^3)]u \\
&= 4kv_{xx}u + 2k^2v_{xxt}u + \frac{1}{3}h^2kv_{xxxx}u + k^3v_{xxtt}u + O(h^3k) + O(k^4), \\
C_2hu_x &= k[4v_x + 2kv_{xt} + \frac{2}{3}h^2v_{xxx} + k^2v_{xtt} + O(h^3) + O(k^3)]u_x \\
&= 4kv_xu_x + 2k^2v_{xt}u_x + \frac{2}{3}h^2kv_{xxx}u_x + k^3v_{xtt}u_x + O(h^3k) + O(k^4), \\
C_1ku_t &= 2k^2[v_{xx} + \frac{1}{2}kv_{xxt} + \frac{1}{12}h^2v_{xxxx} + \frac{1}{4}k^2v_{xxtt} + O(h^3) + O(k^3)]u_t \\
&= 2k^2v_{xx}u_t + k^3v_{xxt}u_t + O(h^2k^2) + O(k^4), \\
\frac{1}{2}C_1h^2u_{xx} &= h^2k[v_{xx} + \frac{1}{2}kv_{xxt} + \frac{1}{12}h^2v_{xxxx} + \frac{1}{4}k^2v_{xxtt} + O(h^3) + O(k^3)]u_{xx} \\
&= h^2kv_{xx}u_{xx} + O(h^4) + O(k^5), \\
C_3hku_{xt} &= hk[2\frac{k}{h}v_x + \frac{k^2}{h}v_{xt} + 2\frac{k^2}{h}v_xv_{xx} + O(h^2) + O(k^2)]u_{xt} \\
&= 2k^2v_xu_{xt} + k^3v_{xt}u_{xt} + 2k^3v_xv_{xx}u_{xt} + O(h^4) + O(k^2), \\
\frac{1}{2}k^2C_1u_{tt} &= k^3[v_{xx} + \frac{1}{2}kv_{xxt} + \frac{1}{12}h^2v_{xxxx} + \frac{1}{4}k^2v_{xxtt} + O(h^3) + O(k^3)]u_{tt} \\
&= k^3v_{xx}u_{tt} + O(h^2k^3) + O(k^4), \\
\frac{1}{6}C_2h^3u_{xxx} &= \frac{1}{6}h^2k[4v_x + 2kv_{xt} + \frac{2}{3}h^2v_{xxx} + k^2v_{xtt} + O(h^3) + O(k^3)]u_{xxx} \\
&= \frac{2}{3}h^2kv_xu_{xxx} + O(h^4k) + O(h^2k^2), \\
\frac{1}{2}C_3hk^2u_{xtt} &= \frac{1}{2}hk^2[2\frac{k}{h}v_x + \frac{k^2}{h}v_{xt} + 2\frac{k^2}{h}v_xv_{xx} + O(h^2) + O(k^2)]u_{xtt} \\
&= k^3v_xu_{xtt} + O(h^3k^2) + O(k^4),
\end{aligned}
$$

$$
\begin{aligned}
\frac{1}{2}C_4h^2ku_{xxt} &= O(h^2k^2), \\
\frac{1}{6}C_1K^3u_{ttt} &= \frac{1}{3}k^4[v_{xx} + \frac{1}{2}kv_{xxt} + \frac{1}{12}h^2v_{xxxx} + \frac{1}{4}k^2v_{xxtt} + O(h^3) + O(k^3)]u_{ttt} \\
&= O(h^2k^4) + O(k^4).
\end{aligned}
$$

Hence, the equation (2.32) can be written as

$$\begin{aligned}
&\frac{1}{4k}[(\tilde{F}_i^{n+1} - \tilde{F}_{i-1}^{n+1}) - (\tilde{F}_i^{n} - \tilde{F}_{i-1}^{n})] \\
&\quad = v_{xx}u + v_x u_x + \frac{1}{2}k\big[v_{xxt}u + v_{xt}u_x + v_x u_{xt} + v_{xx}u_t\big] \\
&\quad + h^2\big[\frac{1}{12}v_{xxxx}u + \frac{1}{6}v_{xxx}u_x + \frac{1}{4}v_{xx}u_{xx} + \frac{1}{6}v_x u_{xxx}\big] \\
&\quad + \frac{1}{4}k^2\big[v_{xxtt}u + v_{xtt}u_x + v_{xxt}u_t + v_{xt}u_{xt} + 2v_x v_{xx}u_{xt} + v_{xx}u_{tt} + v_x u_{xtt}\big] + O(h^3) + O(k^3).
\end{aligned} \tag{2.33}$$

By adding (2.31) and (2.33) we get

$$\begin{aligned}
\tau_i^{n+1} &= [u_t - \kappa u_{xx} + (v_x u)_x] + \frac{1}{2}k\frac{\partial}{\partial t}[u_t - \kappa u_{xx} + (v_x u)_x] \\
&+ h^2[\frac{1}{12}v_{xxxx}u + \frac{1}{6}v_{xxx}u_x + \frac{1}{4}v_{xx}u_{xx} + \frac{1}{6}v_x u_{xxx} + \frac{1}{8}u_{xxt} - \frac{1}{12}\kappa u_{xxxx}] \\
&+ \frac{1}{4}k^2[v_{xxtt}u + v_{xtt}u_x + v_{xxt}u_t + v_{xt}u_{xt} + 2v_x v_{xx}u_{xt} + v_{xx}u_{tt} + v_x u_{xtt} + \frac{2}{3}u_{ttt} - \kappa u_{xxtt}] \\
&+ O(h^3) + O(k^3).
\end{aligned}$$

Since the expression in the first two brackets is the left hand side of the differential equation, it follows that the scheme is second order in both time and space.

2.5 Stability

The next step in our discussion is the stability of the method. We devote this section to discuss this property.

Definition 2.5.1 Let Q be an $M \times M$ matrix and $\{U^n\}_0^N$ be a sequence of $M \times 1$ vectors, then, the difference scheme

$$U^{n+1} = QU^n, \quad n \geq 0$$

is stable with respect to the norm $\|\cdot\|$ if there exist positive constants Δx_0 and Δt_0, and non-negative constants K and C so that

$$\|U^N\| \leq Ke^{CT}\|U^0\|$$

for $0 \leq T = N\Delta t$, $0 < \Delta x \leq \Delta x_0$ and $0 < \Delta t \leq \Delta t_0$.

First we will discuss the stability of the method in case there is no convection ($v = 0$) and then move to the general case.

Case I: $\Theta_i^n = 0$ for all i: In this case, the scheme (2.20) has the matrix form

$$\left(\frac{1}{8}B + \frac{1}{2}r_\kappa A\right) U^{n+1} = \left(\frac{1}{8}B - \frac{1}{2}r_\kappa A\right) U^n \tag{2.34}$$

where A and B are $M \times M$ matrices given by

$$A = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} 7 & 1 & & & \\ 1 & 6 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 6 & 1 \\ & & & 1 & 7 \end{bmatrix}. \tag{2.35}$$

These two matrices have the eigenvalues

$$\lambda_k = 2\left(1 - \cos\left(\frac{k\pi}{M}\right)\right), \tag{2.36}$$

$$\mu_k = 6 + 2\cos\left(\frac{k\pi}{M}\right), \tag{2.37}$$

respectively with associated eigenvectors

$$v^k = \left[\cos\left(\frac{(j-\frac{1}{2})k\pi}{M}\right)\right]_{1\leq j\leq M}, \qquad k = 0, 1, ..., M-1 \tag{2.38}$$

[see Appendix A]. The eigenvalues of the left hand matrix of the scheme (2.34) are given by

$$\left(\frac{1}{8}\mu_k + \frac{1}{2}r_\kappa \lambda_k\right).$$

Since $\frac{1}{8}\mu_k + \frac{1}{2}r_\kappa\lambda_k > 0$ for all k , the matrix $(\frac{1}{8}B + \frac{1}{2}r_\kappa A)$ is guaranteed to be invertible. Therefore, for eigenvector v^k we have

$$\left(\frac{1}{8}B + \frac{1}{2}r_\kappa A\right)^{-1}\left(\frac{1}{8}B - \frac{1}{2}r_\kappa A\right) v^k = \frac{\frac{1}{8}\mu_k - \frac{1}{2}r_\kappa\lambda_k}{\frac{1}{8}\mu_k + \frac{1}{2}r_\kappa\lambda_k} v^k$$

i.e the eigenvalues of the iteration matrix of the scheme (2.34) are given by

$$\frac{\frac{1}{8}\mu_k - \frac{1}{2}r_\kappa\lambda_k}{\frac{1}{8}\mu_k + \frac{1}{2}r_\kappa\lambda_k}, \qquad k = 0, 1, ..., M-1.$$

Since $\left|\frac{\frac{1}{8}\mu_k - \frac{1}{2}r_\kappa\lambda_k}{\frac{1}{8}\mu_k + \frac{1}{2}r_\kappa\lambda_k}\right| \le 1$, the scheme (2.34) is unconditionally stable.

Case II: $\Theta_i^n \neq 0$ *for some* i: In this case the scheme (2.20) has the matrix form

$$\left(\frac{1}{8}B + \frac{1}{2}r_\kappa A + \frac{1}{4}F_L^n\right)U^{n+1} = \left(\frac{1}{8}B - \frac{1}{2}r_\kappa A + \frac{1}{4}F_R^n\right)U^n \tag{2.39}$$

where

$$F_L^n = \begin{bmatrix} -A_1 & -a_1 & & & \\ A_1 & (a_1 - A_2) & -a_2 & & \\ & \ddots & \ddots & \ddots & \\ & & A_{M-2} & (a_{M-2} - A_{M-1}) & -a_{M-1} \\ & & & A_{M-1} & a_{M-1} \end{bmatrix}, \tag{2.40}$$

$$F_R^n = \begin{bmatrix} a_1 & A_1 & & & \\ -a_1 & (a_2 - A_1) & A_2 & & \\ & \ddots & \ddots & \ddots & \\ & & -a_{M-2} & (a_{M-1} - A_{M-2}) & A_{M-1} \\ & & & -a_{M-1} & -A_{M-1} \end{bmatrix}. \tag{2.41}$$

Similar to the first case, the left hand matrix in (2.39) needs to be invertible. To show this we need the following proposition.

Levy-Desplanques Theorem: Any strictly diagonally dominant square matrix must be invertible [32].

Proof: Let $Q = [q_{ij}]$ be an $M \times M$ strictly diagonal dominant matrix and let

$$R_i = \sum_{\substack{j=1 \\ j \neq i}}^{M} |q_{ij}|.$$

Then the diagonal dominance property implies that $R_i < |q_{ii}|$. Suppose that Q is singular, then the system $Qx = 0$ has a non-trivial solution $x = (x_1, x_2, ..., x_M)$. Let r be the index such that

$$|x_r| \geq |x_i|, \qquad i = 1, 2, ..., M.$$

Then we have

$$|q_{rr}||x_r| = \Big| - \sum_{\substack{j=1 \\ j\neq r}}^{M} q_{rj}x_j \Big| \leq \sum_{\substack{j=1 \\ j\neq r}}^{M} |q_{rj}||x_r| \leq R_r|x_r|.$$

By dividing over $|x_r|$ we get $|q_{rr}| \leq R_r$ which is a contradiction to the strictly diagonally dominance. Therefore, Q is invertible. $\square$

In the following we show that the matrix $(\frac{1}{8}B + \frac{1}{2}r_\kappa A + \frac{1}{4}F_L)$ is strictly diagonally dominant. The matrix is tridiagonal and the non-zero entries in ith row are given by

$$\left(\frac{1}{8} - \frac{1}{2}r_\kappa + \frac{1}{4}A_{i-1}\right) \qquad \left(\frac{6}{8} + r_\kappa + \frac{1}{4}(a_{i-1} - A_i)\right) \qquad \left(\frac{1}{8} - \frac{1}{2}r_\kappa - \frac{1}{4}a_i\right)$$

Since $A_i \in [-\frac{1}{4}, \frac{3}{4}]$, $a_i \in [-\frac{3}{4}, \frac{1}{4}]$, then $\frac{1}{4}(a_{i-1} - A_i) \in [-\frac{3}{8}, \frac{1}{8}]$, $\frac{1}{2} + A_i > 0$ and $\frac{1}{2} - a_i > 0$. Adding the absolute values of the off diagonal entries to get

$$\begin{aligned}\left|\tfrac{1}{8} - \tfrac{1}{2}r_\kappa + \tfrac{1}{4}A_{i-1}\right| + \left|\tfrac{1}{8} - \tfrac{1}{2}r_\kappa - \tfrac{1}{4}a_i\right| &\leq r_\kappa + \tfrac{1}{4}\left(\left|\tfrac{1}{2} + A_{i-1}\right| + \left|\tfrac{1}{2} - a_i\right|\right) \\ &= r_\kappa + \tfrac{2}{8} + \tfrac{1}{4}(A_{i-1} - a_i) = r_\kappa + \tfrac{1}{4} + \tfrac{1}{4}\big[\Theta_{i-1} + \Theta_{i-1}^2 - \Theta_i + \Theta_i^2\big].\end{aligned} \tag{2.42}$$

However, the absolute value of the diagonal entry leads to

$$\left|r_\kappa + \tfrac{6}{8} + \tfrac{1}{4}(a_{i-1} - A_i)\right| = r_\kappa + \tfrac{6}{8} + \tfrac{1}{4}(a_{i-1} - A_i) = r_\kappa + \tfrac{3}{4} + \tfrac{1}{4}\big[\Theta_{i-1} - \Theta_{i-1}^2 - \Theta_i - \Theta_i^2\big]. \tag{2.43}$$

The difference of the right hand sides of (2.42)-(2.43) gives

$$-\frac{1}{2} + \frac{1}{2}(\Theta_{i-1}^2 + \Theta_i^2) < 0,$$

which implies that

$$\left|\tfrac{1}{8} - \tfrac{1}{2}r_\kappa + \tfrac{1}{4}A_{i-1}\right| + \left|\tfrac{1}{8} - \tfrac{1}{2}r_\kappa - \tfrac{1}{4}a_i\right| < \left|r_\kappa + \tfrac{6}{8} + \tfrac{1}{4}(a_{i-1} - A_i)\right|. \tag{2.44}$$

Therefore, the matrix $(\frac{1}{8}B + \frac{1}{2}r_\kappa A + \frac{1}{4}F^n_L)$ is strictly diagonally dominant and hence invertible. So, U^{n+1} which is given by

$$\begin{aligned} U^{n+1} &= \left(\frac{1}{8}B + \frac{1}{2}r_\kappa A + \frac{1}{4}F^n_L\right)^{-1}\left(\frac{1}{8}B - \frac{1}{2}r_\kappa A + \frac{1}{4}F^n_R\right)U^n \\ &= \left(I + \frac{1}{4}\left(\frac{1}{8}B + \frac{1}{2}r_\kappa A\right)^{-1}F^n_L\right)^{-1}\left(\frac{1}{8}B + \frac{1}{2}r_\kappa A\right)^{-1}\left(\frac{1}{8}B - \frac{1}{2}r_\kappa A + \frac{1}{4}F^n_R\right)U^n \end{aligned} \tag{2.45}$$

is well defined.

In order to prove the stability of the scheme, we are going to prove that the following inequality

$$\|\tfrac{1}{4}(\tfrac{1}{8}B + \tfrac{1}{2}r_\kappa A)^{-1}F^n_L\| < 1 \tag{2.46}$$

is satisfied and then apply the Neumann series. Recall that the eigenvalues of the matrix $(\frac{1}{8}B + \frac{1}{2}rA)$ are given by

$$\begin{aligned} \frac{1}{8}\mu_k + \frac{1}{2}r_\kappa\lambda_k &= \tfrac{1}{8}(6 + 2\cos(\tfrac{k\pi}{M})) + r_\kappa(1 - \cos(\tfrac{k\pi}{M})) \\ &= \tfrac{3}{4} + (\tfrac{1}{4} - r_\kappa)\cos(\tfrac{k\pi}{M}) + r_\kappa, \qquad k = 0, 1, 2, ..., M-1. \end{aligned}$$

These eigenvalues satisfy the inequalities

$$\frac{1}{8}\mu_k + \frac{1}{2}r_\kappa\lambda_k \geq \begin{cases} 1, & r_\kappa > \frac{1}{4} \\ \dfrac{1}{2} + 2r_\kappa & r_\kappa < \frac{1}{4} \end{cases}.$$

From the definition (2.15) we get

$$|\Theta^n_i| \leq \frac{\Delta t}{\Delta x}\frac{1}{2}\left\{\frac{|V^n_{i+1} - V^n_i|}{\Delta x} + \frac{|V^{n+1}_{i+1} - V^{n+1}_i|}{\Delta x}\right\}.$$

Assuming that $\frac{|V^{n+1}_{i+1} - V^{n+1}_i|}{\Delta x}$ and $\frac{|V^n_{i+1} - V^n_i|}{\Delta x}$ are uniformly bounded by C_0 for some constant C_0. Then, for fixed Δx we have

$$\begin{aligned} |\Theta^n_i| &\leq \frac{\Delta t}{\Delta x}\frac{1}{2}\max_i\left\{\frac{|V^n_{i+1} - V^n_i|}{\Delta x} + \frac{|V^{n+1}_{i+1} - V^{n+1}_i|}{\Delta x}\right\} \\ &\leq \frac{\Delta t}{\Delta x}C_0 \\ &\leq \beta\Delta t \leq \beta\Delta t_0 \end{aligned}$$

for some $\Delta t_0 \geq \Delta t$, where $\beta = \frac{1}{\Delta x} C_0$.

Using the stated bound of $|\Theta_i^n|$ we find the upper bound of the flux matrix norm. The 1-norm of the flux matrix F_L^n is bounded above by

$$\begin{aligned}
\|F_L^n\|_1 &= \max_j (|a_{j-1}| + |a_{j-1} - A_j| + |A_j|) \\
&\leq \max_j 2(|a_{j-1}| + |A_j|) = \max_j 2(|\Theta_{j-1}^n - (\Theta_{j-1}^n)^2| + |\Theta_j^n + (\Theta_j^n)^2|) \\
&\leq 4(\beta\Delta t + (\beta\Delta t)^2).
\end{aligned}$$

The sup-norm is bounded above by

$$\begin{aligned}
\|F_L^n\|_\infty &= \max_i (|A_{i-1}| + |a_{i-1} - A_i| + |a_i|) \\
&\leq \max_i (|A_{i-1}| + |a_{i-1}| + |A_i| + |a_i|) \leq 4(\beta\Delta t + (\beta\Delta t)^2).
\end{aligned}$$

Therefore,

$$\|F_L^n\|_2 \leq \sqrt{\|F_L^n\|_1 \|F_L^n\|_\infty} \leq 4(\beta\Delta t + (\beta\Delta t)^2).$$

By the same argument we get

$$\|F_R^n\|_2 \leq 4(\beta\Delta t + (\beta\Delta t)^2).$$

We set $Q = (\frac{1}{8}B + \frac{1}{2}r_\kappa A)^{-1}(\frac{1}{8}B - \frac{1}{2}r_\kappa A)$ and $\delta Q = \frac{1}{4}(\frac{1}{8}B + \frac{1}{2}r_\kappa A)^{-1} F_L^n$. If $r_\kappa > \frac{1}{4}$, then $\|(\frac{1}{8}B + \frac{1}{2}r_\kappa A)^{-1}\|_2 \leq 1$. So, with assumption $\beta\Delta t_0 \in (0, \frac{1}{2})$ we have

$$\|\delta Q\|_2 = \|\tfrac{1}{4}(\frac{1}{8}B + \frac{1}{2}r_\kappa A)^{-1} F_L^n\|_2 \leq \beta\Delta t + (\beta\Delta t)^2 < 1,$$

and the condition (2.46) holds. Therefore, the Neumann series

$$(I + \delta Q)^{-1} = I - \delta Q + (-\delta Q)^2 + ...$$

is convergent and we have

$$\|(I + \delta Q)^{-1}\|_2 = \|I - \delta Q + (-\delta Q)^2 + ...\|_2 \leq 1 + \|\delta Q\|_2 + \|\delta Q\|_2^2 ... = \frac{1}{1 - \|\delta Q\|_2}. \quad (2.47)$$

The right hand side of the above inequality has the following upper bound

$$\frac{1}{1-\|\delta Q\|_2} = 1 + \frac{\|\delta Q\|_2}{1-\|\delta Q\|_2} \leq 1 + \frac{\beta\Delta t + (\beta\Delta t)^2}{1-(\beta\Delta t + (\beta\Delta t)^2)}$$

$$\leq 1 + \frac{2\beta\Delta t}{1-(\beta\Delta t_0 + (\beta\Delta t_0)^2)} = 1 + \tilde{C}\Delta t,$$

where $\tilde{C} = \frac{2\beta}{1-(\beta\Delta t_0+(\beta\Delta t_0)^2)}$. On the other hand, if $r_\kappa < \frac{1}{4}$ we have

$$\|(\frac{1}{8}B + \frac{1}{2}r_\kappa A)^{-1}\|_2 \leq (\tfrac{1}{2} + 2r_\kappa)^{-1}.$$

In this case we let $\beta\Delta t_0 \in (0,\alpha)$ where $\alpha < \frac{1}{2}$. So

$$\|\delta Q\|_2 = \|\tfrac{1}{4}(\frac{1}{8}B + \frac{1}{2}r_\kappa A)^{-1}F_L^n\|_2 \leq (\tfrac{1}{2}+2r_\kappa)^{-1}(\beta\Delta t + (\beta\Delta t)^2) < 2\alpha(\tfrac{1}{2}+2r_\kappa)^{-1}$$

To satisfy the condition (2.46), we let $\alpha \leq \frac{(\frac{1}{2}+2r_\kappa)}{2}$. Hence, we have

$$\frac{1}{1-\|\delta Q\|_2} = 1 + \frac{\|\delta Q\|_2}{1-\|\delta Q\|_2} \leq 1 + \frac{2(\beta\Delta t + (\beta\Delta t)^2)}{1-2(\beta\Delta t + (\beta\Delta t)^2)}$$

$$\leq 1 + \frac{4\beta\Delta t}{1-2(\alpha+\alpha^2)} = 1 + \tilde{C}\Delta t,$$

where $\tilde{C} = \frac{4\beta}{1-2(\alpha+\alpha^2)}$. For simplicity, we write

$$\tilde{C} = \begin{cases} \frac{2\beta}{1-(\beta\Delta t_0+(\beta\Delta t_0)^2)} & \text{if} \quad r_\kappa > \frac{1}{4} \\ \frac{4\beta}{1-2(\alpha+\alpha^2)} & \text{if} \quad r_\kappa < \frac{1}{4} \end{cases}.$$

The iteration matrix of the scheme (2.39) is given by

$$(I+\delta Q)^{-1}(\tfrac{1}{8}B + \tfrac{1}{2}r_\kappa A)^{-1}(\tfrac{1}{8}B - \tfrac{1}{2}r_\kappa A + \tfrac{1}{4}F_R^n) = (I+\delta Q)^{-1}(Q + \frac{1}{4}(\tfrac{1}{8}B + \tfrac{1}{2}r_\kappa A)^{-1}F_R^n).$$

Therefore, if $r_\kappa > \frac{1}{4}$ then

$$\begin{aligned}
\|(I+\delta Q)^{-1}(Q + \tfrac{1}{4}(\tfrac{1}{8}B + \tfrac{1}{2}r_\kappa A)^{-1}F_R^n)\|_2 &\leq (1+\tilde{C}\Delta t)(1 + \tfrac{1}{4}\|(\tfrac{1}{8}B + \tfrac{1}{2}r_\kappa A)^{-1}F_R^n\|_2) \\
&\leq (1+\tilde{C}\Delta t)(1 + (\beta\Delta t + (\beta\Delta t)^2)) \\
&\leq (1+\tilde{C}\Delta t)(1+2\beta\Delta t) \\
&= 1 + \tilde{C}\Delta t + 2\beta\Delta t + 2\tilde{C}\beta\Delta t^2 \\
&= 1 + (\tilde{C} + 2\beta + 2\tilde{C}\beta\Delta t)\Delta t \\
&< 1 + (\tilde{C} + 2\beta + \tilde{C})\Delta t \\
&= 1 + C\Delta t
\end{aligned}$$

where $C = 2(\tilde{C} + \beta)$. If $r_\kappa < \frac{1}{4}$ then $C = 2(\tilde{C} + 2\beta)$. Hence, letting T and N be the final time and total time steps respectively then by iteration we get

$$\begin{aligned} \|U^{n+1}\| &\leq (1 + C\Delta t)^n \|U^0\| \\ &\leq \left(1 + C\tfrac{T}{N}\right)^N \|U^0\| \leq e^{CT}\|U^0\|. \end{aligned}$$

Therefore, the scheme(2.39) is conditionally stable.

CHAPTER 3. A 2D SEMI-LAGRANGIAN SCHEME

3.1 Derivation of the method

In this chapter we are going to extend the idea we used in the one-dimensional case and apply it to a two-dimensional system. We will consider the following system

$$u_t + \nabla \cdot (u\nabla v) = \nabla \cdot (\kappa \nabla u) + g(u, v, x, y, t) \tag{3.1}$$

$$v_t = \nabla \cdot (\sigma \nabla v) - \lambda v + f(u, v, x, y, t) \qquad (x, y) \in \Omega, t > 0 \tag{3.2}$$

$$\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = 0, \qquad (x, y) \in \partial\Omega, t > 0, \tag{3.3}$$

$$u(x, y, 0) = u_0, \qquad v(x, y, 0) = v_0 \tag{3.4}$$

where $\Omega = (0, 1) \times (0, 1)$ is a rectangle in the plane and $g(u, v, x, y, t)$ and $f(u, v, x, y, t)$ are source/decay terms.

Let M_x and M_y denote the number of subdivisions in the x and y directions, respectively, and set $\Delta x = \frac{1}{M_x}$,$\Delta y = \frac{1}{M_y}$

$$x_i = i\Delta x, \quad i = 0, 1, ..., M_x \quad \text{and} \quad y_j = j\Delta y, \quad j = 0, 1, ..., M_y.$$

The cells $R_{i,j}$ (the control volumes) and cell centres $(\bar{x}_i, \bar{y}_j)$ are defined by

$$R_{ij} = [x_{i-1}, x_i] \times [y_{j-1}, y_j], \quad (\bar{x}_i, \bar{y}_j) = \tfrac{1}{2}(x_{i-1} + x_i, y_{j-1} + y_j),$$

for $i = 1, 2, ..., M_x, j = 1, 2, ..., M_y$. The dual grid cells $R^d_{i,j}$ are given by

$$R^d_{i,j} = [\bar{x}_{i-1}, \bar{x}_i] \times [\bar{y}_{j-1}, \bar{y}_j]$$

for $i = 1, 2, ..., M_x, j = 1, 2, ..., M_y$. The goal is to compute the solution at time t_{n+1} using the data at time t_n.

As we did in the one dimensional case, we track the characteristics associated with the left-hand side of the equation (3.1). These are solutions of the initial value problem:

$$\frac{dx}{dt} = v_x, \quad x(t_{n+1}) = \alpha, \quad \frac{dy}{dt} = v_y, \quad y(t_{n+1}) = \beta. \tag{3.5}$$

The image of the rectangle $R_{i,j}$ is obtained under the flow determined by the characteristic by letting (α, β) vary throughout $R_{i,j}$:

$$R_{i,j}(t) = \{x(t, \alpha, \beta), y(t, \alpha, \beta) : (\alpha, \beta) \in R_{i,j}\}.$$

For the time being, let's assume that $R_{i,j}$ is an interior cell and that $R_{i,j}(t)$ does not intersect the boundary $\partial\Omega$.
Let $Q_{i,j}^{n+1}$ denote the interior of the solid $\{(x, y, t) : (x, y) \in R_{i,j}(t), t \in [t_n, t_{n+1}]\}$ swept out under the flow. Integrating (3.1) over the region $Q_{i,j}^{n+1}$ and applying the Divergence Theorem leads to

$$\begin{aligned}\iiint\limits_{Q_{i,j}^{n+1}} (\nabla \cdot (\kappa \nabla u) + g) dV &= \iiint\limits_{Q_{i,j}^{n+1}} \langle \partial_x, \partial_y, \partial_t \rangle \cdot \langle uv_x, uv_y, u \rangle dV = \iint\limits_{\partial Q_{i,j}^{n+1}} u \langle v_x, v_y, 1 \rangle \cdot \vec{n} ds \\ &= \iint\limits_{R_{i,j}} u(x, y, t_{n+1}) dA - \iint\limits_{R_{i,j}(t_n)} u(x, y, t_n) dA.\end{aligned} \tag{3.6}$$

Here $\vec{n}$ denotes the outward unit normal to $\partial Q_{i,j}^{n+1}$. We used the fact that the characteristic directions $\langle v_x, v_y, 1 \rangle$ are tangent to the lateral surface of $\partial Q_{i,j}^{n+1}$ to write the right hand side of the above equation. In the following we will be approximating this integral identity.

To derive the method, we approximate solutions by piecewise bilinear functions on each time level. Since the functions are evaluated at the centre points of the grid cells, the data values will be $V_{i,j}^n = v(\bar{x}_i, \bar{y}_j, t_n)$ and $U_{i,j}^n = u(\bar{x}_i, \bar{y}_j, t_n)$. Let's define the following piecewise linear functions

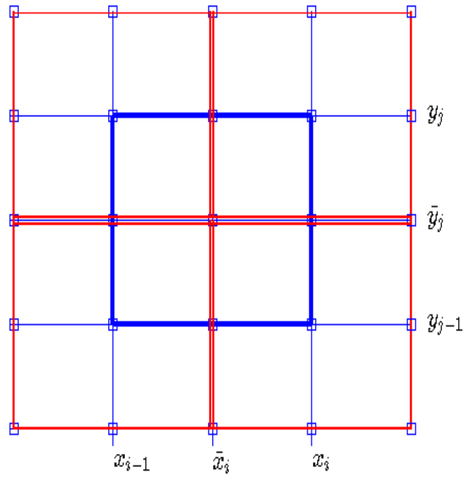

Figure 3.1 Diagonal overlapping supports of $P_{i,j}(x,y)$

$$p_{i,j}[V](x) = \begin{cases} V_{i,j} + \dfrac{V_{i+1,j} - V_{i,j}}{\Delta x}(x - \bar{x}_i), & \bar{x}_i \le x \le \bar{x}_{i+1} \\ V_{i-1,j} + \dfrac{V_{i,j} - V_{i-1,j}}{\Delta x}(x - \bar{x}_{i-1}), & \bar{x}_{i-1} \le x \le \bar{x}_i \end{cases} \quad i = 1, 2..., M-1,$$

$$q_{i,j}[V](y) = \begin{cases} V_{i,j} + \dfrac{V_{i,j+1} - V_{i,j}}{\Delta y}(y - \bar{y}_j), & \bar{y}_j \le y \le \bar{y}_{j+1} \\ V_{i,j-1} + \dfrac{V_{i,j} - V_{i,j-1}}{\Delta y}(y - \bar{y}_{j-1}), & \bar{y}_{j-1} \le y \le \bar{y}_j \end{cases} \quad j = 1, 2, ..., M-1$$

where $V_{i,j} = v(\bar{x}_i, \bar{y}_j)$. Then we can construct the following locally piecewise bilinear functions

$$P_{i,j}[V](x,y) = \begin{cases} p_{i,j}[V](x) + \dfrac{y - \bar{y}_j}{\Delta y}(p_{i,j+1}[V](x) - p_{i,j}[V](x)), & (x,y) \in R^d_{i+1,j+1} \\ p_{i,j-1}[V](x) + \dfrac{y - \bar{y}_{j-1}}{\Delta y}(p_{i,j}[V](x) - p_{i,j-1}[V](x)), & (x,y) \in R^d_{i+1,j} \\ p_{i-1,j}[V](x) + \dfrac{y - \bar{y}_j}{\Delta y}(p_{i-1,j+1}[V](x) - p_{i-1,j}[V](x)), & (x,y) \in R^d_{i,j+1} \\ p_{i-1,j-1}[V](x) + \dfrac{y - \bar{y}_{j-1}}{\Delta y}(p_{i-1,j}[V](x) - p_{i-1,j-1}[V](x)), & (x,y) \in R^d_{i,j} \end{cases} \tag{3.7}$$

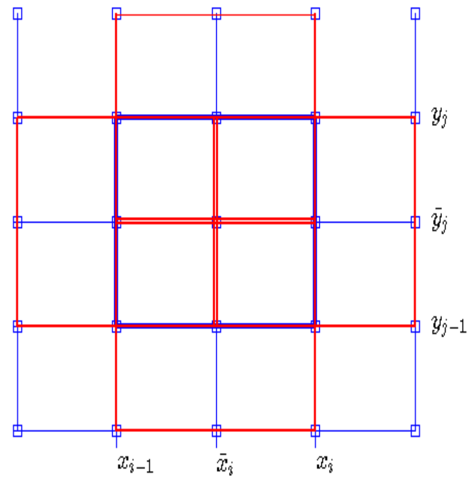

Figure 3.2 Horizontal and vertical overlapping supports of $P_{i,j}(x,y)$

and

$$Q_{i,j}[V](x,y) = \begin{cases} q_{i,j}[V](y) + \dfrac{x-\bar{x}_i}{\Delta x}(q_{i+1,j}[V](y) - q_{i,j}[V](y)), & (x,y) \in R^d_{i+1,j+1} \\ q_{i-1,j}[V](y) + \dfrac{x-\bar{x}_{i-1}}{\Delta x}(q_{i,j}[V](y) - q_{i-1,j}[V](y)), & (x,y) \in R^d_{i,j+1} \\ q_{i,j-1}[V](y) + \dfrac{x-\bar{x}_i}{\Delta x}(q_{i+1,j-1}[V](y) - q_{i,j-1}[V](y)), & (x,y) \in R^d_{i+1,j} \\ q_{i-1,j-1}[V](y) + \dfrac{x-\bar{x}_{i-1}}{\Delta x}(q_{i,j-1}[V](y) - q_{i-1,j-1}[V](y)), & (x,y) \in R^d_{i,j}. \end{cases} \tag{3.8}$$

It is clear that $P_{i,j}[V](x,y) = Q_{i,j}[V](x,y)$ for $(x,y) \in R^d_{i+1,j+1}$. Therefore, we will use them alternatively in deriving the method. In particular, we will use $Q_{i,j}[V](x,y)$ if integrating with respect to x is needed and $P_{i,j}[V](x,y)$ if we need to integrate with respect to y. These functions have the property that $P_{i,j}[V](\bar{x}_i,\bar{y}_j) = Q_{i,j}[V](\bar{x}_i,\bar{y}_j) = V_{i,j}$. Figure 3.1 and 3.2 show the supports of these functions.

Now, we proceed to approximate the integral identity (3.6). The difficulties come from the second integral on the right side of (3.6) while the first one can be shown as a special case of the second integral. The boundaries of the departure cell $R_{i,j}(t_n)$ are smooth curves (Figure 3.3) rather than straight lines so that the departure cell sides must be approximated. The region of $R_{i,j}(t_n)$ is approximated by tracking the vertices moving with the characteristic flow and connecting them by straight lines (Figure 3.4).

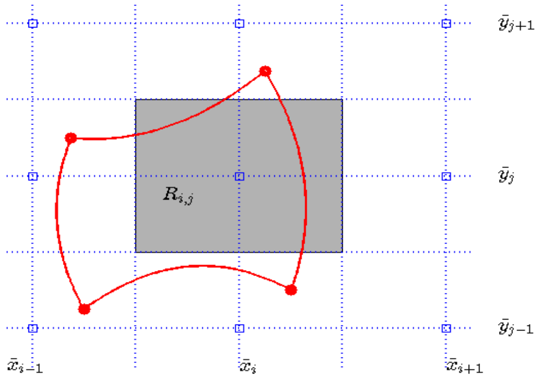

Figure 3.3 The departure cell $R_{i,j}(t_n)$ defined by curves vs. arrival cell $R_{i,j}$ (shaded area).

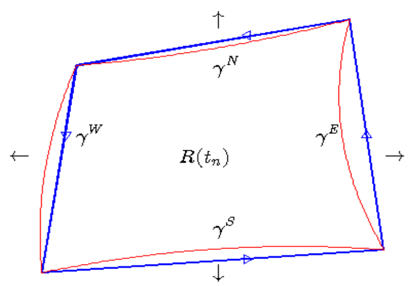

Figure 3.4 Approximating the sides of $R_{i,j}(t_n)$ by straight lines.

In order to achieve that, we need to solve the characteristic equations.

Let $(x_{i,j}(t), y_{i,j}(t))$ be the solution of the characteristic equation

$$\begin{bmatrix} x \\ y \end{bmatrix}' = \begin{bmatrix} v_x(x,y,t) \\ v_y(x,y,t) \end{bmatrix}, \qquad \begin{bmatrix} x \\ y \end{bmatrix}(t_{n+1}) = \begin{bmatrix} x_i \\ y_j \end{bmatrix}. \tag{3.9}$$

By integrating the system (3.9) and approximating the right hand side using trapezoidal rule we get

$$\begin{aligned} \begin{bmatrix} x \\ y \end{bmatrix}(t_{n+1}) &= \begin{bmatrix} x \\ y \end{bmatrix}(t_n) + \int_{t_n}^{t_{n+1}} \begin{bmatrix} v_x(x(s),y(s),s) \\ v_y(x(s),y(s),s) \end{bmatrix} ds \\ &\approx \begin{bmatrix} x \\ y \end{bmatrix}(t_n) + \frac{1}{2}\Delta t \begin{bmatrix} v_x(x(t_n),y(t_n),t_n) + v_x(x(t_{n+1}),y(t_{n+1}),t_{n+1}) \\ v_y(x(t_n),y(t_n),t_n) + v_y(x(t_{n+1}),y(t_{n+1}),t_{n+1}) \end{bmatrix}. \end{aligned} \tag{3.10}$$

In order to solve this system, we need to approximate the derivatives on the right side of (3.10). Since $v(x,y,t) \cong P_{i,j}[V](x,y,t)$, then $\nabla v \cong \nabla P_{i,j}[V]$. So by setting $x(t) = x_{i,j}(t)$ and $y(t) = y_{i,j}(t)$ we can write

$$\begin{aligned} v_x(x_i,y_j,t_{n+1}) &\approx \big(\frac{\partial}{\partial x}P_{i,j}[V^{n+1}]\big)(x_i,y_j) = \frac{1}{2}\big[p'_{i,j}[V^{n+1}] + p'_{i,j+1}[V^{n+1}]\big] = \frac{1}{2}[\tilde{V}_x^{n+1}]_{i,j} \\ v_y(x_i,y_j,t_{n+1}) &\approx \big(\frac{\partial}{\partial y}Q_{i,j}[V^{n+1}]\big)(x_i,y_j) = \frac{1}{2}\big[q'_{i,j}[V^{n+1}] + q'_{i,j+1}[V^{n+1}]\big] = \frac{1}{2}[\tilde{V}_y^{n+1}]_{i,j}, \end{aligned}$$

where, we are writing $V_{i,j}^{n+1} = v(\bar{x}_i, \bar{y}_j, t_{n+1})$,

$$\begin{aligned} [\tilde{V}_x^{n+1}]_{i,j} &= \frac{1}{\Delta x}[V_{i+1,j}^{n+1} - V_{i,j}^{n+1} + V_{i+1,j+1}^{n+1} - V_{i,j+1}^{n+1}], \\ [\tilde{V}_y^{n+1}]_{i,j} &= \frac{1}{\Delta y}[V_{i,j+1}^{n+1} - V_{i,j}^{n+1} + V_{i+1,j+1}^{n+1} - V_{i+1,j}^{n+1}]. \end{aligned}$$

However, the derivatives at t_n are given by

$$\begin{aligned} v_x(x(t_n),y(t_n),t_n) &\approx \big(\frac{\partial}{\partial x}P_{i,j}[V^n]\big)(x(t_n),y(t_n)) \\ &= p'_{i,j}[V^n] + \frac{y(t_n)-\bar{y}_j}{\Delta y}\big[p'_{i,j+1}[V^n] - p'_{i,j}[V^n]\big], \end{aligned}$$

and

$$\begin{aligned} v_y(x(t_n),y(t_n),t_n) &\approx \big(\frac{\partial}{\partial y}Q_{i,j}[V^n]\big)(x(t_n),y(t_n)) \\ &= q'_{i,j}[V^n] + \frac{x(t_n)-\bar{x}_i}{\Delta y}\big[q'_{i+1,j}[V^n] - q'_{i,j}[V^n]\big]. \end{aligned}$$

We notice that $y(t_n) - \bar{y}_j = y(t_n) - (y_j - \frac{1}{2}\Delta y) = y(t_n) - y_j + \frac{1}{2}\Delta y$. So, the system (3.10) can be written as

$$\begin{aligned} x_i &\approx x(t_n) + \frac{1}{2}\Delta t\Big[p'_{i,j}[V^n] + \big(\frac{y(t_n)-y_j}{\Delta y} + \frac{1}{2}\big)\big(p'_{i,j+1}[V^n] - p'_{i,j}[V^n]\big) + [\tilde{V}_x^{n+1}]_{i,j}\Big], \\ y_j &\approx y(t_n) + \frac{1}{2}\Delta t\Big[q'_{i,j}[V^n] + \big(\frac{x(t_n)-x_i}{\Delta x} + \frac{1}{2}\big)\big(q'_{i+1,j}[V^n] - q'_{i,j}[V^n]\big) + [\tilde{V}_y^{n+1}]_{i,j}\Big]. \end{aligned}$$

Rearranging terms and writing the results in matrix form give

$$\begin{bmatrix} 1 & \frac{1}{2}\frac{\Delta t}{\Delta y}\left(p'_{i,j+1}[V^n]-p'_{i,j}[V^n]\right) \\ \frac{1}{2}\frac{\Delta t}{\Delta x}\left(q'_{i+1,j}[V^n]-q'_{i,j}[V^n]\right) & 1 \end{bmatrix}\begin{bmatrix} x(t_n)-x_i \\ y(t_n)-y_j \end{bmatrix}$$

$$= -\frac{1}{2}\Delta t\begin{bmatrix} \frac{1}{2}\left(p'_{i,j}[V^n]+p'_{i,j+1}[V^n]\right)+[\tilde{V}^{n+1}_x]_{i,j} \\ \frac{1}{2}\left(q'_{i,j}[V^n]+q'_{i+1,j}[V^n]\right)+[\tilde{V}^{n+1}_y]_{i,j} \end{bmatrix}.$$

Assuming that the left hand matrix is not singular, we multiply by its inverse to get

$$\begin{bmatrix} x(t_n)-x_i \\ y(t_n)-y_j \end{bmatrix} = -\frac{1}{2}\Delta t\frac{1}{1-\frac{1}{4}\frac{\Delta t}{\Delta x}\frac{\Delta t}{\Delta y}\left(p'_{i,j+1}[V^n]-p'_{i,j}[V^n]\right)\left(q'_{i+1,j}[V^n]-q'_{i,j}[V^n]\right)}\times$$

$$\begin{bmatrix} 1 & -\frac{1}{2}\frac{\Delta t}{\Delta y}\left(p'_{i,j+1}[V^n]-p'_{i,j}[V^n]\right) \\ -\frac{1}{2}\frac{\Delta t}{\Delta x}\left(q'_{i,j+1}[V^n]-q'_{i,j}[V^n]\right) & 1 \end{bmatrix}\times$$

$$\begin{bmatrix} \frac{1}{2}\left(p'_{i,j}[V^n]+p'_{i,j+1}[V^n]\right)+[\tilde{V}^{n+1}_x]_{i,j} \\ \frac{1}{2}\left(q'_{i,j}[V^n]+q'_{i+1,j}[V^n]\right)+[\tilde{V}^{n+1}_y]_{i,j} \end{bmatrix}.$$

It is easy to see that $\dfrac{\left(p'_{i,j+1}[V^n]-p'_{i,j}[V^n]\right)}{\Delta y} = \dfrac{\left(q'_{i+1,j}[V^n]-q'_{i,j}[V^n]\right)}{\Delta x} = [\tilde{V}^n_{xy}]_{i,j}$

where

$$[\tilde{V}^n_{xy}]_{i,j} = \frac{1}{\Delta x\Delta y}[V^n_{i+1,j+1}-V^n_{i+1,j}-V^n_{i,j+1}+V^n_{i,j}].$$

Hence the system can be written in the form

$$\begin{bmatrix} x(t_n)-x_i \\ y(t_n)-y_j \end{bmatrix} = -\frac{1}{2}\Delta t\frac{1}{1-\frac{1}{4}(\Delta t)^2([\tilde{V}^n_{xy}]_{i,j})^2}\times\begin{bmatrix} 1 & -\frac{1}{2}\Delta t[\tilde{V}^n_{xy}]_{i,j} \\ -\frac{1}{2}\Delta t[\tilde{V}^n_{xy}]_{i,j} & 1 \end{bmatrix}\times$$

$$\begin{bmatrix} \frac{1}{2}([\tilde{V}^n_x]_{i,j}+[\tilde{V}^{n+1}_x]_{i,j}) \\ \frac{1}{2}([\tilde{V}^n_y]_{i,j}+[\tilde{V}^{n+1}_y]_{i,j}) \end{bmatrix}.$$

For convenience, we set

$$\theta^{x,n}_{i,j} = \frac{x_{i,j}(t_n)-x_i}{\Delta x}, \qquad \theta^{y,n}_{i,j} = \frac{y_{i,j}(t_n)-y_j}{\Delta y}.$$

By the identity

$$\frac{1}{1-(\frac{1}{2}\Delta t[\tilde{V}^n_{xy}]_{i,j})^2} = 1+(\tfrac{1}{2}\Delta t[\tilde{V}^n_{xy}]_{i,j})^2+(\tfrac{1}{2}\Delta t[\tilde{V}^n_{xy}]_{i,j})^4+O((\Delta t)^6)$$

we can write the solution of the system as (ignoring the higher order terms)

$$\begin{bmatrix} \Delta x \theta_{i,j}^{x,n} \\ \Delta y \theta_{i,j}^{y,n} \end{bmatrix} \approx \\ \begin{bmatrix} \Delta x \Theta_{i,j}^{x,n} \\ \Delta y \Theta_{i,j}^{y,n} \end{bmatrix} = -\tfrac{1}{4}\Delta t \begin{bmatrix} [\tilde{V}_x^n]_{i,j} + [\tilde{V}_x^{n+1}]_{i,j} \\ [\tilde{V}_y^n]_{i,j} + [\tilde{V}_y^{n+1}]_{i,j} \end{bmatrix} - \tfrac{1}{8}(\Delta t)^2 [\tilde{V}_{xy}^n]_{i,j} \begin{bmatrix} [\tilde{V}_y^n]_{i,j} + [\tilde{V}_y^{n+1}]_{i,j} \\ [\tilde{V}_x^n]_{i,j} + [\tilde{V}_x^{n+1}]_{i,j} \end{bmatrix}. \tag{3.11}$$

Remark 3.1: *We assume that* $|\theta_{i,j}^x| \leq \frac{1}{2}$ *and* $|\theta_{i,j}^y| \leq \frac{1}{2}$ *for all* i *and* j *so that* $(x_{i,j}(t_n), y_{i,j}(t_n)) \in R_{i+1,j+1}^d$. *Therefore, the line integral along each side of* $\partial R_{i,j}(t_n)$ *will not be broken into more than two pieces. Saying differently, the integrated function will not be defined by more than two pieces.*

It is clear that

$$x_{i,j}(t_n) = x_i + \theta_{i,j}^{x,n} \Delta x, \qquad y_{i,j}(t_n) = y_j + \theta_{i,j}^{y,n} \Delta y.$$

We will use these equations in parametrizing the curve $\partial R_{i,j}(t_n)$.

In order to evaluate the integrals over the departure cells we use the Divergence Theorem to convert the area integral into a line integral. Let $U(x, y, t_n)$ be a piecewise bilinear approximation of $u(x, y, t_n)$ on the dual grid cells $\{R_{i,j}^d\}$ as defined in (3.7) and (3.8) with V replaced by U. Since t_n will be fixed throughout the coming computations, we write $U(x, y) = U(x, y, t_n)$. Let

$$W^x(x, y) = \int_{\bar{x}_i}^{x} U(z, y) dz, \qquad W^y(x, y) = \int_{\bar{y}_i}^{y} U(x, z) dz,$$

so that $\frac{1}{2} div\langle W^x, W^y \rangle = U(x, y)$. By the Divergence Theorem

$$\iint_{R_{i,,j}(t_n)} u(x, y) dxdy \approx \iint_{R_{i,,j}(t_n)} U(x, y) dxdy = \frac{1}{2} \int_{\partial R_{i,j}(t_n)} \langle W^x, W^y \rangle \cdot \vec{n} ds, \tag{3.12}$$

where $\vec{n}$ is the unit outer normal vector along the boundary $\partial R_{i,j}(t_n)$. To compute the integral on the right side of (3.12), the curve $\partial R_{i,j}(t_n)$ is parametrized by the characteristic flow $(x(t_n,\alpha,\beta), y(t_n,\alpha,\beta))$ defined by (3.5). For example, the east part of the approximation of $\partial R_{i,j}(t_n)$ is parametrized by connecting the two vertices $(x_{i,j-1}(t_n), y_{i,j-1}(t_n))$ and $(x_{i,j}(t_n), y_{i,j}(t_n))$. So the parametrizing equations are given by

$$x_E(\sigma) = x_{i,j-1}(t_n) + \sigma(x_{i,j}(t_n) - x_{i,j-1}(t_n)) = (x_i + \theta^x_{i,j-1}\Delta x) + \sigma(\theta^x_{i,j} - \theta^x_{i,j-1})\Delta x,$$

$$y_E(\sigma) = y_{i,j-1}(t_n) + \sigma(y_{i,j}(t_n) - y_{i,j-1}(t_n)) = (y_{j-1} + \theta^y_{i,j-1}\Delta y) + \sigma(1 + \theta^y_{i,j} - \theta^y_{i,j-1})\Delta y.$$

The unit outer normal vector $\vec{n}$ is given as follows

$$\langle x'(\sigma), y'(\sigma)\rangle \cdot \vec{n} = 0,$$

which implies that $\vec{n} = \frac{\langle y'(\sigma), -x'(\sigma)\rangle}{\sqrt{(x'(\sigma)^2 + (y'(\sigma))^2}}$. Let γ^E, γ^N, γ^W and γ^S be the east, north, west and south parametrizations of $\partial R_{i,j}(t_n)$, then

$$\begin{aligned}
\iint\limits_{R(t_n)} u(x,y)dA \approx \iint\limits_{R(t_n)} U(x,y)dA &= \frac{1}{2}\int\limits_{\partial R(t_n)} \langle W^x, W^y\rangle \cdot \vec{n} ds \\
&= \frac{1}{2}\Big\{\int\limits_{\gamma^E}(W^x y' - W^y x')d\sigma - \int\limits_{\gamma^W}(W^x y' - W^y x')d\sigma \\
&+ \int\limits_{\gamma^N}(W^y x' - W^x y')d\sigma - \int\limits_{\gamma^S}(W^y x' - W^x y')d\sigma\Big\}.
\end{aligned} \tag{3.13}$$

3.1.1 A line integral along the east edge of $R(t_n)$

We now compute the integrals on the right sides of (3.13). The parametrized equations of the east edge γ^E are given by

$$\begin{aligned}
x_E(\sigma) &= (x_i + \Delta x\theta^x_{i,j-1}) + \sigma(\theta^x_{i,j} - \theta^x_{i,j-1})\Delta x, \\
y_E(\sigma) &= (y_{j-1} + \Delta y\theta^y_{i,j-1}) + \sigma(1 + \theta^y_{i,j} - \theta^y_{i,j-1})\Delta y, \qquad \sigma \in [0,1].
\end{aligned} \tag{3.14}$$

By the assumption $(x_{i,j}(t_n), y_{i,j}(t_n)) \in [\bar{x}_i, \bar{x}_{i+1}] \times [\bar{y}_j, \bar{y}_{j+1}]$ we see that the parametrized line $(x_E(\sigma), y_E(\sigma))$ will be intersecting the line $y = \bar{y}_j$. Let σ^* be the intersecting parameter of these lines. Then

$$\int_{\gamma^E} \langle W^x, W^y \rangle \cdot \vec{n} ds = \left(\int_0^{\sigma *} + \int_{\sigma^*}^1 \right) [W^x(x_E(\sigma), y_E(\sigma)) y_E'(\sigma) - W^y(x_E(\sigma), y_E(\sigma)) x_E'(\sigma)] d\sigma, \tag{3.15}$$

where

$$W^x(x,y) = \begin{cases} q_{i,j}[U](y)(x-\bar{x}_i) + \frac{1}{2}\frac{(x-\bar{x}_i)^2}{\Delta x}(q_{i+1,j}[U](y) - q_{i,j}[U](y)), \\ \hfill (x,y) \in R^d_{i+1,j+1} \\ q_{i,j-1}[U](y)(x-\bar{x}_i) + \frac{1}{2}\frac{(x-\bar{x}_i)^2}{\Delta x}(q_{i+1,j-1}[U](y) - q_{i,j-1}[U](y)), \\ \hfill (x,y) \in R^d_{i+1,j} \end{cases}$$

and

$$W^y(x,y) = \begin{cases} p_{i,j}[U](x)(y-\bar{y}_j) + \frac{1}{2}\dfrac{(y-\bar{y}_i)^2}{\Delta y}(p_{i,j+1}[U](x) - p_{i,j}[U](x)), \\ \hfill (x,y) \in R^d_{i+1,j+1} \\ p_{i,j-1}[U](x)(y-\bar{y}_j) + \dfrac{(y-\bar{y}_{j-1})^2 - (\Delta y)^2}{2\Delta y}(p_{i,j}[U](x) - p_{i,j-1}[U](x)), \\ \hfill (x,y) \in R^d_{i+1,j}. \end{cases}$$

In the following we will be computing the first integral in (3.15). For simplicity we will write $W_0^x = W^x(x_E(0), y_E(0))$, $W_1^x = W^x(x_E(1), y_E(1))$, $W_{\sigma^*}^x = W^x(x_E(\sigma^*), y_E(\sigma^*))$, and $q_{i,j} = q_{i,j}[U]$. We notice that the derivative $y_E'(\sigma)$ is constant. So, using the trapezoidal rule gives

$$\begin{aligned}(\int_0^{\sigma^*} + \int_{\sigma^*}^1) W^x(x_E(\sigma), y_E(\sigma)) y_E'(\sigma)(\sigma) d\sigma &\cong \frac{1}{2} y_E'[\sigma^*(W_0^x + W_{\sigma *}^x) + (1-\sigma^*)(W_1^x + W_{\sigma^*}^x)] \\ &= \frac{1}{2} y_E'(\sigma)[\sigma^* W_0^x + (1-\sigma^*) W_1^x + W_{\sigma^*}^x],\end{aligned}$$

where

$$W_0^x = q_{i,j-1}(y_E(0))(x_E(0) - \bar{x}_i) + \tfrac{1}{2}\frac{(x_E(0) - \bar{x}_i)^2}{\Delta x}\left[q_{i+1,j-1}(y_E(0)) - q_{i,j-1}(y_E(0))\right],$$

$$W_1^x = q_{i,j}(y_E(1))(x_E(1) - \bar{x}_i) + \tfrac{1}{2}\frac{(x_E(1) - \bar{x}_i)^2}{\Delta x}\left[q_{i+1,j}(y_E(1)) - q_{i,j}(y_E(1))\right]$$

$$W_{\sigma^*}^x = q_{i,j-1}(y_E(\sigma^*))(x_E(\sigma^*) - \bar{x}_i) + \tfrac{1}{2}\frac{(x_E(\sigma^*) - \bar{x}_i)^2}{\Delta x}[q_{i+1,j-1}(y_E(\sigma^*)) - q_{i,j-1}(y_E(\sigma^*))].$$

In order to write the above integral we need the following equalities which are obtained from the parametrized equations (3.14)

$$\begin{aligned}
y_E(0) - \bar{y}_j &= (\theta^y_{i,j-1} - \tfrac{1}{2})\Delta y, \\
y_E(0) - \bar{y}_{j-1} &= (\theta^y_{i,j-1} + \tfrac{1}{2})\Delta y, \\
y_E(1) - \bar{y}_j &= (\theta^y_{i,j} + \tfrac{1}{2})\Delta y, \\
y_E(\sigma*) &= \bar{y}_j, \\
x_E(0) - \bar{x}_i &= (\theta^x_{i,j-1} + \tfrac{1}{2})\Delta x, \\
x_E(1) - \bar{x}_i &= (\theta^x_{i,j} + \tfrac{1}{2})\Delta x,
\end{aligned}$$

$$\begin{aligned}
x_E(\sigma^*) - \bar{x}_i &= x_i + \Delta x \theta^x_{i,j-1} + \sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\Delta x - \bar{x}_i \\
&= \left(\tfrac{1}{2} + \theta^x_{i,j-1} + \sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\right)\Delta x \\
&= \left(\tfrac{1}{2} + \theta^x_{i,j} + (1-\sigma^*)(\theta^x_{i,j-1} - \theta^x_{i,j})\right)\Delta x.
\end{aligned}$$

To keep the balance of the flux, we will use the first equation from the last equality with the lower dual cell $R^d_{i,j-1}$ and the second one with the upper dual cell $R^d_{i,j}$ and ignore the higher order terms $\sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\Delta x$. From these equalities we get

$$\begin{aligned}
q_{i,j-1}(y_E(0)) &= (\tfrac{1}{2} - \theta^y_{i,j-1})U_{i,j-1} + (\tfrac{1}{2} + \theta^y_{i,j-1})U_{i,j}, \\
q_{i+1,j-1}(y_E(0)) &= (\tfrac{1}{2} - \theta^y_{i,j-1})U_{i+1,j-1} + (\theta^y_{i,j-1} - \tfrac{1}{2})U_{i+1,j+1}, \\
q_{i+1,j}(y_E(1)) &= (\tfrac{1}{2} - \theta^y_{i,j})U_{i+1,j} + (\tfrac{1}{2} + \theta^y_{i,j})U_{i+1,j+1}, \\
q_{i,j}(y_E(1)) &= (\tfrac{1}{2} - \theta^y_{i,j})U_{i,j} + (\tfrac{1}{2} + \theta^y_{i,j})U_{i,j+1}, \\
q_{i+1,j-1}(y_E(\sigma*)) &= q_{i+1,j-1}(\bar{y}_j) = q_{i,j-1}(\bar{y}) = U_{i,j}.
\end{aligned}$$

Then, using all the above equalities leads to

$$
\begin{aligned}
W_0^x &= (\tfrac{1}{2} + \theta^x_{i,j-1})\Delta x\Big[(\tfrac{1}{2} - \theta^y_{i,j-1})U_{i,j-1} + (\tfrac{1}{2} + \theta^y_{i,j-1})U_{i,j}\Big] \\
&+ \tfrac{1}{2}(\tfrac{1}{2} + \theta^x_{i,j-1})^2\Delta x\Big[(\tfrac{1}{2} - \theta^y_{i,j-1})U_{i+1,j-1} + (\tfrac{1}{2} + \theta^y_{i,j-1})U_{i+1,j} \\
&- (\tfrac{1}{2} - \theta^y_{i,j-1})U_{i,j-1} - (\tfrac{1}{2} + \theta^y_{i,j-1})U_{i,j}\Big] \\
&= \tfrac{1}{2}\Delta x\Big\{U_{i,j-1}\Big[(\tfrac{1}{2} - \theta^y_{i,j-1})(\tfrac{1}{2} + \theta^x_{i,j-1})(\tfrac{3}{2} - \theta^x_{i,j-1})\Big] \\
&+ U_{i,j}\Big[(\tfrac{1}{2} + \theta^y_{i,j-1})(\tfrac{1}{2} + \theta^x_{i,j-1})(\tfrac{3}{2} - \theta^x_{i,j-1})\Big] \\
&+ U_{i+1,j-1}\Big[(\tfrac{1}{2} - \theta^y_{i,j-1})(\tfrac{1}{2} + \theta^x_{i,j-1})^2\Big] + U_{i+1,j}\Big[(\tfrac{1}{2} + \theta^y_{i,j-1})(\tfrac{1}{2} + \theta^x_{i,j-1})^2\Big]\Big\},
\end{aligned}
$$

$$
\begin{aligned}
W^x(x_E(\sigma^*),\bar{y}_j) &= \Big[\tfrac{1}{2} + \theta^x_{i,j-1} + \sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\Big]\Delta x U_{i,j} \\
&+ \tfrac{1}{2}\Big[\tfrac{1}{2} + \theta^x_{i,j-1} + \sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\Big]^2\Delta x(U_{i+1,j} - U_{i,j}) \\
&= \tfrac{1}{2}\Delta x\Big\{\Big[\tfrac{1}{2} + \theta^x_{i,j-1} + \sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\Big]\Big[\tfrac{3}{2} - \theta^x_{i,j-1} + \sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\Big]U_{i,j} \\
&+ \Big[\tfrac{1}{2} + \theta^x_{i,j-1} + \sigma^*(\theta^x_{i,j} - \theta^x_{i,j-1})\Big]^2 U_{i+1,j}\Big\} \\
&= \tfrac{1}{2}\Delta x\Big\{\Big[\tfrac{1}{2} + \theta^x_{i,j} + (1-\sigma^*)(\theta^x_{i,j-1} - \theta^x_{i,j})\Big]\Big[\tfrac{3}{2} - \theta^x_{i,j} + (1-\sigma^*)(\theta^x_{i,j-1} - \theta^x_{i,j})\Big]U_{i,j} \\
&+ \Big[\tfrac{1}{2} + \theta^x_{i,j} + (1-\sigma^*)(\theta^x_{i,j-1} - \theta^x_{i,j})\Big]^2 U_{i+1,j}\Big\},
\end{aligned}
$$

$$
\begin{aligned}
W_1^x &= \tfrac{1}{2}\Delta x\Big\{U_{i,j}\Big[(\tfrac{1}{2} - \theta^y_{i,j})(\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{3}{2} - \theta^x_{i,j})\Big] + U_{i,j+1}\Big[(\tfrac{1}{2} + \theta^y_{i,j})(\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{3}{2} - \theta^x_{i,j})\Big] \\
&+ U_{i+1,j}\Big[(\tfrac{1}{2} - \theta^y_{i,j})(\tfrac{1}{2} + \theta^x_{i,j})^2\Big] + U_{i+1,j+1}\Big[(\tfrac{1}{2} + \theta^y_{i,j})(\tfrac{1}{2} + \theta^x_{i,j})^2\Big]\Big\}.
\end{aligned}
$$

From the equality $y_E(\sigma^*) = \bar{y}_j$ we get

$$
\begin{aligned}
&(\theta^y_{i,j-1} + \sigma^*(1 + \theta^y_{i,j} - \theta^y_{i,j-1}))\Delta y = \theta^y_{i,j-1}\Delta y + \sigma^* y'_E = \tfrac{1}{2}\Delta y, \\
&\Rightarrow \quad \sigma^* y'_E = (\tfrac{1}{2} - \theta^y_{i,j-1})\Delta y,
\end{aligned}
$$

and

$$y_E'(1-\sigma^*) = [(1+\theta^y_{i,j}-\theta^y_{i,j-1}) - (\tfrac{1}{2}-\theta^y_{i,j-1})]\Delta y = (\tfrac{1}{2}+\theta^y_{i,j})\Delta y.$$

Also, in the following we will write $y_E'(\sigma)$ as

$$y_E'(\sigma) = (1-\theta^y_{i,j-1}+\theta^y_{i,j})\Delta y = \left[(\tfrac{1}{2}-\theta^y_{i,j-1}) + (\tfrac{1}{2}+\theta^y_{i,j})\right]\Delta y.$$

Hence, the integral along the east edge is given by

$$\begin{aligned}
&\left(\int_0^{\sigma^*} + \int_{\sigma^*}^1\right)[W^x(x_E(\sigma), y_E(\sigma))y_E'(\sigma)]d\sigma \\
&\quad\approx \tfrac{1}{2}\Delta y\Big\{(\tfrac{1}{2}-\theta^y_{i,j-1})W^x_0 + (\tfrac{1}{2}+\theta^y_{i,j})W^x_1 + [(\tfrac{1}{2}-\theta^y_{i,j-1}) + (\tfrac{1}{2}+\theta^y_{i,j})]W^x_{\sigma^*}\Big\} \\
&\quad\approx \tfrac{1}{4}\Delta x\Delta y\Big\{U_{i,j}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})(\tfrac{3}{2}+\theta^y_{i,j-1})(\tfrac{1}{2}+\theta^x_{i,j-1})(\tfrac{3}{2}-\theta^x_{i,j-1}) \\
&\qquad\qquad + (\tfrac{1}{2}+\theta^y_{i,j})(\tfrac{3}{2}-\theta^y_{i,j})(\tfrac{1}{2}+\theta^x_{i,j})(\tfrac{3}{2}-\theta^x_{i,j})\Big] \\
&\quad+U_{i,j-1}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})^2(\tfrac{1}{2}+\theta^x_{i,j-1})(\tfrac{3}{2}-\theta^x_{i,j-1})\Big] + U_{i,j+1}\Big[(\tfrac{1}{2}+\theta^y_{i,j})^2(\tfrac{1}{2}+\theta^x_{i,j})(\tfrac{3}{2}-\theta^x_{i,j})\Big] \\
&\quad+U_{i+1,j}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})(\tfrac{3}{2}+\theta^y_{i,j-1})(\tfrac{1}{2}+\theta^x_{i,j-1})^2 + (\tfrac{1}{2}+\theta^y_{i,j})(\tfrac{3}{2}-\theta^y_{i,j})(\tfrac{1}{2}+\theta^x_{i,j})^2\Big] \\
&\quad+U_{i+1,j-1}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})^2(\tfrac{1}{2}+\theta^x_{i,j-1})^2\Big] + U_{i+1,j+1}\Big[(\tfrac{1}{2}+\theta^y_{i,j})^2(\tfrac{1}{2}+\theta^x_{i,j})^2\Big]\Big\}.
\end{aligned} \tag{3.16}$$

Remark 3.2: *In the above approximation we ignored the second integral in (3.15) because it gives higher order terms of the increments as we show. Let $\Delta t = \nu h$ and $\Delta x = \Delta y = h$, then assuming v_x and v_{xy} are bounded we obtain*

$$\begin{aligned}
x_E' &= (\theta^x_{i,j}-\theta^x_{i,j-1})\Delta x = \left(\frac{x_{i,j}(t_n) - x_{i,j-1}(t_n)}{\Delta x}\right)\Delta x \\
&= \int_{t_{n+1}}^{t_n} [v_x(x_i(t), y_j(t), t) - v_x(x_i(t), y_{j-1}(t), t)]dt \\
&\approx \frac{\Delta t}{2}[v_x(x_i, y_j, t_{n+1}) - v_x(x_i, y_{j-1}, t_{n+1}) + v_x(x_i(t_n), y_j(t_n), t_n) - v_x(x_i(t_n), y_{j-1}(t_n), t_n)] \\
&\cong C\Delta t\Delta y,
\end{aligned}$$

for some constant C. Hence

$$\int_0^1 W^y(x(\sigma), y(\sigma))x'(\sigma)d\sigma \cong C\Delta t\Delta x\Delta y = C\nu h^3.$$

3.1.2 A line integral along the south edge of $R(t_n)$

Next we compute the line integral along the south edge γ^S. The parametrized equations of γ^S are given by connecting the two vertices $(x_{i-1,j-1}(t_n), y_{i-1,j-1}(t_n)$ and $(x_{i,j-1}(t_n), y_{i,j-1}(t_n))$. So, we have

$$\begin{aligned} x_S(\sigma) &= x_{i-1} + \Delta x\theta^x_{i-1,j-1} + \sigma(1 + \theta^x_{i,j-1} - \theta^x_{i-1,j-1})\Delta x, \\ y_S(\sigma) &= y_{j-1} + \Delta y\theta^y_{i-1,j-1} + \sigma(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})\Delta y, \quad \sigma \in [0,1]. \end{aligned} \tag{3.17}$$

Again we will consider σ^* to be the intersecting parameter of the line $x = \bar{x}_i$ with the line defined by the parametrized equations $(x_S(\sigma), y_S(\sigma))$. Along the south edge γ^S we have

$$\int_{\gamma^S} \langle W^x, W^y\rangle \cdot \vec{n}ds = \left(\int_0^{\sigma^*} + \int_{\sigma^*}^1\right)[W^y(x_S(\sigma), y_S(\sigma))x'_S(\sigma) - W^x(x_S(\sigma), y_S(\sigma)y'_S(\sigma)]d\sigma. \tag{3.18}$$

As we did before we compute the first integral and ignore the second one since it leads to higher order of the increments . The function $W^y(x,y)$ is given by

$$W^y(x,y) = \begin{cases} p_{i,j-1}(x)(y - \bar{y}_j) + \frac{1}{2\Delta y}[(y - \bar{y}_{j-1})^2 - (\Delta y)^2](p_{i,j}(x) - p_{i,j-1}(x)), \\ \hfill (x,y) \in R^d_{i+1,j}, \\ p_{i-1,j-1}(x)(y - \bar{y}_j) + \frac{1}{2\Delta y}[(y - \bar{y}_{j-1})^2 - (\Delta y)^2](p_{i-1,j}(x) - p_{i-1,j-1}(x)), \\ \hfill (x,y) \in R^d_{i,j}. \end{cases}$$

Using the trapezoidal rule, we get

$$\left(\int_0^{\sigma^*} + \int_{\sigma^*}^1\right) W^y(x_S(\sigma), y_S(\sigma))x_S'(\sigma)d\sigma \approx \frac{1}{2}x_S'[\sigma^*(W_0^y + W_{\sigma^*}^y) + (1-\sigma^*)(W_1^y + W_{\sigma^*}^y]$$
$$= \frac{1}{2}x_S'[\sigma^* W_0^y + W_{\sigma^*}^y + (1-\sigma^*)W_1^y] \tag{3.19}$$

where

$$W_0^y = p_{i-1,j-1}(x_S(0))(y_S(0) - \bar{y}_j) + \frac{1}{2}\Delta y[(y_S(0) - \bar{y}_{j-1})^2 - (\Delta y)^2][p_{i-1,j}(x_S(0)) - p_{i-1,j-1}(x_S(0))],$$
$$W_{\sigma^*}^y = p_{i,j-1}(x_S(\sigma^*))(y_S(\sigma^*) - \bar{y}_j) + \frac{1}{2\Delta y}[(y_S(\sigma^*) - \bar{y}_{j-1})^2 - (\Delta y)^2][p_{i,j}(x_S(\sigma^*)) - p_{i,j-1}(x_S(\sigma^*))],$$
$$W_1^y = p_{i,j-1}(x_S(1))(y_S(1) - \bar{y}_j) + \frac{1}{2}\Delta y[(y_S(1) - \bar{y}_{j-1})^2 - (\Delta y)^2][p_{i,j}(x_S(1)) - p_{i,j-1}(x_S(1))].$$

To write the full expansion of the above integral, we need the following equalities which are obtained from the parametrized equations (3.17)

$$x_S(0) - \bar{x}_{i-1} = (\tfrac{1}{2} + \theta^x_{i-1,j-1})\Delta x,$$
$$x_S(0) - \bar{x}_i = (\theta^x_{i-1,j-1} - \tfrac{1}{2})\Delta x,$$
$$x_S(1) - \bar{x}_i = (\theta^x_{i,j-1} + \tfrac{1}{2})\Delta x,$$
$$y_S(0) - \bar{y}_{j-1} = (\tfrac{1}{2} + \theta^y_{i-1,j-1})\Delta y,$$
$$y_S(1) - \bar{y}_{j-1} = (\tfrac{1}{2} + \theta^y_{i,j-1})\Delta y,$$
$$y_S(0) - \bar{y}_j = (\theta^y_{i-1,j-1} - \tfrac{1}{2})\Delta y,$$
$$y_S(1) - \bar{y}_j = (\theta^y_{i,j-1} - \tfrac{1}{2})\Delta y,$$
$$y_S(\sigma^*) - \bar{y}_{j-1} = (\tfrac{1}{2} + \theta^y_{i-1,j-1})\Delta y + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})\Delta y$$
$$= (\tfrac{1}{2} + \theta^y_{i,j-1})\Delta y + (1-\sigma^*)(\theta^y_{i-1,j-1} - \theta^y_{i,j-1})\Delta y,$$
$$y_S(\sigma^*) - \bar{y}_j = (\theta^y_{i-1,j-1} - \tfrac{1}{2})\Delta y + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})\Delta y$$
$$= (\theta^y_{i,j-1} - \tfrac{1}{2})\Delta y + (1-\sigma^*)(\theta^y_{i-1,j-1} - \theta^y_{i,j-1})\Delta y.$$

Also, the following equalities are necessary for our computations

$$
\begin{aligned}
p_{i-1,j-1}(x_S(0)) &= U_{i-1,j-1} + \frac{U_{i,j-1} - U_{i-1,j-1}}{\Delta x}(\tfrac{1}{2} + \theta^x_{i-1,j-1})\Delta x \\
&= (\tfrac{1}{2} - \theta^x_{i-1,j-1})U_{i-1,j-1} + (\tfrac{1}{2} + \theta^x_{i-1,j-1})U_{i,j-1}, \\
p_{i-1,j}(x_S(0)) &= (\tfrac{1}{2} - \theta^x_{i-1,j-1})U_{i-1,j} + (\tfrac{1}{2} + \theta^x_{i-1,j-1})U_{i,j}, \\
p_{i,j-1}(x_S(1)) &= (\tfrac{1}{2} - \theta^x_{i,j-1})U_{i,j-1} + (\tfrac{1}{2} + \theta^x_{i,j-1})U_{i+1,j-1}, \\
p_{i,j}(x_S(1)) &= (\tfrac{1}{2} - \theta^x_{i,j-1})U_{i,j} + (\tfrac{1}{2} + \theta^x_{i,j-1})U_{i+1,j}, \\
p_{i,j}(x_S(\sigma^*)) &= p_{i,j}(\bar{x}_i) = p_{i-1,j}(\bar{x}_i) = U_{i,j} \\
(\tfrac{1}{2} + \theta^y_{i-1,j-1})^2 - 1 &= (\theta^y_{i-1,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i-1,j-1}).
\end{aligned}
$$

By the above equalities, we get

$$
\begin{aligned}
W^y_0 &= (\theta^y_{i-1,j-1} - \tfrac{1}{2})\Delta y\Big[(\tfrac{1}{2} - \theta^x_{i-1,j-1})U_{i-1,j-1} + (\tfrac{1}{2} + \theta^x_{i-1,j-1})U_{i,j-1}\Big] \\
&\quad + \tfrac{1}{2}\Delta y(\theta^y_{i-1,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i-1,j-1})\Big[(\tfrac{1}{2} - \theta^x_{i-1,j-1})U_{i-1,j} + (\tfrac{1}{2} + \theta^x_{i-1,j-1})U_{i,j} \\
&\quad - (\tfrac{1}{2} - \theta^x_{i-1,j-1})U_{i-1,j-1} - (\tfrac{1}{2} + \theta^x_{i-1,j-1})U_{i,j-1}\Big] \\
&= \tfrac{1}{2}\Delta y\Big\{ - U_{i-1,j-1}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j-1})(\theta^y_{i-1,j-1} - \tfrac{1}{2})^2\Big] - U_{i,j-1}\Big[(\tfrac{1}{2} + \theta^x_{i-1,j-1})(\theta^y_{i-1,j-1} - \tfrac{1}{2})^2\Big] \\
&\quad + U_{i,j}\Big[(\theta^y_{i-1,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i-1,j-1})(\tfrac{1}{2} + \theta^x_{i-1,j-1})\Big] \\
&\quad + U_{i-1,j}\Big[(\theta^y_{i-1,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i-1,j-1})(\tfrac{1}{2} - \theta^x_{i-1,j-1})\Big]\Big\},
\end{aligned}
$$

$$
\begin{aligned}
W^y_{\sigma^*} &= \Delta y \Bigg\{ \Bigg[[\theta^y_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})] \\
&- \tfrac{1}{2}\big[\theta^y_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})\big]\big[\tfrac{3}{2} + \theta^y_{i-1,j-1} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})\big]\Bigg] U_{i,j-1} \\
&+ \tfrac{1}{2}\Bigg[[\theta^y_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})][\tfrac{3}{2} + \theta^y_{i-1,j-1} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})]\Bigg] U_{i,j} \\
&= \tfrac{1}{2}\Delta y \Bigg\{ - \Bigg[\theta^y_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})\Bigg]^2 U_{i,j-1} \\
&+ \Bigg[[\theta^y_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})][\tfrac{3}{2} + \theta^y_{i-1,j-1} + \sigma^*(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})]\Bigg] U_{i,j} \Bigg\} \\
&= \tfrac{1}{2}\Delta y \Bigg\{ - \Bigg[\theta^y_{i,j-1} - \tfrac{1}{2} + (1-\sigma^*)(\theta^y_{i-1,j-1} - \theta^y_{i,j-1})\Bigg]^2 U_{i,j-1} \\
&+ \Bigg[[\theta^y_{i,j-1} - \tfrac{1}{2} + (1-\sigma^*)(\theta^y_{i-1,j-1} - \theta^y_{i,j-1})][\tfrac{3}{2} + \theta^y_{i,j-1} + (1-\sigma^*)(\theta^y_{i-1,j-1} - \theta^y_{i,j-1})]\Bigg] U_{i,j} \Bigg\},
\end{aligned}
$$

$$
\begin{aligned}
W^y_1 &= (\theta^y_{i,j-1} - \tfrac{1}{2})\Delta y \Bigg[(\tfrac{1}{2} - \theta^x_{i,j-1})U_{i,j-1} + (\tfrac{1}{2} + \theta^x_{i,j-1})U_{i+1,j-1} \Bigg] \\
&+ \tfrac{1}{2}\Delta y (\theta^y_{i,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i,j-1}) \Bigg[(\tfrac{1}{2} - \theta^x_{i,j-1})U_{i,j} + (\tfrac{1}{2} + \theta^x_{i,j-1})U_{i+1,j} \\
&- (\tfrac{1}{2} - \theta^x_{i,j-1})U_{i,j-1} - (\tfrac{1}{2} + \theta^x_{i,j-1})U_{i+1,j-1} \Bigg] \\
&= \tfrac{1}{2}\Delta y \Bigg\{ - U_{i,j-1}(\tfrac{1}{2} - \theta^x_{i,j-1})(\theta^y_{i,j-1} - \tfrac{1}{2})^2 - U_{i+1,j-1}(\tfrac{1}{2} + \theta^x_{i,j-1})(\theta^y_{i,j-1} - \tfrac{1}{2})^2 \\
&+ U_{i,j}(\theta^y_{i,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i,j-1})(\tfrac{1}{2} - \theta^x_{i,j-1}) + U_{i+1,j}(\theta^y_{i,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i,j-1})(\tfrac{1}{2} + \theta^x_{i,j-1}) \Bigg\}.
\end{aligned}
$$

By combining the last three equations and ignoring the higher order term $(\theta^y_{i,j-1} - \theta^y_{i-1,j-1})\Delta y$, we obtain the integral along the south edge. Using the equality $\sigma^* x'_S(\sigma) = (\frac{1}{2} - \theta^x_{i-1,j-1})\Delta x$ and writing $x'_S(\sigma)$ as

$$
x'_S(\sigma) = (1 - \theta^x_{i-1,j-1} + \theta^x_{i,j-1})\Delta x = \big[(\frac{1}{2} - \theta^x_{i-1,j-1}) + (\frac{1}{2} + \theta^x_{i,j-1})\big]\Delta x,
$$

we get

$$
\begin{aligned}
&\left(\int_0^{\sigma^*} + \int_{\sigma^*}^{1}\right) W^y(x_S(\sigma), y_S(\sigma)) x_S'(\sigma) d\sigma \\
\approx \quad & \frac{1}{2}\Delta x \Big\{ (\frac{1}{2} - \theta^x_{i-1,j-1}) W^y_0 + \big[(\frac{1}{2} - \theta^x_{i-1,j-1}) + (\frac{1}{2} + \theta^x_{i,j-1})\big] W^y_{\sigma^*} + (\frac{1}{2} + \theta^x_{i,j-1}) W^y_1 \Big\} \\
\approx & \frac{1}{4}\Delta x \Delta y \Big\{ - U_{i-1,j-1}(\tfrac{1}{2} - \theta^x_{i-1,j-1})^2(\theta^y_{i-1,j-1} - \tfrac{1}{2})^2 \\
& -U_{i,j-1}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j-1})(\tfrac{3}{2} + \theta^x_{i-1,j-1})(\theta^y_{i-1,j-1} - \tfrac{1}{2})^2 + (\tfrac{1}{2} + \theta^x_{i,j-1})(\tfrac{3}{2} - \theta^x_{i,j-1})(\theta^y_{i,j-1} - \tfrac{1}{2})^2\Big] \\
& +U_{i,j}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j-1})(\tfrac{3}{2} + \theta^x_{i-1,j-1})(\tfrac{3}{2} + \theta^y_{i-1,j-1})(\theta^y_{i-1,j-1} - \tfrac{1}{2}) \\
& \qquad + (\tfrac{1}{2} + \theta^x_{i,j-1})(\tfrac{3}{2} - \theta^x_{i,j-1})(\tfrac{3}{2} + \theta^y_{i,j-1})(\theta^y_{i,j-1} - \tfrac{1}{2})\Big] \\
& +U_{i-1,j}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j-1})^2(\theta^y_{i-1,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i-1,j-1})\Big] \\
& -U_{i+1,j-1}\Big[(\tfrac{1}{2} + \theta^x_{i,j-1})^2(\theta^y_{i,j-1} - \tfrac{1}{2})^2\Big] + U_{i+1,j}\Big[(\tfrac{1}{2} + \theta^x_{i,j-1})^2(\theta^y_{i,j-1} - \tfrac{1}{2})(\tfrac{3}{2} + \theta^y_{i,j-1})\Big]\Big\}.
\end{aligned}
\tag{3.20}
$$

3.1.3 Aline integral along the west edge of $R(t_n)$

We advance in our process and compute the line integral along the west edge. The parametrized equations of the west edge γ^W are given by

$$
\begin{aligned}
x_W(\sigma) &= x_{i-1} + \Delta x \theta^x_{i-1,j-1} + \sigma(\theta^x_{i-1,j} - \theta^x_{i-1,j-1})\Delta x, \\
y_W(\sigma) &= y_{j-1} + \Delta y \theta^y_{i-1,j-1} + \sigma(1 + \theta^y_{i-1,j} - \theta^y_{i-1,j-1})\Delta y, \qquad \sigma \in [0,1].
\end{aligned}
$$

Define σ^* such that $y_W(\sigma^*) = \bar{y}_j$. So, integrating along γ^W gives

$$
\int_{\gamma^W} \langle W^x, W^y\rangle \cdot \vec{n} ds = \left(\int_0^{\sigma*} + \int_{\sigma^*}^{1}\right)[W^x(x_W(\sigma), y_W(\sigma)) y_W'(\sigma) - W^y(x_W(\sigma), y_W(\sigma)) x_W'(\sigma)] d\sigma.
\tag{3.21}
$$

Again, we ignore the second integral in the above equation. The function $W^x(x,y)$ is given by

$$
W^x(x,y) = \begin{cases} q_{i-1,j}(y)(x-\bar{x}_i) + \frac{1}{2\Delta x}[(x-\bar{x}_{i-1})^2 - (\Delta x)^2](q_{i,j}(y) - q_{i-1,j}(y)), \\ \hfill (x,y) \in R^d_{i,j+1} \\ q_{i-1,j-1}(y)(x-\bar{x}_i) + \frac{1}{2\Delta x}[(x-\bar{x}_{i-1})^2 - (\Delta x)^2](q_{i,j-1}(y) - q_{i-1,j-1}(y)), \\ \hfill (x,y) \in R^d_{i,j} \end{cases}.
$$

Using the trapezoidal rule, we obtain

$$
\left(\int_0^{\sigma^*} + \int_{\sigma^*}^1\right) W^x(x_W(\sigma), y_W(\sigma))y'_W(\sigma)d\sigma \approx \frac{1}{2}y'_W\{\sigma^* W^x_0 + (1-\sigma^*)W^x_1 + W^x_{\sigma^*}\} \tag{3.22}
$$

where

$$
\begin{aligned}
W^x_0 &= q_{i-1,j-1}(y_W(0))(x_W(0)-\bar{x}_i) \\
&\qquad + \frac{1}{2\Delta x}\left[(x_W(0)-\bar{x}_{i-1})^2 - (\Delta x)^2\right](q_{i,j-1}(y_W(0)) - q_{i-1,j-1}(y_W(0))), \\
W^x_{\sigma^*} &= q_{i-1,j-1}(y_W(\sigma^*))(x_W(\sigma^*)-\bar{x}_i) \\
&\qquad + \frac{1}{2\Delta x}\left[(x_W(\sigma^*)-\bar{x}_{i-1})^2 - (\Delta x)^2\right](q_{i,j-1}(y_W(\sigma^*)) - q_{i-1,j-1}(y_W(\sigma^*))), \\
W^x_1 &= q_{i-1,j}(y_W(1))(x_W(1)-\bar{x}_i) \\
&\qquad + \frac{1}{2\Delta x}\left[(x_W(1)-\bar{x}_{i-1})^2 - (\Delta x)^2\right](q_{i,j}(y_W(1)) - q_{i-1,j}(y_W(1))).
\end{aligned}
$$

The following equalities are necessary to write the above integral

$$
\begin{aligned}
x_W(0) - \bar{x}_{i-1} &= (\tfrac{1}{2} + \theta^x_{i-1,j-1})\Delta x, \\
x_W(0) - \bar{x}_i &= (\theta^x_{i-1,j-1} - \tfrac{1}{2})\Delta x, \\
x_W(\sigma^*) - \bar{x}_{i-1} &= (\tfrac{1}{2} + \theta^x_{i-1,j-1})\Delta x + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1})\Delta x, \\
&= (\tfrac{1}{2} + \theta^x_{i-1,j})\Delta x + (1-\sigma^*)(\theta^x_{i-1,j-1} - \theta^x_{i-1,j})\Delta x \\
x_W(\sigma^*) - \bar{x}_i &= (\theta^x_{i-1,j-1} - \tfrac{1}{2})\Delta x + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1})\Delta x \\
&= (\theta^x_{i-1,j} - \tfrac{1}{2})\Delta x + (1-\sigma^*)(\theta^x_{i-1,j-1} - \theta^x_{i-1,j})\Delta x
\end{aligned}
$$

$$
\begin{aligned}
&x_W(1) - \bar{x}_{i-1} = (\tfrac{1}{2} + \theta^x_{i-1,j})\Delta x,\\
&x_W(1) - \bar{x}_i = (\theta^x_{i-1,j} - \tfrac{1}{2})\Delta x,\\
&y_W(0) - \bar{y}_{j-1} = (\tfrac{1}{2} + \theta^y_{i-1,j-1})\Delta y,\\
&y_W(1) - \bar{y}_j = (\tfrac{1}{2} + \theta^y_{i-1,j})\Delta y,\\
&y_W(\sigma^*) - \bar{y}_{j-1} = (\tfrac{1}{2} + \theta^y_{i-1,j-1})\Delta y + \sigma^*(1 + \theta^y_{i-1,j} - \theta^y_{i-1,j-1})\Delta y\\
&y_W(\sigma^*) - \bar{y}_j = (\theta^y_{i-1,j-1} - \tfrac{1}{2})\Delta y + \sigma^*(1 + \theta^y_{i-1,j} - \theta^y_{i-1,j-1})\Delta y.
\end{aligned}
$$

Also, the following equalities will be used in our computation

$$
\begin{aligned}
q_{i,j-1}(y_W(0)) &= U_{i,j-1} + \frac{U_{i,j} - U_{i,j-1}}{\Delta y}(y_W(0) - \bar{y}_{j-1})\\
&= (\tfrac{1}{2} - \theta^y_{i-1,j-1})U_{i,j-1} + (\tfrac{1}{2} + \theta^y_{i-1,j-1})U_{i,j},\\
q_{i-1,j-1}(y_W(0)) &= (\tfrac{1}{2} - \theta^y_{i-1,j-1})U_{i-1,j-1} + (\tfrac{1}{2} + \theta^y_{i-1,j-1})U_{i-1,j},\\
q_{i,j}(y_W(1)) &= (\tfrac{1}{2} - \theta^y_{i-1,j})U_{i,j} + (\tfrac{1}{2} + \theta^y_{i-1,j})U_{i,j+1},\\
q_{i-1,j}(y_W(1)) &= (\tfrac{1}{2} - \theta^y_{i-1,j})U_{i-1,j} + (\tfrac{1}{2} + \theta^y_{i-1,j})U_{i-1,j+1},\\
q_{i-1,j}(y_W(\sigma^*)) &= q_{i-1,j}(\bar{y}_j) = q_{i-1,j-1}(\bar{y}_j) = U_{i-1,j}\\
\frac{1}{2\Delta x}[(x_W(0) - \bar{x}_{i-1})^2 - (\Delta x)^2] &= \frac{1}{2}\Delta x(\theta^x_{i-1,j-1} - \tfrac{1}{2})(\theta^x_{i-1,j-1} + \tfrac{3}{2}).
\end{aligned}
$$

Then, using the above equalities we have

$$
\begin{aligned}
W^x_0 = \Delta x\Big\{&(\theta^x_{i-1,j-1} - \tfrac{1}{2})\Big[(\tfrac{1}{2} - \theta^y_{i-1,j-1})U_{i-1,j-1} + (\tfrac{1}{2} + \theta^y_{i-1,j-1})U_{i-1,j}\Big]\\
&+ \tfrac{1}{2}(\theta^x_{i-1,j-1} - \tfrac{1}{2})(\theta^x_{i-1,j-1} + \tfrac{3}{2})\Big[(\tfrac{1}{2} - \theta^y_{i-1,j-1})U_{i,j-1} + (\tfrac{1}{2} + \theta^y_{i-1,j-1})U_{i,j}\\
&- (\tfrac{1}{2} - \theta^y_{i-1,j-1})U_{i-1,j-1} - (\tfrac{1}{2} + \theta^y_{i-1,j-1})U_{i-1,j}\Big]\Big\}\\
= \tfrac{1}{2}\Delta x\Big\{&- U_{i-1,j-1}\Big[(\theta^x_{i-1,j-1} - \tfrac{1}{2})^2(\tfrac{1}{2} - \theta^y_{i-1,j-1})\Big] - U_{i-1,j}\Big[(\theta^x_{i-1,j-1} - \tfrac{1}{2})^2(\tfrac{1}{2} + \theta^y_{i-1,j-1})\Big]\\
&+ U_{i,j-1}\Big[(\theta^x_{i-1,j-1} - \tfrac{1}{2})(\theta^x_{i-1,j-1} + \tfrac{3}{2})(\tfrac{1}{2} - \theta^y_{i-1,j-1})\Big]\\
&+ U_{i,j}\Big[(\theta^x_{i-1,j-1} - \tfrac{1}{2})(\theta^x_{i-1,j-1} + \tfrac{3}{2})(\tfrac{1}{2} + \theta^y_{i-1,j-1})\Big]\Big\},
\end{aligned}
$$

$$
\begin{aligned}
W^x_{\sigma^*} = \Delta x\Big\{&(\theta^x_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1}))U_{i-1,j} \\
&+ \tfrac{1}{2}(\theta^x_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1}))(\theta^x_{i-1,j-1} + \tfrac{3}{2} + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1})) \\
&[U_{i,j} - U_{i-1,j}]\Big\} \\
&= \tfrac{1}{2}\Delta x\Big\{ - U_{i-1,j}\Big[\theta^x_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1})\Big]^2 \\
&+ U_{i,j}\Big[(\theta^x_{i-1,j-1} - \tfrac{1}{2} + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1}))(\theta^x_{i-1,j-1} + \tfrac{3}{2} + \sigma^*(\theta^x_{i-1,j} - \theta^x_{i-1,j-1}))\Big]\Big\} \\
&= \tfrac{1}{2}\Delta x\Big\{ - U_{i-1,j}\Big[\theta^x_{i-1,j} - \tfrac{1}{2} + (1-\sigma^*)(\theta^x_{i-1,j-1} - \theta^x_{i-1,j})\Big]^2 \\
&+ U_{i,j}\Big[(\theta^x_{i-1,j} - \tfrac{1}{2} + (1-\sigma^*)(\theta^x_{i-1,j-1} - \theta^x_{i-1,j}))(\theta^x_{i-1,j} + \tfrac{3}{2} + (1-\sigma^*)(\theta^x_{i-1,j-1} - \theta^x_{i-1,j}))\Big]\Big\},
\end{aligned}
$$

and

$$
\begin{aligned}
W^x_1 = \Delta x\Big\{&(\theta^x_{i-1,j} - \tfrac{1}{2})\Big[(\tfrac{1}{2} - \theta^y_{i-1,j})U_{i-1,j} + (\tfrac{1}{2} + \theta^y_{i-1,j})U_{i-1,j+1}\Big] \\
&+ \tfrac{1}{2}(\theta^x_{i-1,j} - \tfrac{1}{2})(\theta^x_{i-1,j} + \tfrac{3}{2})\Big[(\tfrac{1}{2} - \theta^y_{i-1,j})U_{i,j} + (\tfrac{1}{2} + \theta^y_{i-1,j})U_{i,j+1} \\
&- (\tfrac{1}{2} - \theta^y_{i-1,j})U_{i-1,j} - (\tfrac{1}{2} + \theta^y_{i-1,j})U_{i-1,j+1}\Big]\Big\} \\
&= \tfrac{1}{2}\Delta x\Big\{ - U_{i-1,j}\big[(\theta^x_{i-1,j} - \tfrac{1}{2})^2(\tfrac{1}{2} - \theta^y_{i-1,j})\big] - U_{i-1,j+1}\big[(\theta^x_{i-1,j} - \tfrac{1}{2})^2(\tfrac{1}{2} + \theta^y_{i-1,j})\big] \\
&+ U_{i,j}\big[(\theta^x_{i-1,j} - \tfrac{1}{2})(\theta^x_{i-1,j} + \tfrac{3}{2})(\tfrac{1}{2} - \theta^y_{i-1,j})\big] + U_{i,j+1}\big[(\theta^x_{i-1,j} - \tfrac{1}{2})(\theta^x_{i-1,j} + \tfrac{3}{2})(\tfrac{1}{2} + \theta^y_{i-1,j})\big]\Big\}.
\end{aligned}
$$

From the equality $y_W(\sigma^*) = \bar{y}_j$ we get that $\sigma^* y'_W = (\frac{1}{2} - \theta^y_{i-1,j-1})\Delta y$. By combining all of these equations, we obtain the west edge integral

$$
\begin{aligned}
&\Big(\int_0^{\sigma^*}+\int_{\sigma^*}^{1}\Big)W^x(x_W(\sigma),y_W(\sigma))y'_W(\sigma)d\sigma\\
&\approx \frac{1}{2}\Delta y\Big\{(\frac{1}{2}-\theta^y_{i-1,j-1})W^x_0+(\frac{1}{2}+\theta^y_{i-1,j})W^x_1+[(\frac{1}{2}-\theta^y_{i-1,j-1})+(\frac{1}{2}+\theta^y_{i-1,j})]W^x_{\sigma^*}\Big\}\\
&\cong \tfrac{1}{4}\Delta x\Delta y\Big\{-U_{i-1,j-1}\Big[(\theta^x_{i-1,j-1}-\tfrac{1}{2})^2(\tfrac{1}{2}-\theta^y_{i-1,j-1})^2\Big]\\
&-U_{i-1,j}\Big[(\theta^x_{i-1,j-1}-\tfrac{1}{2})^2(\tfrac{1}{2}-\theta^y_{i-1,j-1})(\tfrac{3}{2}+\theta^y_{i-1,j-1})+(\theta^x_{i-1,j}-\tfrac{1}{2})^2(\tfrac{1}{2}+\theta^y_{i-1,j})(\tfrac{3}{2}-\theta^y_{i-1,j})\Big]\\
&+U_{i,j-1}\Big[(\theta^x_{i-1,j-1}-\tfrac{1}{2})(\theta^x_{i-1,j-1}+\tfrac{3}{2})(\tfrac{1}{2}-\theta^y_{i-1,j-1})^2\Big]\\
&+U_{i,j}\Big[(\theta^x_{i-1,j-1}-\tfrac{1}{2})(\theta^x_{i-1,j-1}+\tfrac{3}{2})(\tfrac{1}{2}-\theta^y_{i-1,j-1})(\tfrac{3}{2}+\theta^y_{i-1,j-1})\\
&\qquad+(\theta^x_{i-1,j}-\tfrac{1}{2})(\theta^x_{i-1,j}+\tfrac{3}{2})(\tfrac{1}{2}+\theta^y_{i-1,j})(\tfrac{3}{2}-\theta^y_{i-1,j})\Big]\\
&-U_{i-1,j+1}\Big[(\theta^x_{i-1,j}-\tfrac{1}{2})^2(\tfrac{1}{2}+\theta^y_{i-1,j})^2\Big]+U_{i,j+1}\Big[(\theta^x_{i-1,j}-\tfrac{1}{2})(\theta^x_{i-1,j}+\tfrac{3}{2})(\tfrac{1}{2}+\theta^y_{i-1,j})^2\Big]\Big\}.
\end{aligned}
\tag{3.23}
$$

3.1.4 A line integral along the north edge of $R(t_n)$

The last integral will be along the north edge γ^N whose parametrized equations are given by

$$
\begin{aligned}
x_N(\sigma) &= x_{i-1}+\Delta x\theta^x_{i-1,j}+\sigma(1+\theta^x_{i,j}-\theta^x_{i-1,j})\Delta x,\\
y_N(\sigma) &= y_j+\Delta y\theta^y_{i-1,j}+\sigma(\theta^y_{i,j}-\theta^y_{i-1,j})\Delta y, \qquad \sigma\in[0,1].
\end{aligned}
$$

Let σ^* be the intersecting parameter such that $x_N(\sigma^*)=\bar{x}_i$. So, integrating along γ^N gives

$$
\int_{\gamma^N}\langle W^x,W^y\rangle\cdot\vec{n}ds=\Big(\int_0^{\sigma^*}+\int_{\sigma^*}^{1}\Big)[-W^x(x_N(\sigma),y_N(\sigma))y'_N(\sigma)+W^y(x_N(\sigma),y_N(\sigma))x'_N(\sigma)]d\sigma.
\tag{3.24}
$$

We ignore the first integral. The function $W^y(x,y)$ is given by

$$W^y(x,y) = \begin{cases} p_{i-1,j}(x)(y-\bar{y}_j) + \frac{1}{2\Delta y}(y-\bar{y}_j)^2(p_{i-1,j+1}(x) - p_{i-1,j}(x)), \\ \hfill (x,y) \in R^d_{i,j+1} \\ p_{i,j}(x)(y-\bar{y}_j) + \frac{1}{2\Delta y}(y-\bar{y}_j)^2(p_{i,j+1}(x) - p_{i,j}(x)), \\ \hfill (x,y) \in R^d_{i+1,j+1} \end{cases}.$$

Applying the trapezoidal rule, we get

$$\Big(\int_0^{\sigma^*} + \int_{\sigma^*}^1\Big) W^y(x_N(\sigma), y_N(\sigma))x_N'(\sigma)d\sigma \cong \frac{1}{2}x_N'\{\sigma^* W_0^y + (1-\sigma^*)W_1^y + W_{\sigma^*}^y\} \tag{3.25}$$

where

$$\begin{aligned} W_0^y &= p_{i-1,j}(x_N(0))(y_N(0)-\bar{y}_j) + \frac{1}{2\Delta y}(y_N(0)-\bar{y}_j)^2(p_{i-1,j+1}(x_N(0)) - p_{i-1,j}(x_N(0))), \\ W_{\sigma^*}^y &= p_{i-1,j}(x_N(\sigma^*))(y_N(\sigma^*)-\bar{y}_j) + \frac{1}{2\Delta y}(y_N(\sigma^*)-\bar{y}_j)^2(p_{i-1,j+1}(x_N(\sigma^*)) - p_{i-1,j}(x_N(\sigma^*))), \\ W_1^y &= p_{i,j}(x_N(1))(y_N(1)-\bar{y}_j) + \frac{1}{2\Delta y}(y_N(1)-\bar{y}_j)^2(p_{i,j+1}(x_N(1)) - p_{i,j}(x_N(1))). \end{aligned}$$

The above equations involve the following equalities

$$\begin{aligned} x_N(0) - \bar{x}_{i-1} &= (\tfrac{1}{2} + \theta^x_{i-1,j})\Delta x, \\ x_N(0) - \bar{x}_i &= (\theta^x_{i-1,j} - \tfrac{1}{2})\Delta x, \\ x_N(1) - \bar{x}_i &= (\tfrac{1}{2} + \theta^x_{i,j})\Delta x, \\ y_N(0) - \bar{y}_j &= (\tfrac{1}{2} + \theta^y_{i-1,j})\Delta y, \\ y_N(\sigma^*) - \bar{y}_j &= (\tfrac{1}{2} + \theta^y_{i-1,j})\Delta y + \sigma^*(\theta^y_{i,j} - \theta^y_{i-1,j}) \\ &= (\tfrac{1}{2} + \theta^y_{i,j})\Delta y + (1-\sigma^*)(\theta^y_{i-1,j} - \theta^y_{i,j}), \\ y_N(1) - \bar{y}_j &= (\tfrac{1}{2} + \theta^y_{i,j})\Delta y \end{aligned}$$

$$p_{i-1,j+1}(x_N(0)) = (\tfrac{1}{2} - \theta^x_{i-1,j})U_{i-1,j+1} + (\tfrac{1}{2} + \theta^x_{i-1,j})U_{i,j+1},$$
$$p_{i-1,j}(x_N(0)) = (\tfrac{1}{2} - \theta^x_{i-1,j})U_{i-1,j} + (\tfrac{1}{2} + \theta^x_{i-1,j})U_{i,j},$$
$$p_{i,j+1}(x_N(1)) = (\tfrac{1}{2} - \theta^x_{i,j})U_{i,j+1} + (\tfrac{1}{2} + \theta^x_{i,j})U_{i+1,j+1},$$
$$p_{i,j}(x_N(1)) = (\tfrac{1}{2} - \theta^x_{i,j})U_{i,j} + (\tfrac{1}{2} + \theta^x_{i,j})U_{i+1,j}.$$

By the above equalities we can write the following equations

$$\begin{aligned}
W^y_0 &= (\tfrac{1}{2} + \theta^y_{i-1,j})\Delta y\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})U_{i-1,j} + (\tfrac{1}{2} + \theta^x_{i-1,j})U_{i,j}\Big] \\
&\quad + \tfrac{1}{2}\Delta y(\tfrac{1}{2} + \theta^y_{i-1,j})^2\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})U_{i-1,j+1} + (\tfrac{1}{2} + \theta^x_{i-1,j})U_{i,j+1} \\
&\quad - (\tfrac{1}{2} - \theta^x_{i-1,j})U_{i-1,j} - (\tfrac{1}{2} + \theta^x_{i-1,j})U_{i,j}\Big] \\
&= \tfrac{1}{2}\Delta y\Big\{U_{i-1,j}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})(\tfrac{1}{2} + \theta^y_{i-1,j})(\tfrac{3}{2} - \theta^y_{i-1,j})\Big] \\
&\quad + U_{i,j}\Big[(\tfrac{1}{2} + \theta^x_{i-1,j})(\tfrac{1}{2} + \theta^y_{i-1,j})(\tfrac{3}{2} - \theta^y_{i-1,j})\Big] \\
&\quad + U_{i-1,j+1}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})(\tfrac{1}{2} + \theta^y_{i-1,j})^2\Big] + U_{i,j+1}\Big[(\tfrac{1}{2} + \theta^x_{i-1,j})(\tfrac{1}{2} + \theta^y_{i-1,j})^2\Big]\Big\},
\end{aligned}$$

$$\begin{aligned}
W^y_{\sigma^*} &= \tfrac{1}{2}\Delta y\Big\{\big[(\tfrac{1}{2} + \theta^y_{i-1,j} + \sigma^*(\theta^y_{i,j} - \theta^y_{i-1,j}))(\tfrac{3}{2} - \theta^y_{i-1,j} + \sigma^*(\theta^y_{i,j} - \theta^y_{i-1,j}))\big]U_{i,j} \\
&\qquad + (\tfrac{1}{2} + \theta^y_{i-1,j} + \sigma^*(\theta^y_{i,j} - \theta^y_{i-1,j}))^2 U_{i,j+1}\Big\}, \\
&= \tfrac{1}{2}\Delta y\Big\{\big[(\tfrac{1}{2} + \theta^y_{i,j} + (1-\sigma^*)(\theta^y_{i-1,j} - \theta^y_{i,j}))(\tfrac{3}{2} - \theta^y_{i,j} + (1-\sigma^*)(\theta^y_{i-1,j} - \theta^y_{i,j}))\big]U_{i,j} \\
&\qquad + (\tfrac{1}{2} + \theta^y_{i,j} + (1-\sigma^*)(\theta^y_{i-1,j} - \theta^y_{i,j}))^2 U_{i,j+1}\Big\},
\end{aligned}$$

and

$$\begin{aligned}
W^y_1 = \tfrac{1}{2}\Delta y\Big\{&U_{i,j}\big[(\tfrac{1}{2} - \theta^x_{i,j})(\tfrac{1}{2} + \theta^y_{i,j})(\tfrac{3}{2} - \theta^y_{i,j})\big] + U_{i+1,j}\big[(\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{1}{2} + \theta^y_{i,j})(\tfrac{3}{2} - \theta^y_{i,j})\big] \\
&+ U_{i,j+1}\big[(\tfrac{1}{2} - \theta^x_{i,j})(\tfrac{1}{2} + \theta^y_{i,j})^2\big] + U_{i+1,j+1}\big[(\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{1}{2} + \theta^y_{i,j})^2\big]\Big\}.
\end{aligned}$$

From the equality $x_N(\sigma^*) = \bar{x}_i$ we get that $\sigma^* x'_N = (\frac{1}{2} - \theta^x_{i-1,j})\Delta x$. Hence, by combining this with the above equations we get the integral along the north edge

$$
\begin{aligned}
&\Big(\int_0^{\sigma^*} + \int_{\sigma^*}^{1}\Big) W^y(x_N(\sigma), y_N(\sigma)) x'_N(\sigma) d\sigma \\
\approx &\quad \tfrac{1}{2}\Delta x \Big\{ (\tfrac{1}{2} - \theta^x_{i-1,j}) W^y_0 + (\tfrac{1}{2} + \theta^x_{i,j}) W^y_1 + \big[(\tfrac{1}{2} - \theta^x_{i-1,j}) + (\tfrac{1}{2} + \theta^x_{i,j})\big] W^y_{\sigma^*} \Big\} \\
\cong &\ \tfrac{1}{4}\Delta x \Delta y \Big\{ U_{i-1,j} \Big[(\tfrac{1}{2} - \theta^x_{i-1,j})^2 (\tfrac{1}{2} + \theta^y_{i-1,j})(\tfrac{3}{2} - \theta^y_{i-1,j}) \Big] \\
&+ U_{i-1,j+1} \Big[(\tfrac{1}{2} - \theta^x_{i-1,j})^2 (\tfrac{1}{2} + \theta^y_{i-1,j})^2 \Big] \\
&\quad + U_{i,j} \Big[(\tfrac{1}{2} - \theta^x_{i-1,j})(\tfrac{3}{2} + \theta^x_{i-1,j})(\tfrac{1}{2} + \theta^y_{i-1,j})(\tfrac{3}{2} - \theta^y_{i-1,j}) \\
&\qquad\qquad + (\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{3}{2} - \theta^x_{i,j})(\tfrac{1}{2} + \theta^y_{i,j})(\tfrac{3}{2} - \theta^y_{i,j}) \Big] \\
&+ U_{i,j+1} \Big[(\tfrac{1}{2} - \theta^x_{i-1,j})(\tfrac{3}{2} + \theta^x_{i-1,j})(\tfrac{1}{2} + \theta^y_{i-1,j})^2 + (\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{3}{2} - \theta^x_{i,j})(\tfrac{1}{2} + \theta^y_{i,j})^2 \Big] \\
&+ U_{i+1,j} \Big[(\tfrac{1}{2} + \theta^x_{i,j})^2 (\tfrac{1}{2} + \theta^y_{i,j})(\tfrac{3}{2} - \theta^y_{i,j}) \Big] + U_{i+1,j+1} \Big[(\tfrac{1}{2} + \theta^x_{i,j})^2 (\tfrac{1}{2} + \theta^y_{i,j})^2 \Big] \Big\}
\end{aligned}
\tag{3.26}
$$

Now we add the equations (3.16), (3.20), (3.23) and (3.26) to get

$$\iint\limits_{R_{i,j}(t_n)} U(x,y,t_n)dA \approx \frac{1}{2}\Big\{ \int\limits_{\gamma^E} W^x y' dy + \int\limits_{\gamma^N} W^y x' dx - \int\limits_{\gamma^W} W^x y' dy - \int\limits_{\gamma^S} W^y x' dy \Big\}$$

$$\begin{aligned}
&\approx \frac{1}{8}\Delta x \Delta y \Big\{ \\
&2U_{i-1,j+1}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j})^2(\tfrac{1}{2}+\theta^y_{i-1,j})^2\Big] + 2U_{i+1,j+1}\Big[(\tfrac{1}{2}+\theta^x_{i,j})^2(\tfrac{1}{2}+\theta^y_{i,j})^2\Big] \\
&+2U_{i,j+1}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j})(\tfrac{3}{2}+\theta^x_{i-1,j})(\tfrac{1}{2}+\theta^y_{i-1,j})^2 + (\tfrac{1}{2}+\theta^x_{i,j})(\tfrac{3}{2}-\theta^x_{i,j})(\tfrac{1}{2}+\theta^y_{i,j})^2\Big] \\
&+2U_{i-1,j}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j-1})^2(\tfrac{3}{2}+\theta^y_{i-1,j-1})(\tfrac{1}{2}-\theta^y_{i-1,j-1}) + (\tfrac{1}{2}-\theta^x_{i-1,j})^2(\tfrac{3}{2}-\theta^y_{i-1,j})(\tfrac{1}{2}+\theta^y_{i-1,j})\Big] \\
&+2U_{i,j}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j})(\tfrac{3}{2}+\theta^x_{i-1,j})(\tfrac{1}{2}+\theta^y_{i-1,j})(\tfrac{3}{2}-\theta^y_{i-1,j}) \\
&\qquad + (\tfrac{1}{2}+\theta^x_{i,j})(\tfrac{3}{2}-\theta^x_{i,j})(\tfrac{1}{2}+\theta^y_{i,j})(\tfrac{3}{2}-\theta^y_{i,j}) \\
&\qquad + (\tfrac{1}{2}-\theta^x_{i-1,j-1})(\tfrac{3}{2}+\theta^x_{i-1,j-1})(\tfrac{1}{2}-\theta^y_{i-1,j-1})(\tfrac{3}{2}+\theta^y_{i-1,j-1}) \\
&\qquad + (\tfrac{1}{2}+\theta^x_{i,j-1})(\tfrac{3}{2}-\theta^x_{i,j-1})(\tfrac{1}{2}-\theta^y_{i,j-1})(\tfrac{3}{2}+\theta^y_{i,j-1})\Big] \\
&+2U_{i+1,j}\Big[(\tfrac{1}{2}+\theta^x_{i,j-1})^2(\tfrac{1}{2}+\theta^y_{i,j-1})(\tfrac{3}{2}-\theta^y_{i,j-1}) + (\tfrac{1}{2}+\theta^x_{i,j})^2(\tfrac{1}{2}+\theta^y_{i,j})(\tfrac{3}{2}-\theta^y_{i,j})\Big] \\
&+2U_{i-1,j-1}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j-1})^2(\tfrac{1}{2}-\theta^y_{i-1,j-1})^2\Big] + 2U_{i+1,j-1}\Big[(\tfrac{1}{2}+\theta^x_{i,j-1})^2(\tfrac{1}{2}-\theta^y_{i,j-1})^2\Big] \\
&+2U_{i,j-1}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j-1})(\tfrac{3}{2}+\theta^x_{i-1,j-1})(\tfrac{1}{2}-\theta^y_{i-1,j-1})^2 + (\tfrac{1}{2}+\theta^x_{i,j-1})(\tfrac{3}{2}-\theta^x_{i,j-1})(\tfrac{1}{2}-\theta^y_{i,j-1})^2\Big].
\end{aligned}$$

With some algebraic manipulation, the above equation can be written as

$$\iint\limits_{R_{i,j}(t_n)} U(x,y,t_n)dA \approx \frac{1}{2}\Big\{ \int\limits_{\gamma^E} W^x y' dy + \int\limits_{\gamma^N} W^y x' dx - \int\limits_{\gamma^W} W^x y' dy - \int\limits_{\gamma^S} W^y x' dy \Big\}$$

$$\approx \frac{1}{8}\Delta x \Delta y \Bigg\{ 2U_{i-1,j-1}\Bigg[\left[\tfrac{1}{4} - \theta^x_{i-1,j}(1-\theta^x_{i-1,j})\right]\left[\tfrac{1}{4} + \theta^y_{i-1,j}(1+\theta^y_{i-1,j})\right]\Bigg]$$

$$+2U_{i+1,j+1}\Bigg[\left[\tfrac{1}{4} + \theta^x_{i,j}(1+\theta^x_{i-1,j})\right]\left[\tfrac{1}{4} + \theta^y_{i,j}(1+\theta^y_{i,j})\right]\Bigg]$$

$$+2U_{i,j+1}\Bigg[\left[\tfrac{6}{4} - \theta^x_{i-1,j}(1+\theta^x_{i-1,j}) + \theta^x_{i,j}(1-\theta^x_{i,j})\right]\left[\tfrac{1}{4} + \theta^y_{i,j}(1+\theta^y_{i,j})\right]\Bigg]$$

$$+2U_{i-1,j}\Bigg[\left[\tfrac{1}{4} - \theta^x_{i-1,j-1}(1-\theta^x_{i-1,j-1})\right]\left[\tfrac{3}{4} - \theta^y_{i-1,j-1}(1+\theta^y_{i-1,j-1})\right]$$

$$+ \left[\tfrac{1}{4} - \theta^x_{i-1,j}(1-\theta^x_{i-1,j})\right]\left[\tfrac{3}{4} + \theta^y_{i-1,j}(1-\theta^y_{i-1,j})\right]\Bigg]$$

$$+2U_{i-1,j-1}\Bigg[\left[\tfrac{1}{4} - \theta^x_{i-1,j-1}(1-\theta^x_{i-1,j-1})\right]\left[\tfrac{1}{4} - \theta^y_{i-1,j-1}(1-\theta^y_{i-1,j-1})\right]\Bigg]$$

$$+2U_{i+1,j-1}\Bigg[\left[\tfrac{1}{4} + \theta^x_{i,j-1}(1+\theta^x_{i,j-1})\right]\left[\tfrac{1}{4} - \theta^y_{i,j-1}(1-\theta^y_{i,j-1})\right]\Bigg]$$

$$+2U_{i,j-1}\Bigg[\left[\tfrac{3}{4} - \theta^x_{i-1,j-1}(1-\theta^x_{i-1,j-1})\right]\left[\tfrac{1}{4} - \theta^y_{i-1,j-1}(1-\theta^y_{i-1,j-1})\right]$$

$$+ \left[\tfrac{3}{4} + \theta^x_{i,j-1}(1-\theta^x_{i,j-1})\right]\left[\tfrac{1}{4} - \theta^y_{i,j-1}(1-\theta^y_{i,j-1})\right]\Bigg]$$

$$+2U_{i+1,j}\Bigg[\left[\tfrac{1}{4} + \theta^x_{i,j-1}(1+\theta^x_{i,j-1})\right]\left[\tfrac{3}{4} + \theta^y_{i,j-1}(1-\theta^y_{i,j-1})\right]$$

$$+ \left[\tfrac{1}{4} + \theta^x_{i,j}(1+\theta^x_{i,j})\right]\left[\tfrac{3}{4} + \theta^y_{i,j}(1-\theta^y_{i,j})\right]\Bigg]$$

$$+2U_{i,j}\Bigg[\left[\tfrac{3}{4} - \theta^x_{i-1,j}(1+\theta^x_{i-1,j})\right]\left[\tfrac{3}{4} + \theta^y_{i-1,j}(1-\theta^y_{i-1,j})\right]$$

$$+ \left[\tfrac{3}{4} + \theta^x_{i,j}(1-\theta^x_{i,j})\right]\left[\tfrac{3}{4} + \theta^y_{i,j}(1-\theta^y_{i,j})\right]$$

$$+ \left[\tfrac{3}{4} - \theta^x_{i-1,j-1}(1-\theta^x_{i-1,j-1})\right]\left[\tfrac{3}{4} - \theta^y_{i-1,j-1}(1-\theta^y_{i-1,j-1})\right]$$

$$+ \left[\tfrac{3}{4} + \theta^x_{i,j-1}(1-\theta^x_{i,j-1})\right]\left[\tfrac{3}{4} - \theta^y_{i,j-1}(1+\theta^y_{i,j-1})\right]\Bigg]\Bigg\}.$$

For convenience, we set

$$a_{i,j} = \Theta^x_{i,j}(1-\Theta^x_{i,j}), \qquad A_{i,j} = \Theta^x_{i,j}(1+\Theta^x_{i,j}),$$

$$b_{i,j} = \Theta^y_{i,j}(1-\Theta^y_{i,j}), \qquad B_{i,j} = \Theta^y_{i,j}(1+\Theta^y_{i,j}),$$

where $\Theta^{x,n}_{i,j} \approx \theta^{x,n}_{i,j}$ and $\Theta^{y,n}_{i,j} \approx \theta^{y,n}_{i,j}$.

So, the above equation can be written in a compact form as

$$\iint\limits_{R_{i,j}(t_n)} U(x,y,t_n)dA = \frac{1}{2}\Big\{ \int\limits_{\gamma^E} W^x y' dy + \int\limits_{\gamma^N} W^y x' dx - \int\limits_{\gamma^W} W^x y' dy - \int\limits_{\gamma^S} W^y x' dy \Big\}$$

$$\begin{aligned}
&\cong \frac{1}{8}\Delta x \Delta y \Big\{ \\
&\frac{1}{8}U_{i-1,j+1} + \frac{6}{8}U_{i-1,j} + \frac{1}{8}U_{i-1,j-1} + \frac{6}{8}U_{i,j-1} + \frac{36}{8}U_{i,j} + \frac{6}{8}U_{i,j+1} + \frac{1}{8}U_{i+1,j-1} \\
&+ \frac{6}{8}U_{i+1,j} + \frac{1}{8}U_{i+1,j+1} \\
&+ \frac{3}{2}\Big[-(a_{i-1,j-1}U_{i-1,j} + A_{i-1,j-1}U_{i,j}) + (a_{i,j-1}U_{i,j} + A_{i,j-1}U_{i+1,j}) \Big] \\
&+ \frac{3}{2}\Big[-(a_{i-1,j}U_{i-1,j} + A_{i-1,j}U_{i,j}) + (a_{i,j}U_{i,j} + A_{i,j}U_{i+1,j}) \Big] \\
&+ \frac{1}{2}\Big[-(a_{i-1,j-1}U_{i-1,j-1} + A_{i-1,j-1}U_{i,j-1}) + (a_{i,j-1}U_{i,j-1} + A_{i,j-1}U_{i+1,j-1}) \Big] \\
&+ \frac{1}{2}\Big[-(a_{i-1,j}U_{i-1,j+1} + A_{i-1,j}U_{i,j+1}) + (a_{i,j}U_{i,j+1} + A_{i,j}U_{i+1,j+1}) \Big] \\
&+ \frac{3}{2}\Big[-(b_{i-1,j-1}U_{i,j-1} + B_{i-1,j-1}U_{i,j}) + (b_{i-1,j}U_{i,j} + B_{i-1,j}U_{i,j+1}) \Big] \\
&+ \frac{3}{2}\Big[-(b_{i,j-1}U_{i,j-1} + B_{i,j-1}U_{i,j}) + (b_{i,j}U_{i,j} + B_{i,j}U_{i,j+1}) \Big] \\
&+ \frac{1}{2}\Big[-(b_{i-1,j-1}U_{i-1,j-1} + B_{i-1,j-1}U_{i-1,j}) + (b_{i-1,j}U_{i-1,j} + B_{i-1,j}U_{i-1,j+1}) \Big] \\
&+ \frac{1}{2}\Big[-(b_{i,j-1}U_{i+1,j-1} + B_{i,j-1}U_{i+1,j}) + (b_{i,j}U_{i+1,j} + B_{i,j}U_{i+1,j+1}) \Big] \\
&- 2\Big[a_{i-1,j}b_{i-1,j}U_{i-1,j} + a_{i-1,j}B_{i-1,j}U_{i-1,j+1} + A_{i-1,j}b_{i-1,j}U_{i,j} + A_{i-1,j}B_{i-1,j}U_{i,j+1} \Big] \\
&+ 2\Big[a_{i,j}b_{i,j}U_{i,j} + a_{i,j}B_{i,j}U_{i,j+1} + A_{i,j}b_{i,j}U_{i+1,j} + A_{i,j}B_{i,j}U_{i+1,j+1} \Big] \\
&- 2\Big[a_{i,j-1}b_{i,j-1}U_{i,j-1} + a_{i,j-1}B_{i,j-1}U_{i,j} + A_{i,j-1}b_{i,j-1}U_{i+1,j-1} + A_{i,j-1}B_{i,j-1}U_{i+1,j} \Big] \\
&+ 2\Big[a_{i-1,j}b_{i-1,j-1}U_{i-1,j-1} + a_{i-1,j-1}B_{i-1,j-1}U_{i-1,j} + A_{i-1,j-1}b_{i-1,j-1}U_{i,j-1} \\
&\qquad + A_{i-1,j-1}B_{i-1,j-1}U_{i,j} \Big] \Big\}.
\end{aligned} \tag{3.27}$$

For the benefit of verifying the mass conservative property it is good to write the equation (3.27) in a conservative form. To distinguish between the forward and backward values we write the super-scripts n. Let

$$F^{x,n}_{i,j} = a^n_{i,j}U^n_{i,j} + A^n_{i,j}U^n_{i+1,j}, \qquad F^{y,n}_{i,j} = b^n_{i,j}U^n_{i,j} + B^n_{i,j}U^n_{i,j+1},$$
$$\tilde{F}^{x,n}_{i,j} = a^n_{i,j-1}U^n_{i,j} + A^n_{i,j-1}U^n_{i+1,j}, \qquad \tilde{F}^{y,n}_{i,j} = b^n_{i-1,j}U^n_{i,j} + B^n_{i-1,j}U^n_{i,j+1}$$

and

$$H^n_{i,j} = a^n_{i,j}b^n_{i,j}U^n_{i,j} + a^n_{i,j}B^n_{i,j}U^n_{i,j+1} + A^n_{i,j}b^n_{i,j}U^n_{i+1,j} + A^n_{i,j}B^n_{i,j}U^n_{i+1,j+1}.$$

Now the equation (3.27) is written as

$$\begin{aligned}
&\iint\limits_{R_{i,j}(t_n)} U(x,y,t_n) \\
&\qquad \approx \frac{1}{8}\Delta x\Delta y\Big\{\frac{1}{8}U^n_{i-1,j+1} + \frac{6}{8}U^n_{i-1,j} + \frac{1}{8}U^n_{i-1,j-1} + \frac{6}{8}U^n_{i,j-1} + \frac{36}{8}U^n_{i,j} + \frac{6}{8}U^n_{i,j+1} \\
&\qquad + \frac{1}{8}U^n_{i+1,j-1} + \frac{6}{8}U^n_{i+1,j} + \frac{1}{8}U^n_{i+1,j+1} + 2(H^n_{i,j} - H^n_{i,j-1} + H^n_{i-1,j-1} - H^n_{i-1,j}) \\
&\qquad + \frac{1}{2}(F^{x,n}_{i,j-1} - F^{x,n}_{i-1,j-1} + \tilde{F}^{x,n}_{i,j+1} - \tilde{F}^{x,n}_{i-1,j+1}) + \frac{3}{2}(F^{x,n}_{i,j} - F^{x,n}_{i-1,j} + \tilde{F}^{x,n}_{i,j} - \tilde{F}^{x,n}_{i-1,j}) \\
&\qquad + \frac{1}{2}(F^{y,n}_{i-1,j} - F^{y,n}_{i-1,j-1} + \tilde{F}^{y,n}_{i+1,j} - \tilde{F}^{y,n}_{i+1,j-1}) + \frac{3}{2}(F^{y,n}_{i,j} - F^{y,n}_{i,j-1} + \tilde{F}^{y,n}_{i,j} - \tilde{F}^{y,n}_{i,j-1})\Big\}.
\end{aligned}$$

Remark 3.3: *If $x_{i,j}(t_n) = x_i$ and $y_{i,j}(t_n) = y_j$ then $\theta^x_{i,j} = \theta^y_{i,j} = 0$ for all i and j. Then we have*

$$\begin{aligned}
&\iint\limits_{R_{i,j}} U(x,y,t_n) \\
&\qquad \approx \frac{1}{8}\Delta x\Delta y\Big\{\frac{1}{8}U^n_{i-1,j+1} + \frac{6}{8}U^n_{i-1,j} + \frac{1}{8}U^n_{i-1,j-1} + \frac{6}{8}U^n_{i,j-1} + \frac{36}{8}U^n_{i,j} + \frac{6}{8}U^n_{i,j+1} \\
&\qquad + \frac{1}{8}U^n_{i+1,j-1} + \frac{6}{8}U^n_{i+1,j} + \frac{1}{8}U^n_{i+1,j+1},
\end{aligned}$$

which is a quadrature for the bilinear approximating functions.

3.1.5 Approximation of the diffusion term

To complete the scheme we must include the diffusion and source terms. The diffusion term is treated in the same way discussed above. applying the trapezoidal rule and

Divergence Theorem on the diffusion term gives

$$\iiint\limits_{Q_{i,j}^{n+1}} \nabla\cdot(\kappa\nabla u)dV \approx \tfrac{1}{2}\Delta t\Big(\iint\limits_{R_{i,j}} \nabla\cdot(\kappa\nabla u)|_{t_{n+1}}dA + \iint\limits_{R_{i,j}(t_n)} \nabla\cdot(\kappa\nabla u)|_{t_n}dA\Big)$$
$$= \tfrac{1}{2}\Delta t\Big(\int\limits_{\partial R_{i,j}} (\kappa\nabla u)|_{t_{n+1}}\cdot\vec{n}\ ds + \int\limits_{\partial R_{i,j}(t_n)} (\kappa\nabla u)|_{t_n}\cdot\vec{n}\ ds\Big).$$

By the same argument we compute the line integral along the boundaries of the departure cell $\partial R_{i,j}(t_n)$. Along the east edge γ^E we have

$$\int\limits_{\gamma^E} \nabla U|_{t_n}\cdot\vec{n}\ ds = \int\limits_0^1 \big[U_x(x_E(\sigma), y_E(\sigma))y_E'(\sigma) - U_y(x_E(\sigma), y_E(\sigma))x_E'(\sigma)\big]d\sigma.$$

As we did before we will ignore the second integral because of higher order of the increments. Also, we let σ^* be the intersection parameter of the parametrizing equations and the line $y_E(\sigma^*) = \bar{y}_j$. We write $U_x^0 = U_x^0(x_E(0), y_E(0))$, $U_x^1 = U(x_E(1), y_E(1))$, $U_x^{\sigma^*} = U_x(x_E(\sigma^*), y_E(\sigma^*))$ and $p_{i,j}'(x) = p_{i,j}'$. The function $U_x(x, y)$ is given by

$$U_x(x,y) = \begin{cases} p_{i,j-1}' + \dfrac{y-\bar{y}_{j-1}}{\Delta y}[p_{i,j}' - p_{i,j-1}'], & (x,y)\in R_{i+1,j}^d \\ p_{i,j}' + \dfrac{y-\bar{y}_j}{\Delta y}[p_{i,j+1}' - p_{i,j}'], & (x,y)\in R_{i+1,j+1}^d. \end{cases}$$

Applying the trapezoidal rule, we obtain

$$\int\limits_0^1 U_x y_E'(\sigma)d\sigma = \frac{1}{2}y_E'\big\{\sigma^* U_x^0 + (1-\sigma^*)U_x^1 + U_x^{\sigma^*}\big\},$$

where

$$U_x^0 = (\frac{1}{2} - \theta_{i,j-1}^y)p_{i,j-1}' + (\frac{1}{2} + \theta_{i,j-1}^y)p_{i,j}',$$
$$U_x^1 = (\frac{1}{2} - \theta_{i,j}^y)p_{i,j}' + (\frac{1}{2} + \theta_{i,j}^y)p_{i,j+1}',$$
$$U_x^{\sigma^*} = p_{i,j}'(x_E(\sigma^*)).$$

Hence

$$
\begin{aligned}
\int\limits_{\gamma^E} U_x y_E' \; ds &\approx \tfrac{1}{2}\Delta y\Big\{(\tfrac{1}{2}-\theta^y_{i,j-1})^2 p'_{i,j-1} + (\tfrac{1}{2}-\theta^y_{i,j-1})(\tfrac{3}{2}+\theta^y_{i,j-1})p'_{i,j} \\
&\qquad + (\tfrac{1}{2}+\theta^y_{i,j})(\tfrac{3}{2}-\theta^y_{i,j})p'_{i,j} + (\tfrac{1}{2}+\theta^y_{i,j})^2 p'_{i,j+1}\Big\} \\
&\approx \tfrac{1}{2}\frac{\Delta y}{\Delta x}\Big\{U_{i+1,j-1}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})^2\Big] - U_{i,j-1}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})^2\Big] \\
&\quad -U_{i,j}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})(\tfrac{3}{2}+\theta^y_{i,j-1}) + (\tfrac{1}{2}+\theta^y_{i,j})(\tfrac{3}{2}-\theta^y_{i,j})\Big] \\
&\quad +U_{i+1,j}\Big[(\tfrac{1}{2}-\theta^y_{i,j-1})(\tfrac{3}{2}+\theta^y_{i,j-1}) + (\tfrac{1}{2}+\theta^y_{i,j})(\tfrac{3}{2}-\theta^y_{i,j})\Big] \\
&\quad +U_{i+1,j+1}\Big[(\tfrac{1}{2}+\theta^y_{i,j})^2\Big] - U_{i,j+1}\Big[(\tfrac{1}{2}+\theta^y_{i,j})^2\Big]\Big\}.
\end{aligned}
\tag{3.28}
$$

Similarly, integrating along the west edge γ^W gives

$$
\begin{aligned}
\int\limits_{\gamma^W} U_x y_W' \; d\sigma &\approx \tfrac{1}{2}\Delta y\Big\{(\tfrac{1}{2}-\theta^y_{i-1,j-1})^2 p'_{i-1,j-1} + (\tfrac{1}{2}-\theta^y_{i-1,j-1})(\tfrac{3}{2}+\theta^y_{i-1,j-1})p'_{i-1,j} \\
&\qquad + (\tfrac{1}{2}+\theta^y_{i-1,j})(\tfrac{3}{2}-\theta^y_{i-1,j})p'_{i-1,j} + (\tfrac{1}{2}+\theta^y_{i-1,j})^2 p'_{i-1,j+1}\Big\} \\
&\approx \tfrac{1}{2}\frac{\Delta y}{\Delta x}\Big\{U_{i,j-1}\Big[(\tfrac{1}{2}-\theta^y_{i-1,j-1})^2\Big] - U_{i-1,j-1}\Big[(\tfrac{1}{2}-\theta^y_{i-1,j-1})^2\Big] \\
&\quad +U_{i,j}\Big[(\tfrac{1}{2}-\theta^y_{i-1,j-1})(\tfrac{3}{2}+\theta^y_{i-1,j-1}) + (\tfrac{1}{2}+\theta^y_{i-1,j})(\tfrac{3}{2}-\theta^y_{i-1,j})\Big] \\
&\quad -U_{i-1,j}\Big[(\tfrac{1}{2}-\theta^y_{i-1,j-1})(\tfrac{3}{2}+\theta^y_{i-1,j-1}) + (\tfrac{1}{2}+\theta^y_{i-1,j})(\tfrac{3}{2}-\theta^y_{i-1,j})\Big] \\
&\quad +U_{i,j+1}\Big[(\tfrac{1}{2}+\theta^y_{i-1,j})^2\Big] - U_{i-1,j+1}\Big[(\tfrac{1}{2}+\theta^y_{i-1,j})^2\Big]\Big\}.
\end{aligned}
\tag{3.29}
$$

Integrating along the north edge γ^N leads to

$$\int\limits_{\gamma^N} \nabla U|_{t_n} \cdot \vec{n} \; ds = \int\limits_0^1 \big[-U_x(x_N(\sigma), y_N(\sigma))y_N'(\sigma) + U_y(x_N(\sigma), y_N(\sigma))x'(\sigma)]d\sigma.$$

By ignoring the first integral we have

$$\int\limits_0^1 U_y x_N' \; d\sigma \approx \tfrac{1}{2}x_N'\big\{\sigma^* U_y^0 + (1-\sigma^*)U_y^1 + U_y(\sigma^*)\big\},$$

where

$$U_y(x,y) = \begin{cases} q'_{i-1,j} + \dfrac{x - \bar{x}_{j-1}}{\Delta x}[q'_{i,j} - q'_{i-1,j}], & (x,y) \in R^d_{1,j+1} \\ q'_{i,j} + \dfrac{q - \bar{q}_i}{\Delta x}[q'_{i+1,j} - q'_{i,j}], & (x,y) \in R^d_{2,j+1} \end{cases}$$

So we have

$$\begin{aligned} U_y^0 &= (\tfrac{1}{2} - \theta^x_{i-1,j})q'_{i-1,j} + (\tfrac{1}{2} + \theta^x_{i-1,j})q'_{i,j}, \\ U_y^1 &= (\tfrac{1}{2} - \theta^x_{i,j})q'_{i,j} + (\tfrac{1}{2} + \theta^x_{i,j})q'_{i+1,j}, \\ U_y(\sigma^*) &= q'_{i,j}. \end{aligned}$$

Then, the integral is given by

$$\begin{aligned} \int\limits_{\gamma^N} U_y x'_N \, d\sigma \approx \tfrac{1}{2}\Delta x \Big\{ &(\tfrac{1}{2} - \theta^x_{i-1,j})^2 q'_{i-1,j} + (\tfrac{1}{2} - \theta^x_{i-1,j})(\tfrac{3}{2} + \theta^x_{i-1,j})q'_{i,j} \\ &+ (\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{3}{2} - \theta^x_{i,j})q'_{i,j} + (\tfrac{1}{2} + \theta^x_{i,j})^2 q'_{i+1,j} \Big\} \\ \approx \tfrac{1}{2}\tfrac{\Delta x}{\Delta y} \Big\{ &U_{i-1,j+1}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})^2\Big] - U_{i-1,j}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})^2\Big] \\ &-U_{i,j}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})(\tfrac{3}{2} + \theta^x_{i-1,j}) + (\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{3}{2} - \theta^x_{i,j})\Big] \\ &+U_{i,j+1}\Big[(\tfrac{1}{2} - \theta^x_{i-1,j})(\tfrac{3}{2} + \theta^x_{i-1,j}) + (\tfrac{1}{2} + \theta^x_{i,j})(\tfrac{3}{2} - \theta^y_{i,j})\Big] \\ &+U_{i+1,j+1}\Big[(\tfrac{1}{2} + \theta^x_{i,j})^2\Big] - U_{i+1,j}\Big[(\tfrac{1}{2} + \theta^x_{i,j})^2\Big] \Big\}. \end{aligned} \tag{3.30}$$

The last integral will be along the south edge γ^S that is similar to the integral along the north edge with the jth index shifted down one unit. Hence

$$
\begin{aligned}
\int\limits_{\gamma^S} U_y x'_S \, d\sigma \approx \tfrac{1}{2}\Delta x \Big\{ & (\tfrac{1}{2}-\theta^x_{i-1,j-1})^2 q'_{i-1,j-1} + (\tfrac{1}{2}-\theta^x_{i-1,j-1})(\tfrac{3}{2}+\theta^x_{i-1,j-1}) q'_{i,j-1} \\
& + (\tfrac{1}{2}+\theta^x_{i,j-1})(\tfrac{3}{2}-\theta^x_{i,j-1}) q'_{i,j-1} + (\tfrac{1}{2}+\theta^x_{i,j-1})^2 q'_{i+1,j-1} \Big\} \\
\approx \tfrac{1}{2}\tfrac{\Delta x}{\Delta y} \Big\{ & U_{i-1,j}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j-1})^2\Big] - U_{i-1,j-1}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j-1})^2\Big] \\
& + U_{i,j}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j-1})(\tfrac{3}{2}+\theta^x_{i-1,j-1}) + (\tfrac{1}{2}+\theta^x_{i,j-1})(\tfrac{3}{2}-\theta^x_{i,j-1})\Big] \\
& - U_{i,j-1}\Big[(\tfrac{1}{2}-\theta^x_{i-1,j-1})(\tfrac{3}{2}+\theta^x_{i-1,j-1}) + (\tfrac{1}{2}+\theta^x_{i,j-1})(\tfrac{3}{2}-\theta^y_{i,j-1})\Big] \\
& + U_{i+1,j}\Big[(\tfrac{1}{2}+\theta^x_{i,j-1})^2\Big] - U_{i+1,j-1}\Big[(\tfrac{1}{2}+\theta^x_{i,j-1})^2\Big] \Big\}.
\end{aligned}
\tag{3.31}
$$

By combining similar terms in (3.30) and (3.31) we can write

$$
\begin{aligned}
\Big(\int\limits_{\gamma^N} - \int\limits_{\gamma^S}\Big) & \nabla u \cdot \vec{n} \, ds \\
\approx \tfrac{1}{2}\frac{\Delta x}{\Delta y} \Big\{ & \tfrac{1}{4}U_{i-1,j+1} + \tfrac{6}{4}U_{i,j+1} + \tfrac{1}{4}U_{i+1,j+1} - \tfrac{2}{4}U_{i-1,j} \\
& - \tfrac{12}{4}U_{i,j} - \tfrac{2}{4}U_{i+1,j} + \tfrac{1}{4}U_{i-1,j-1} + \tfrac{6}{4}U_{i,j-1} + \tfrac{1}{4}U_{i+1,j-1} \\
& - (a_{i-1,j}U_{i-1,j+1} + A_{i-1,j}U_{i,j+1}) + (a_{i-1,j}U_{i-1,j} + A_{i-1,j}U_{i,j}) \\
& - (a_{i,j}U_{i,j} + A_{i,j}U_{i+1,j}) + (a_{i,j}U_{i,j+1} + A_{i,j}U_{i+1,j+1}) \\
& - (a_{i-1,j-1}U_{i-1,j-1} + A_{i-1,j-1}U_{i,j-1}) + (a_{i-1,j-1}U_{i-1,j} + A_{i-1,j-1}U_{i,j}) \\
& - (a_{i,j-1}U_{i,j} + A_{i,j-1}U_{i+1,j}) + (a_{i,j-1}U_{i,j-1} + A_{i,j-1}U_{i+1,j-1}) \Big\} \\
\approx \tfrac{1}{2}\frac{\Delta x}{\Delta y} \Big\{ & \tfrac{1}{4}U_{i-1,j+1} + \tfrac{6}{4}U_{i,j+1} + \tfrac{1}{4}U_{i+1,j+1} - \tfrac{2}{4}U_{i-1,j} \\
& - \tfrac{12}{4}U_{i,j} - \tfrac{2}{4}U_{i+1,j} + \tfrac{1}{4}U_{i-1,j-1} + \tfrac{6}{4}U_{i,j-1} + \tfrac{1}{4}U_{i+1,j-1} \\
& - F^x_{i-1,j-1} + F^x_{i-1,j} - F^x_{i,j} + F^x_{i,j-1} - \tilde{F}^x_{i,j} + \tilde{F}^x_{i,j+1} - \tilde{F}^x_{i-1,j+1} + \tilde{F}^x_{i-1,j} \Big\}.
\end{aligned}
\tag{3.32}
$$

Similarly, we add (3.28) and (3.29) to get

$$
\begin{aligned}
\left(\int_{\gamma^E} - \int_{\gamma^W}\right)\nabla u|_{t_n}\cdot \vec{n}\; ds \approx \tfrac{1}{2}\frac{\Delta y}{\Delta x}\Big\{ & \tfrac{1}{4}U_{i-1,j+1} - \tfrac{2}{4}U_{i,j+1} + \tfrac{1}{4}U_{i+1,j+1} + \tfrac{6}{4}U_{i-1,j} - \tfrac{12}{4}U_{i,j} \\
& + \tfrac{6}{4}U_{i+1,j} + \tfrac{1}{4}U_{i-1,j-1} - \tfrac{2}{4}U_{i,j-1} + \tfrac{1}{4}U_{i+1,j-1} \\
& - (b_{i,j-1}U_{i+1,j-1} + B_{i,j-1}U_{i+1,j}) + (b_{i,j-1}U_{i,j-1} + B_{i,j-1}U_{i,j}) \\
& - (b_{i,j}U_{i,j} + B_{i,j}U_{i,j+1}) + (b_{i,j}U_{i+1,j} + B_{i,j}U_{i+1,j+1}) \\
& - (b_{i-1,j-1}U_{i-1,j-1} + B_{i-1,j-1}U_{i-1,j}) + (b_{i-1,j-1}U_{i,j-1} + B_{i-1,j-1}U_{i,j}) \\
& - (b_{i-1,j}U_{i,j} + B_{i-1,j}U_{i,j+1}) + (b_{i-1,j}U_{i-1,j} + B_{i-1,j}U_{i-1,j+1})\Big\} \\
\approx \tfrac{1}{2}\frac{\Delta y}{\Delta x}\Big\{ & \tfrac{1}{4}U_{i-1,j+1} - \tfrac{2}{4}U_{i,j+1} + \tfrac{1}{4}U_{i+1,j+1} + \tfrac{6}{4}U_{i-1,j} - \tfrac{12}{4}U_{i,j} \\
& + \tfrac{6}{4}U_{i+1,j} + \tfrac{1}{4}U_{i-1,j-1} - \tfrac{2}{4}U_{i,j-1} + \tfrac{1}{4}U_{i+1,j-1} \\
& - F^y_{i-1,j-1} + F^y_{i,j-1} - F^y_{i,j} + F^y_{i-1,j} - \tilde{F}^y_{i+1,j-1} + \tilde{F}^y_{i,j-1} - \tilde{F}^y_{i,j} + \tilde{F}^y_{i+1,j}\Big\}.
\end{aligned}
\tag{3.33}
$$

The source term $g(x, y, t, u, v)$ can be approximated in a standard way.

By setting $\delta_x^2 U^n_{i,j} = U^n_{i-1,j} - 2U^n_{i,j} + U^n_{i+1,j}$ and $\delta_y^2 U^n_{i,j} = U^n_{i,j-1} - 2U^n_{i,j} + U^n_{i,j+1}$, the scheme can be written as

$$
\begin{aligned}
&\tfrac{1}{64}\Big\{U^{n+1}_{i-1,j+1} + 6U^{n+1}_{i-1,j} + U^{n+1}_{i-1,j-1} + 6U^{n+1}_{i,j-1} + 36U^{n+1}_{i,j} + 6U^{n+1}_{i,j+1} + U^{n+1}_{i+1,j+1} \\
&\quad + 6U^{n+1}_{i+1,j} + U^{n+1}_{i+1,j-1}\Big\} - \tfrac{\kappa}{16}\frac{\Delta t}{(\Delta x)^2}\Big\{\delta_x^2 U^{n+1}_{i,j-1} + 6\delta_x^2 U^{n+1}_{i,j} + \delta_x^2 U^{n+1}_{i,j+1}\Big\} \\
&\quad - \tfrac{\kappa}{16}\frac{\Delta t}{(\Delta y)^2}\Big\{\delta_y^2 U^{n+1}_{i-1,j} + 6\delta_y^2 U^{n+1}_{i,j} + \delta_y^2 U^{n+1}_{i+1,j}\Big\} \\
= &\tfrac{1}{64}\Big\{U^{n}_{i-1,j+1} + 6U^{n}_{i-1,j} + U^{n}_{i-1,j-1} + 6U^{n}_{i,j-1} + 36U^{n}_{i,j} + 6U^{n}_{i,j+1} + U^{n}_{i+1,j+1} \\
&\quad + 6U^{n}_{i+1,j} + U^{n}_{i+1,j-1}\Big\} + \tfrac{1}{4}\Big\{H^n_{i,j} - H^n_{i,j-1} + H^n_{i-1,j-1} - H^n_{i-1,j}\Big\} \\
&\quad + \tfrac{1}{16}\Big\{(F^{x,n}_{i,j-1} - F^{x,n}_{i-1,j-1} + \tilde{F}^{x,n}_{i,j+1} - \tilde{F}^{x,n}_{i-1,j+1}) + 3(F^{x,n}_{i,j} - F^{x,n}_{i-1,j} + \tilde{F}^{x,n}_{i,j} - \tilde{F}^{x,n}_{i-1,j}) \\
&\quad + (F^{y,n}_{i-1,j} - F^{y,n}_{i-1,j-1} + \tilde{F}^{y,n}_{i+1,j} - \tilde{F}^{y,n}_{i+1,j-1}) + 3(F^{y,n}_{i,j} - F^{y,n}_{i,j-1} + \tilde{F}^{y,n}_{i,j} - \tilde{F}^{y,n}_{i,j-1})\Big\} \\
&\quad + \tfrac{\kappa}{4}\frac{\Delta t}{(\Delta x)^2}\Big\{\tfrac{1}{4}(\delta_x^2 U^n_{i,j-1} + 6\delta_x^2 U^n_{i,j} + \delta_x^2 U^n_{i,j+1}) \\
&\quad - F^{y,n}_{i-1,j-1} + F^{y,n}_{i,j-1} - F^{y,n}_{i,j} + F^{y,n}_{i-1,j} - \tilde{F}^{y,n}_{i+1,j-1} + \tilde{F}^{y,n}_{i,j-1} - \tilde{F}^{y,n}_{i,j} + \tilde{F}^{y,n}_{i+1,j}\Big\} \\
&\quad + \tfrac{\kappa}{4}\frac{\Delta t}{(\Delta y)^2}\Big\{\tfrac{1}{4}(\delta_y^2 U^n_{i-1,j} + 6\delta_y^2 U^n_{i,j} + \delta_y^2 U^n_{i+1,j}) \\
&\quad - F^{x,n}_{i-1,j-1} + F^{x,n}_{i-1,j} - F^{x,n}_{i,j} + F^{x,n}_{i,j-1} - \tilde{F}^{x,n}_{i,j} + \tilde{F}^{x,n}_{i,j+1} - \tilde{F}^{x,n}_{i-1,j+1} + \tilde{F}^{x,n}_{i-1,j}\Big\} + \Delta t G^n_{i,j},
\end{aligned}
\tag{3.34}
$$

where $G^n_{i,j} = g(U^n_{i,j}, V^n_{i,j}, \bar{x}_i, \bar{y}_j, t_n)$.

As we did in the one dimensional case, we are going to reverse the process and looking for the characteristics in the forward time. Let $(x_{i,j}(t_{n+1}), y_{i,j}(t_{n+1}))$ be the solution of the characteristic equation

$$
\begin{bmatrix} x \\ y \end{bmatrix}' = \begin{bmatrix} v_x(x,y,t) \\ v_y(x,y,t) \end{bmatrix}, \qquad \begin{bmatrix} x \\ y \end{bmatrix}(t_n) = \begin{bmatrix} x_i \\ y_j \end{bmatrix}.
$$

Let $\theta^{x,n+1}_{i,j}$ and $\theta^{y,n+1}_{i,j}$ be the deviations of the solutions ,$x_{i,j}(t_{n+1})$ and $y_{i,j}(t_{n+1})$,of the characteristic equation from the grid point x_i and y_j respectively in the forward time . By the same argument made to solve the system (3.9), we obtain

$$\begin{bmatrix} \Delta x\theta^{x,n+1}_{i,j} \\ \Delta y\theta^{y,n+1}_{i,j} \end{bmatrix} \approx \tfrac{1}{4}\Delta t \begin{bmatrix} [\tilde{V}^n_x]_{i,j} + [\tilde{V}^{n+1}_x]_{i,j} \\ [\tilde{V}^n_y]_{i,j} + [\tilde{V}^{n+1}_y]_{i,j} \end{bmatrix} - \tfrac{1}{8}(\Delta t)^2[\tilde{V}^{n+1}_{xy}]_{i,j} \begin{bmatrix} [\tilde{V}^n_y]_{i,j} + [\tilde{V}^{n+1}_y]_{i,j} \\ [\tilde{V}^n_x]_{i,j} + [\tilde{V}^{n+1}_x]_{i,j} \end{bmatrix}. \quad (3.35)$$

Let

$$\begin{aligned} F^{x,n+1}_{i,j} &= a^{n+1}_{i,j}U^{n+1}_{i,j} + A^{n+1}_{i,j}U^{n+1}_{i+1,j}, & F^{y,n+1}_{i,j} &= b^{n+1}_{i,j}U^{n+1}_{i,j} + B^{n+1}_{i,j}U^{n+1}_{i,j+1}, \\ \tilde{F}^{x,n+1}_{i,j} &= a^{n+1}_{i,j-1}U^{n+1}_{i,j} + A^{n+1}_{i,j-1}U^{n+1}_{i+1,j}, & \tilde{F}^{y,n+1}_{i,j} &= b^{n+1}_{i-1,j}U^{n+1}_{i,j} + B^{n+1}_{i-1,j}U^{n+1}_{i,j+1} \end{aligned}$$

and

$$H^{n+1}_{i,j} = a^{n+1}_{i,j}b^{n+1}_{i,j}U^{n+1}_{i,j} + a^{n+1}_{i,j}B^{n+1}_{i,j}U^{n+1}_{i,j+1} + A^{n+1}_{i,j}b^{n+1}_{i,j}U^{n+1}_{i+1,j} + A^{n+1}_{i,j}B^{n+1}_{i,j}U^{n+1}_{i+1,j+1}.$$

Then, the scheme in the forward time is given by

$$\begin{aligned}
&\tfrac{1}{64}\Big\{U^{n+1}_{i-1,j+1} + 6U^{n+1}_{i-1,j} + U^{n+1}_{i-1,j-1} + 6U^{n+1}_{i,j-1} + 36U^{n+1}_{i,j} + 6U^{n+1}_{i,j+1} + U^{n+1}_{i+1,j+1} \\
&\quad + 6U^{n+1}_{i+1,j} + U^{n+1}_{i+1,j-1}\Big\} + \tfrac{1}{4}\Big\{H^{n+1}_{i,j} - H^{n+1}_{i-1,j} + H^{n+1}_{i,j-1} - H^{n+1}_{i-1,j-1}\Big\} \\
&\quad + \tfrac{1}{16}\Big\{3\big(F^{x,n+1}_{i,j} - F^{x,n+1}_{i-1,j} + \tilde{F}^{x,n+1}_{i,j} - \tilde{F}^{x,n+1}_{i-1,j}\big) + \big(F^{x,n+1}_{i,j-1} - F^{x,n+1}_{i-1,j-1} + \tilde{F}^{x,n+1}_{i,j+1} - \tilde{F}^{x,n+1}_{i-1,j+1}\big) \\
&\quad + 3\big(F^{y,n+1}_{i,j} - F^{y,n+1}_{i,j-1} + \tilde{F}^{y,n+1}_{i,j} - \tilde{F}^{y,n+1}_{i,j-1}\big) + \big(F^{y,n+1}_{i-1,j} - F^{y,n+1}_{i-1,j-1} + \tilde{F}^{y,n+1}_{i+1,j} - \tilde{F}^{y,n+1}_{i+1,j-1}\big)\Big\} \\
&\quad - \tfrac{\kappa}{4}\frac{\Delta t}{(\Delta x)^2}\Big\{\tfrac{1}{4}\big(\delta^2_x U^{n+1}_{i,j-1} + 6\delta^2_x U^{n+1}_{i,j} + \delta^2_x U^{n+1}_{i,j+1}\big) \\
&\quad + F^{y,n+1}_{i,j-1} - F^{y,n+1}_{i,j} + F^{y,n+1}_{i-1,j} - F^{y,n+1}_{i-1,j-1} + \tilde{F}^{y,n+1}_{i,j-1} - \tilde{F}^{y,n+1}_{i+1,j-1} + \tilde{F}^{y,n+1}_{i+1,j} - \tilde{F}^{y,n+1}_{i,j}\Big\} \\
&\quad - \tfrac{\kappa}{4}\frac{\Delta t}{(\Delta y)^2}\Big\{\tfrac{1}{4}\big(\delta^2_y U^{n+1}_{i-1,j} + 6\delta^2_y U^{n+1}_{i,j} + \delta^2_y U^{n+1}_{i+1,j}\big) \\
&\quad + F^{x,n+1}_{i-1,j} - F^{x,n+1}_{i,j} + F^{x,n+1}_{i,j-1} - F^{x,n+1}_{i-1,j-1} + \tilde{F}^{x,n+1}_{i-1,j} - \tilde{F}^{x,n+1}_{i-1,j+1} + \tilde{F}^{x,n+1}_{i,j+1} - \tilde{F}^{x,n+1}_{i,j}\Big\} \\
&= \tfrac{1}{64}\Big\{U^n_{i-1,j+1} + 6U^n_{i-1,j} + U^n_{i-1,j-1} + 6U^n_{i,j-1} + 36U^n_{i,j} + 6U^n_{i,j+1} + U^n_{i+1,j+1} \\
&\quad + 6U^n_{i+1,j} + U^n_{i+1,j-1}\Big\} + \tfrac{\kappa}{16}\frac{\Delta t}{(\Delta x)^2}\Big\{\delta^2_x U^n_{i,j-1} + 6\delta^2_x U^n_{i,j} + \delta^2_x U^n_{i,j+1}\Big\} \\
&\quad + \tfrac{\kappa}{16}\frac{\Delta t}{(\Delta y)^2}\Big\{\delta^2_y U^n_{i-1,j} + 6\delta^2_y U^n_{i,j} + \delta^2_y U^n_{i+1,j}\Big\} + \Delta t G^{n+1}_{i,j}.
\end{aligned} \quad (3.36)$$

For algorithmic purpose we will set

$$
\begin{aligned}
I^n_{i,j} &= U^n_{i-1,j+1} + 6U^n_{i-1,j} + U^n_{i-1,j-1} + 6U^n_{i,j-1} + 36U^n_{i,j} + 6U^n_{i,j+1} + U^n_{i+1,j+1} \\
&\quad + 6U^n_{i+1,j} + U^n_{i+1,j-1}, \\
\omega^{x,n}_{i,j} &= F^{x,n}_{i,j} - F^{x,n}_{i-1,j} + \tilde{F}^{x,n}_{i,j} - \tilde{F}^{x,n}_{i-1,j}, \\
\tilde{\omega}^{x,n}_{i,j} &= F^{x,n}_{i,j-1} - F^{x,n}_{i-1,j-1} + \tilde{F}^{x,n}_{i,j+1} - \tilde{F}^{x,n}_{i-1,j+1}, \\
\omega^{y,n}_{i,j} &= F^{y,n}_{i,j} - F^{y,n}_{i,j-1} + \tilde{F}^{y,n}_{i,j} - \tilde{F}^{y,n}_{i,j-1}, \\
\tilde{\omega}^{y,n}_{i,j} &= F^{y,n}_{i-1,j} - F^{y,n}_{i-1,j-1} + \tilde{F}^{y,n}_{i+1,j} - \tilde{F}^{y,n}_{i+1,j-1}, \\
\Delta^x_{i,j}\{U^n\} &= \frac{\Delta t}{(\Delta x)^2}\left\{\delta^2_x U^n_{i,j-1} + 6\delta^2_x U^n_{i,j} + \delta^2_x U^n_{i,j+1}\right\}, \\
\Delta^y_{i,j}\{U^n\} &= \frac{\Delta t}{(\Delta y)^2}\left\{\delta^2_y U^n_{i-1,j} + 6\delta^2_y U^n_{i,j} + \delta^2_y U^n_{i+1,j}\right\}.
\end{aligned}
$$

By taking the average of the two schemes (3.34)-(3.36) we get

$$
\begin{aligned}
&\tfrac{1}{64} I^{n+1}_{i,j} + \tfrac{1}{32}\big\{3\omega^{x,n+1}_{i,j} + \tilde{\omega}^{x,n+1}_{i,j} + 3\omega^{y,n+1}_{i,j} + \tilde{\omega}^{y,n+1}_{i,j}\big\} - \tfrac{\kappa}{16}\Delta^x_{i,j}\{U^{n+1}\} - \tfrac{\kappa}{16}\Delta^y_{i,j}\{U^{n+1}\} \\
&\quad + \tfrac{1}{8}\left\{H^{n+1}_{i,j} - H^{n+1}_{i-1,j} + H^{n+1}_{i,j-1} - H^{n+1}_{i-1,j-1}\right\} - \tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta y)^2}\big\{-\omega^{x,n+1}_{i,j} + \tilde{\omega}^{x,n+1}_{i,j}\big\} \\
&\quad - \tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta x)^2}\big\{-\omega^{y,n+1}_{i,j} + \tilde{\omega}^{y,n+1}_{i,j}\big\} \\
&= \tfrac{1}{64} I^{n}_{i,j} + \tfrac{1}{32}\big\{3\omega^{x,n}_{i,j} + \tilde{\omega}^{x,n}_{i,j} + 3\omega^{y,n}_{i,j} + \tilde{\omega}^{y,n}_{i,j}\big\} + \tfrac{\kappa}{16}\Delta^x_{i,j}\{U^{n}\} + \tfrac{\kappa}{16}\Delta^y_{i,j}\{U^{n}\} \\
&\quad + \tfrac{1}{8}\left\{H^{n}_{i,j} - H^{n}_{i-1,j} + H^{n}_{i,j-1} - H^{n}_{i-1,j-1}\right\} + \tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta y)^2}\big\{-\omega^{x,n}_{i,j} + \tilde{\omega}^{x,n}_{i,j}\big\} \\
&\quad + \tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta x)^2}\big\{-\omega^{y,n}_{i,j} + \tilde{\omega}^{y,n}_{i,j}\big\} + \tfrac{1}{2}(\Delta t G^n_{i,j} + \Delta t G^{n+1}_{i,j}).
\end{aligned}
\tag{3.37}
$$

3.1.6 Boundary cells

In the above derivation we assumed that $R_{i,j}(t_n)$ is an interior cell. For the boundary cells we use different approximations of the solution $u(x,y,t)$. Let's assume that the solution is approximated by piecewise linear functions in the boundary cells. We assume that there are ghost points $\{x_0, x_{M+1}, y_0, y_{M+1}\}$ outside the domain and set $p_{0,j} = U_{1,j}$,

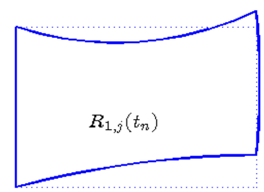

Figure 3.5 The boundary cells for $i = 1$, $R_{1,j}(t_n)$.

$p_{M+1,j} = U_{M,j}$, $q_{i,0} = U_{i,1}$ and $q_{i,M+1} = U_{i,M}$. Hence, we have

$$\begin{aligned}
P_{1,j} &= U_{1,j} + \frac{y - \bar{y}_j}{\Delta y}(U_{1,j+1} - U_{1,j}), \\
P_{M,j} &= U_{M,j} + \frac{y - \bar{y}_j}{\Delta y}(U_{M,j+1} - U_{M,j}), \\
Q_{i,1} &= U_{i,1} + \frac{x - \bar{x}_i}{\Delta x}(U_{i+1,1} - U_{i,1}) \\
Q_{i,M} &= U_{i,M} + \frac{x - \bar{x}_i}{\Delta x}(U_{i+1,M} - U_{i,M}).
\end{aligned}$$

Let's consider the boundary cells on the west edge Γ^W of the domain Ω (Figure 3.5). There will be no change in calculating the integral along the east edge γ^E of the cells $R_{1,j}(t_n)$. For the south edge γ^S, the parametrized equations are given by

$$\begin{aligned}
x_S(\sigma) &= \sigma(1 + \theta^x_{1,j-1})\Delta x \\
y_S(\sigma) &= y_{j-1} + \theta^y_{0,j-1}\Delta y + \sigma(\theta^y_{1,j-1} - \theta^y_{0,j-1})\Delta y, \qquad\qquad \sigma \in [0, 1]
\end{aligned}$$

from which we have

$$
\begin{aligned}
\sigma^* x' &= \bar{x}_1 = \tfrac{1}{2}\Delta x, \\
x' - \sigma^* x' &= (\tfrac{1}{2} + \theta^x_{1,j-1})\Delta x, \\
y_S(0) - \bar{y}_{j-1} &= (\tfrac{1}{2} + \theta^y_{0,j-1})\Delta y, \\
y_S(0) - \bar{y}_j &= (\theta^y_{0,j-1} - \tfrac{1}{2})\Delta y, \\
y_S(1) - \bar{y}_{j-1} &= (\tfrac{1}{2} + \theta^y_{1,j-1})\Delta y, \\
y_S(1) - \bar{y}_j &= (\theta^y_{1,j-1} - \tfrac{1}{2})\Delta y, \\
y_S(\sigma^*) - \bar{y}_j &= (\theta^y_{0,j-1} - \tfrac{1}{2})\Delta y + \sigma^*(\theta^y_{1,j-1} - \theta^y_{0,j-1})\Delta y \\
&= (\theta^y_{1,j-1} - \tfrac{1}{2})\Delta y + (1-\sigma^*)(\theta^y_{0,j-1} - \theta^y_{1,j-1})\Delta y.
\end{aligned}
$$

Along the south edge we have

$$
\int\limits_{\gamma^S} \langle W^x, W^y\rangle \cdot \vec{n} ds = \Big(\int\limits_0^{\sigma^*} + \int\limits_{\sigma^*}^1\Big)\big[W^y x'_S - W^x y'_S\big] d\sigma, \tag{3.38}
$$

where

$$
W^y(x,y) = \begin{cases} U_{1,j-1}(y-\bar{y}_j) + \frac{1}{2\Delta y}[(y-\bar{y}_{j-1})^2 - (\Delta y)^2](U_{1,j} - U_{1,j-1}), & (x,y) \in R^d_{1,j} \\ p_{1,j-1}(x)(y-\bar{y}_j) + \frac{1}{2\Delta y}[(y-\bar{y}_{j-1})^2 - (\Delta y)^2](p_{1,j}(x) - p_{1,j-1}(x)), & \\ & (x,y) \in R^d_{2,j} \end{cases}.
$$

We will be ignoring the second integral. By the trapezoidal rule, the first integral can be written as

$$
\Big(\int\limits_0^{\sigma^*} + \int\limits_{\sigma^*}^1\Big) W^y x'_S d\sigma = \tfrac{1}{2}\Delta x\big[\tfrac{1}{2}W^y_0 + (1 + \theta^x_{1,j-1})W^y_{\sigma^*} + (\tfrac{1}{2} + \theta^x_{1,j-1})W^y_1\big]
$$

where

$$W_0^y = \frac{1}{2}\Delta y\big\{ -U_{1,j-1}\big[(\theta^y_{0,j-1}-\tfrac{1}{2})^2\big] + U_{1,j}\big[(\tfrac{3}{2}+\theta^y_{0,j-1})(\theta^y_{0,j-1}-\tfrac{1}{2})\big]\big\},$$

$$W^y_{\sigma^*} = \Delta y\bigg\{$$

$$U_{1,j-1}\big[(\theta^y_{0,j-1}-\tfrac{1}{2})+\sigma^*(\theta^y_{1,j-1}-\theta^y_{0,j-1})\big]\big[\tfrac{1}{2}(-\theta^y_{0,j-1}+\tfrac{1}{2})+\sigma^*(\theta^y_{1,j-1}-\theta^y_{0,j-1})\big]$$

$$+U_{1,j}\big[(\theta^y_{0,j-1}-\tfrac{1}{2})+\sigma^*(\theta^y_{1,j-1}-\theta^y_{0,j-1})\big]\big[\tfrac{1}{2}(\theta^y_{0,j-1}+\tfrac{3}{2})+\sigma^*(\theta^y_{1,j-1}-\theta^y_{0,j-1})\big]\bigg\},$$

$$= \Delta y\bigg\{$$

$$U_{1,j-1}\big[(\theta^y_{1,j-1}-\tfrac{1}{2})+(1-\sigma^*)(\theta^y_{0,j-1}-\theta^y_{1,j-1})\big]\big[\tfrac{1}{2}(-\theta^y_{1,j-1}+\tfrac{1}{2})+(1-\sigma^*)(\theta^y_{0,j-1}-\theta^y_{1,j-1})\big]$$

$$+U_{1,j}\big[(\theta^y_{1,j-1}-\tfrac{1}{2})+(1-\sigma^*)(\theta^y_{0,j-1}-\theta^y_{1,j-1})\big]\big[\tfrac{1}{2}(\theta^y_{1,j-1}+\tfrac{3}{2})+(1-\sigma^*)(\theta^y_{0,j-1}-\theta^y_{1,j-1})\big]\bigg\},$$

$$W_1^y = \Delta y\bigg\{p_{1,j-1}(x)(\theta^y_{1,j-1}-\tfrac{1}{2})+\tfrac{1}{2}(\theta^y_{1,j-1}-\tfrac{1}{2})(\theta^y_{1,j-1}+\tfrac{3}{2})\big[p_{1,j}(x)-p_{1,j-1}(x)\big]\bigg\}$$

$$= \tfrac{1}{2}\Delta y\bigg\{U_{1,j-1}\big[-(\tfrac{1}{2}-\theta^x_{1,j-1})(\theta^y_{1,j-1}-\tfrac{1}{2})^2\big]+U_{2,j-1}\big[-(\tfrac{1}{2}+\theta^x_{1,j-1})(\theta^y_{1,j-1}-\tfrac{1}{2})^2\big]$$

$$+U_{1,j}\big[(\tfrac{1}{2}-\theta^x_{1,j-1})(\theta^y_{1,j-1}-\tfrac{1}{2})(\theta^y_{1,j-1}+\tfrac{3}{2})\big]$$

$$+U_{2,j}\big[(\tfrac{1}{2}+\theta^x_{1,j})(\theta^y_{1,j-1}-\tfrac{1}{2})(\theta^y_{1,j-1}+\tfrac{3}{2})\big]\bigg\}.$$

By combining the above equations we get (ignoring the higher order terms $(\theta^y_{1,j-1}-\theta^y_{0,j-1})\Delta y$)

$$\int_{\gamma^S} W^y x'_S d\sigma \approx \frac{1}{4}\Delta x\Delta y\bigg\{$$

$$-U_{1,j-1}\Big[(\theta^y_{0,j-1}-\tfrac{1}{2})^2+(\tfrac{1}{2}+\theta^x_{1,j-1})(\theta^y_{1,j-1}-\tfrac{1}{2})^2(\tfrac{3}{2}-\theta^y_{1,j-1})\Big]$$

$$+U_{1,j}\Big[(\tfrac{3}{2}+\theta^y_{0,j-1})(\theta^y_{0,j-1}-\tfrac{1}{2})+(\tfrac{1}{2}+\theta^x_{1,j-1})(\tfrac{3}{2}-\theta^x_{1,j-1})(\theta^y_{1,j-1}-\tfrac{1}{2})(\theta^y_{1,j-1}+\tfrac{3}{2})\Big]$$

$$-U_{2,j-1}\Big[(\tfrac{1}{2}+\theta^x_{1,j-1})^2(\theta^y_{1,j-1}-\tfrac{1}{2})^2\Big]+U_{2,j}\Big[(\tfrac{1}{2}+\theta^x_{1,j-1})^2(\theta^y_{1,j-1}-\tfrac{1}{2})(\theta^y_{1,j-1}+\tfrac{3}{2})\Big]\bigg\}.$$

Similarly we get the integral along the north edge. So

$$\int_{\gamma^N} W^y x'_N d\sigma \approx \frac{1}{4}\Delta x \Delta y \Big\{$$

$$U_{1,j}\Big[(\theta^y_{0,j}+\tfrac{1}{2})(\tfrac{3}{2}-\theta^y_{0,j})+(\tfrac{1}{2}+\theta^x_{1,j})(\tfrac{3}{2}-\theta^x_{1,j})(\tfrac{1}{2}+\theta^y_{1,j})(\tfrac{3}{2}-\theta^y_{1,j})\Big]$$
$$+U_{1,j+1}\Big[(\theta^y_{0,j}+\tfrac{1}{2})^2+(\tfrac{1}{2}+\theta^x_{1,j})(\tfrac{3}{2}-\theta^x_{1,j})(\tfrac{1}{2}+\theta^y_{1,j})^2\Big]$$
$$+U_{2,j}\Big[(\tfrac{1}{2}+\theta^x_{1,j})^2(\tfrac{1}{2}+\theta^y_{1,j})(\tfrac{3}{2}-\theta^y_{1,j})\Big]+U_{2,j+1}\Big[(\tfrac{1}{2}+\theta^x_{1,j})^2(\tfrac{1}{2}+\theta^y_{1,j})^2\Big]\Big\}.$$

Along the west edge γ^W of $R_{1,j}(t_n)$, we consider the function

$$W^x(x,y)=\begin{cases} U_{1,j-1}(x-\bar{x}_1)+\frac{1}{\Delta y}(y-\bar{y}_{j-1})(x-\bar{x}_1)(U_{1,j}-U_{1,j-1}), & (x,y)\in R^d_{1,j} \\ U_{1,j}(x-\bar{x}_1)+\frac{1}{\Delta y}(y-\bar{y}_{j})(x-\bar{x}_1)(U_{1,j+1}-U_{1,j}), & (x,y)\in R^d_{1,j+1} \end{cases}.$$

Then the integral is approximated by

$$\int_0^1 W^x y'_W \approx -\tfrac{1}{4}\Delta x\Delta y\Big\{U_{1,j-1}\Big[(\tfrac{1}{2}-\theta^y_{0,j-1})^2\Big]+U_{1,j+1}\Big[(\tfrac{1}{2}+\theta^y_{0,j})^2\Big]$$
$$+U_{i,j}\Big[(\tfrac{1}{2}-\theta^y_{0,j-1})(\tfrac{3}{2}+\theta^y_{0,j-1})+(\tfrac{1}{2}+\theta^y_{0,j})(\tfrac{3}{2}-\theta^y_{0,j})\Big]\Big\}.$$

We notice that the integrals computed above can be deduced from the general forms of these integrals made on interior cells by applying the idea of the ghost points (i.e. by applying the equality $U_{0,j}=U_{1,j}$). We will apply this idea for the remaining boundaries of the domain Ω. Also, we should take care of the corners. For example we have $U_{1,1}=U_{0,1}=U_{1,0}=U_{0,0}$ on the lower left corner.

3.2 Mass preserving property

In this section we show that in absence of the source terms the scheme (3.37) is mass preserving under the assumption that all values at the grid points are non-negative. It is clear that the locally conservative form of the scheme leads to telescoping sums. Also, since the boundary conditions are homogeneous Neumann conditions we can apply the idea of the ghost points. For example, when $i=1$, we have $U_{0,j}=U_{1,j}$, $1\le j\le M$. The same thing goes for the other boundaries of the domain Ω. Saying differently, if we set $\delta_x U^n_{i,j}=U^n_{i+1,j}-U^n_{i,j}$, and $\delta_y U^n_{i,j}=U^n_{i,j+1}-U^n_{i,j}$, $0\le i,j\le M$, then $\delta_x U^n_{0,j}=\delta_x U^n_{M,j}=\delta_y U^n_{i,0}=\delta_y U^n_{i,M}=0$.

Lemma 3.1: If $\{U^{n+1}_{i,j}\}$ is the solution of the scheme (3.37), given the data $\{U^n_{i,j}\}^M_{i,j=1}$, then

$$\sum_{i,j=1}^{M} U^{n+1}_{i,j}=\sum_{i,j=1}^{M} U^{n}_{i,j}. \tag{3.39}$$

Remark 3.4: *Since* $\dfrac{\partial v}{\partial n}=0$ *on* $\partial\Omega$, *then for* $i,j\in\{0,1,...,M\}$, $\theta^{x,n+1}_{0,j}=\theta^{x,n+1}_{M,j}=\theta^{y,n+1}_{i,0}=\theta^{y,n+1}_{i,M}=0$. *Therefore,* $a^{n+1}_{0,j}=A^{n+1}_{0,j}=a^{n+1}_{M,j}=A^{n+1}_{M,j}=b^{n+1}_{i,0}=B^{n+1}_{i,0}=b^{n+1}_{i,M}=B^{n+1}_{i,M}=0$ *and ,hence,*
$F^{x,n+1}_{0,j}=F^{x,n+1}_{M,j}=\tilde F^{x,n+1}_{0,j}=\tilde F^{x,n+1}_{M,j}=F^{y,n+1}_{i,0}=F^{y,n+1}_{i,M}=\tilde F^{y,n+1}_{i,0}=\tilde F^{y,n+1}_{i,M}=0.$

Proof: In the following we will sum the left hand side of the scheme (3.37) whereas the terms of the right hand side will be summed in the same way. The sum of the left hand side is given by

$$\begin{aligned}
&\sum_{i,j=1}^{M}\Bigg[\tfrac{1}{64}I^{n+1}_{i,j}+\tfrac{1}{32}\big\{3\omega^{x,n+1}_{i,j}+\tilde\omega^{x,n+1}_{i,j}+3\omega^{y,n+1}_{i,j}+\tilde\omega^{y,n+1}_{i,j}\big\}-\tfrac{\kappa}{16}\Delta^x_{i,j}\{U^{n+1}\}-\tfrac{\kappa}{16}\Delta^y_{i,j}\{U^{n+1}\}\\
&\qquad+\tfrac{1}{8}\big\{H^{n+1}_{i,j}-H^{n+1}_{i-1,j}+H^{n+1}_{i,j-1}-H^{n+1}_{i-1,j-1}\big\}-\tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta y)^2}\big\{-\omega^{x,n+1}_{i,j}+\tilde\omega^{x,n+1}_{i,j}\big\}\\
&\qquad-\tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta x)^2}\big\{-\omega^{y,n+1}_{i,j}+\tilde\omega^{y,n+1}_{i,j}\big\}\Bigg]\\
&\qquad=\sum_{i,j=1}^{M}\tfrac{1}{64}I^{n+1}_{i,j}+\tfrac{1}{32}\sum_{i,j=1}^{M}\big\{3\omega^{x,n+1}_{i,j}+\tilde\omega^{x,n+1}_{i,j}+3\omega^{y,n+1}_{i,j}+\tilde\omega^{y,n+1}_{i,j}\big\}\\
&\qquad-\tfrac{\kappa}{16}\sum_{i,j=1}^{M}(\Delta^x_{i,j}\{U^{n+1}\}+\Delta^y_{i,j}\{U^{n+1}\})+\tfrac{1}{8}\sum_{i,j=1}^{M}\big\{H^{n+1}_{i,j}-H^{n+1}_{i-1,j}+H^{n+1}_{i,j-1}-H^{n+1}_{i-1,j-1}\big\}\\
&\qquad-\tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta y)^2}\sum_{i,j=1}^{M}\big\{-\omega^{x,n+1}_{i,j}+\tilde\omega^{x,n+1}_{i,j}\big\}-\tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta x)^2}\sum_{i,j=1}^{M}\big\{-\omega^{y,n+1}_{i,j}+\tilde\omega^{y,n+1}_{i,j}\big\}.
\end{aligned}\tag{3.40}$$

We will sum the flux terms first and leave the mass terms to the end. Therefore, we have

$$\sum_{i,j=1}^{M}\omega_{i,j}^{x,n+1}=\sum_{i,j=1}^{M}(F_{i,j}^{x,n+1}-F_{i-1,j}^{x,n+1}+\tilde{F}_{i,j}^{x,n+1}-\tilde{F}_{i-1,j}^{x,n+1}).$$

It is clear that the above sum is a telescoping sum. So, by summing over i first we get

$$\sum_{i,j=1}^{M}\omega_{i,j}^{x,n+1}=\sum_{j=1}^{M}(F_{M,j}^{x,n+1}-F_{0,j}^{x,n+1}+\tilde{F}_{M,j}^{x,n+1}-\tilde{F}_{0,j}^{x,n+1}).$$

Since $F_{M,j}^{x,n+1}=F_{0,j}^{x,n+1}=\tilde{F}_{M,j}^{x,n+1}=\tilde{F}_{0,j}^{x,n+1}=0$, then

$$\sum_{i,j=1}^{M}\omega_{i,j}^{x,n+1}=0.$$

Similarly, we get that

$$\begin{aligned}\sum_{i,j=1}^{M}\tilde{\omega}_{i,j}^{x,n+1}&=\sum_{i,j=1}^{M}(F_{i,j-1}^{x,n+1}-F_{i-1,j-1}^{x,n+1}+\tilde{F}_{i,j+1}^{x,n+1}-\tilde{F}_{i-1,j+1}^{x,n+1})\\&=\sum_{j=1}^{M}(F_{M,j-1}^{x,n+1}-F_{0,j-1}^{x,n+1}+\tilde{F}_{M,j+1}^{x,n+1}-\tilde{F}_{0,j+1}^{x,n+1})=0.\end{aligned}$$

The sum of the fluxes in the y-direction will be the same. Next we compute the sum of the corner fluxes which is given by

$$\tfrac{1}{8}\sum_{i,j=1}^{M}\left\{H_{i,j}^{n+1}-H_{i-1,j}^{n+1}+H_{i,j-1}^{n+1}-H_{i-1,j-1}^{n+1}\right\}=\tfrac{1}{8}\sum_{j=1}^{M}\left\{H_{M,j}^{n+1}-H_{0,j}^{n+1}+H_{M,j-1}^{n+1}-H_{0,j-1}^{n+1}\right\}.$$

By the same argument, we have $H_{M,j}^{n+1}=H_{0,j}^{n+1}=H_{M,j-1}^{n+1}=H_{0,j-1}^{n+1}=0$. So,

$$\tfrac{1}{8}\sum_{i,j=1}^{M}\left\{H_{i,j}^{n+1}-H_{i-1,j}^{n+1}+H_{i,j-1}^{n+1}-H_{i-1,j-1}^{n+1}\right\}=0.$$

Therefore, we have

$$\begin{aligned}&\tfrac{1}{32}\sum_{i,j=1}^{M}\left\{3\omega_{i,j}^{x,n+1}+\tilde{\omega}_{i,j}^{x,n+1}+3\omega_{i,j}^{y,n+1}+\tilde{\omega}_{i,j}^{y,n+1}\right\}-\tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta y)^2}\sum_{i,j=1}^{M}\left\{-\omega_{i,j}^{x,n+1}+\tilde{\omega}_{i,j}^{x,n+1}\right\}\\&\quad-\tfrac{\kappa}{8}\tfrac{\Delta t}{(\Delta x)^2}\sum_{i,j=1}^{M}\left\{-\omega_{i,j}^{y,n+1}+\tilde{\omega}_{i,j}^{y,n+1}\right\}+\tfrac{1}{8}\sum_{i,j=1}^{M}\left\{H_{i,j}^{n+1}-H_{i-1,j}^{n+1}+H_{i,j-1}^{n+1}-H_{i-1,j-1}^{n+1}\right\}=0.\end{aligned}$$

Hence, the right hand side of (3.40) will be given by

$$\sum_{i,j=1}^{M} \tfrac{1}{64} I_{i,j}^{n+1} - \tfrac{\kappa}{16} \sum_{i,j=1}^{M} (\Delta_{i,j}^{x}\{U^{n+1}\} + \Delta_{i,j}^{y}\{U^{n+1}\})$$
$$= \sum_{i,j=1}^{M} \tfrac{1}{64} I_{i,j}^{n+1} - \tfrac{\kappa}{16} \sum_{i,j=1}^{M} \left(\frac{\Delta t}{(\Delta x)^2} \left\{ \delta_x^2 U_{i,j-1}^{n+1} + 6\delta_x^2 U_{i,j}^{n+1} + \delta_x^2 U_{i,j+1}^{n+1} \right\} \right.$$
$$\left. + \frac{\Delta t}{(\Delta y)^2} \left\{ \delta_y^2 U_{i-1,j}^{n+1} + 6\delta_y^2 U_{i,j}^{n+1} + \delta_y^2 U_{i+1,j}^{n+1} \right\} \right).$$

Computing the first sum in the above brackets gives

$$\begin{aligned}
\sum_{i,j=1}^{M} \delta_x^2 U_{i,j-1}^{n+1} &= \sum_{i,j=1}^{M} (U_{i-1,j-1}^{n+1} - 2U_{i,j-1}^{n+1} + U_{i+1,j-1}^{n+1}) \\
&= \sum_{i,j=1}^{M} \left[(U_{i-1,j-1}^{n+1} - U_{i,j-1}^{n+1}) + (U_{i+1,j-1}^{n+1} - U_{i,j-1}^{n+1}) \right] \\
&= \sum_{i,j=1}^{M} \left[(U_{i+1,j-1}^{n+1} - U_{i,j-1}^{n+1}) - (U_{i,j-1}^{n+1} - U_{i-1,j-1}^{n+1}) \right] \\
&= \sum_{j=1}^{M} \left[(U_{M+1,j-1}^{n+1} - U_{M,j-1}^{n+1}) - (U_{1,j-1}^{n+1} - U_{0,j-1}^{n+1}) \right] \\
&= \sum_{j=1}^{M} \left[\delta_x U_{M,j-1}^{n+1} - \delta_x U_{0,j-1}^{n+1} \right] = 0.
\end{aligned}$$

Therefore, it is easy to see that

$$\sum_{i,j=1}^{M} \left(\frac{\Delta t}{(\Delta x)^2} \left\{ \delta_x^2 U_{i,j-1}^{n+1} + 6\delta_x^2 U_{i,j}^{n+1} + \delta_x^2 U_{i,j+1}^{n+1} \right\} + \frac{\Delta t}{(\Delta y)^2} \left\{ \delta_y^2 U_{i-1,j}^{n+1} + 6\delta_y^2 U_{i,j}^{n+1} + \delta_y^2 U_{i+1,j}^{n+1} \right\} \right)$$
$$= 0.$$

Hence, the remaining sum is

$$\begin{aligned}
\sum_{i,j=1}^{M} \tfrac{1}{64} I_{i,j}^{n+1} = \tfrac{1}{64} \Bigg\{ &\sum_{j=1}^{M} \left[\sum_{i=1}^{M} \left(U_{i-1,j-1}^{n+1} + 6U_{i,j-1}^{n+1} + U_{i+1,j-1}^{n+1} \right) \right] \\
&+ \sum_{j=1}^{M} \left[\sum_{i=1}^{M} \left(6U_{i-1,j}^{n+1} + 36U_{i,j}^{n+1} + 6U_{i+1,j}^{n+1} \right) \right] \\
&+ \sum_{i=1}^{M} \left[\sum_{i=1}^{M} \left(U_{i-1,j+1}^{n+1} + 6U_{i,j+1}^{n} + U_{i+1,j+1}^{n} \right) \right] \Bigg\}.
\end{aligned}$$

We will simplify the first sum in the above equation whereas the argument for the other two sums will be the same. So we have

$$
\begin{aligned}
\sum_{j=1}^{M}\Big[\sum_{i=1}^{M}&\left(U_{i-1,j-1}^{n+1}+6U_{i,j-1}^{n+1}+U_{i+1,j-1}^{n+1}\right)\Big]\\
&=\sum_{j=1}^{M}\left[U_{0,j-1}^{n+1}+\sum_{i=1}^{M-1}U_{i,j-1}^{n+1}+6\sum_{i=1}^{M}U_{i,j-1}^{n+1}+U_{M+1,j-1}^{n+1}+\sum_{i=2}^{M}U_{i,j-1}^{n+1}\right]\\
&=\sum_{j=1}^{M}\left[\left(U_{M+1,j-1}^{n+1}+\sum_{i=1}^{M-1}U_{i,j-1}^{n+1}\right)+6\sum_{i=1}^{M}U_{i-1,j}^{n+1}+\left(U_{0,j-1}^{n+1}+\sum_{i=2}^{M}U_{i,j-1}^{n+1}\right)\right]\\
&=\sum_{j=1}^{M}\left[\left(U_{M,j-1}^{n+1}+\sum_{i=1}^{M-1}U_{i,j-1}^{n+1}\right)+6\sum_{i=1}^{M}U_{i-1,j}^{n+1}+\left(U_{1,j-1}^{n+1}+\sum_{i=2}^{M}U_{i,j-1}^{n+1}\right)\right]\\
&=\sum_{j=1}^{M}\left[\sum_{i=1}^{M}U_{i,j-1}^{n+1}+6\sum_{i=1}^{M}U_{i-1,j}^{n+1}+\sum_{i=1}^{M}U_{i,j-1}^{n+1}\right]\\
&=8\sum_{j=1}^{M}\sum_{i=1}^{M}U_{i,j-1}^{n+1}.
\end{aligned}
$$

Therefore, we get

$$
\begin{aligned}
\sum_{i,j=1}^{M}\tfrac{1}{64}I_{i,j}^{n+1}&=\tfrac{1}{64}\left[8\sum_{j=1}^{M}\sum_{i=1}^{M}U_{i,j-1}^{n+1}+48\sum_{j=1}^{M}\sum_{i=1}^{M}U_{i,j}^{n+1}+8\sum_{j=1}^{M}\sum_{i=1}^{M}U_{i,j+1}^{n+1}\right]\\
&=\tfrac{1}{64}\sum_{i=1}^{M}\left[8\sum_{j=1}^{M}U_{i,j-1}^{n+1}+48\sum_{j=1}^{M}U_{i,j}^{n+1}+8\sum_{j=1}^{M}U_{i,j+1}^{n+1}\right]\\
&=\sum_{i,j=1}^{M}U_{i,j}^{n+1}.
\end{aligned}
$$

Hence, we conclude that

$$\sum_{i,j=1}^{M}U_{i,j}^{n+1}=\sum_{i,j=1}^{M}U_{i,j}^{n}. \qquad \square$$

3.3 Stability

In this section, the stability of the scheme (3.37) will be discussed. We will follow the same argument made in Section 2.5. First, the stability of the unperturbed scheme is discussed, then we move to show the stability of the perturbed scheme (3.37).

3.3.1 Stability of the unperturbed scheme ($\nabla v = 0$)

The constant matrices of the scheme have a special form of construction known as Kronecker product. Therefore, in the following we introduce this product and use some of its properties.

Definition 3.1: If L is an $m \times n$ matrix and U is a $p \times q$ matrix, then the Kronecker product $L \otimes U$ is the $mp \times nq$ block matrix

$$L \otimes U = \begin{bmatrix} l_{11}U & \dots & l_{1n}U \\ \vdots & \ddots & \vdots \\ l_{m1}U & \dots & l_{mn}U \end{bmatrix}.$$

Let

$$A = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} 7 & 1 & & & \\ 1 & 6 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 6 & 1 \\ & & & 1 & 7 \end{bmatrix} \tag{3.41}$$

be $M \times M$ matrices. Assuming that $\Delta x = \Delta y$, the unperturbed scheme is written in the matrix form as

$$[\frac{1}{64}E + \frac{1}{16}r_\kappa(A_x + A_y)]U^{n+1} = [\frac{1}{64}E - \frac{1}{16}r_\kappa(A_x + A_y)]U^n \tag{3.42}$$

where

$$E = B \otimes B = \begin{bmatrix} 7B & B & & & \\ B & 6B & B & & \\ & \ddots & \ddots & \ddots & \\ & & B & 6B & B \\ & & & B & 7B \end{bmatrix}, \qquad A_x = B \otimes A = \begin{bmatrix} 7A & A & & & \\ A & 6A & A & & \\ & \ddots & \ddots & \ddots & \\ & & A & 6A & A \\ & & & A & 7A \end{bmatrix},$$

and

$$A_y = A \otimes B = \begin{bmatrix} -B & B & & & \\ B & -2B & B & & \\ & \ddots & \ddots & \ddots & \\ & & B & -2B & B \\ & & & B & -B \end{bmatrix}$$

are $M^2 \times M^2$ matrices. In order to find the eigenvalues of the above matrices and ,hence, study the stability of the scheme we need the following theorems.

Theorem 3.1: *Let $U \in \mathbb{R}^{m\times n}$, $L \in \mathbb{R}^{p\times q}$ $C \in \mathbb{R}^{n\times s}$ and $D \in \mathbb{R}^{q\times t}$. Then*

$$(U \otimes L)(C \otimes D) = UC \otimes LD \quad (\in \mathbb{R}^{mp\times st})$$

Proof: This is clear from the following calculation

$$\begin{aligned} (U \otimes L)(C \otimes D) &= \begin{bmatrix} u_{11}L & \dots & u_{1n}L \\ \vdots & \ddots & \vdots \\ u_{m1}L & \dots & u_{mn}L \end{bmatrix} \begin{bmatrix} c_{11}D & \dots & c_{1s}D \\ \vdots & \ddots & \vdots \\ c_{n1}D & \dots & c_{ns}D \end{bmatrix} \\ &= \begin{bmatrix} \sum_{k=1}^{n} u_{1k}c_{k1}LD & \dots & \sum_{k=1}^{n} u_{1k}c_{ks}LD \\ \vdots & \ddots & \vdots \\ \sum_{k=1}^{n} u_{mk}c_{k1}LD & \dots & \sum_{k=1}^{n} u_{mk}c_{ks}LD \end{bmatrix} \\ &= UC \otimes LD \quad \square \end{aligned}$$

Theorem 3.2. *Let U, $L \in \mathbb{R}^{M\times M}$ have eigenvalues λ_k, μ_l, for $k, l \in \{0, 1, ..., M-1\}$ respectively. Then the M^2 eigenvalues of $U \otimes L$ are given by*

$$\lambda_1\mu_1, ..., \lambda_1\mu_M, \lambda_2\mu_1, ..., \lambda_2\mu_M, ..., \lambda_M\mu_M.$$

Moreover, if X_k is a right eigenvector of U corresponding to λ_k and Y_l is a right eigenvector of L corresponding to μ_l, then $X_k \otimes Y_l \in \mathbb{R}^{M^2}$ is a right eigenvector of $U \otimes L$ corresponding to $\lambda_k\mu_l$.

Proof: Let X_k be a right eigenvector of U corresponding to λ_k and Y_l a right eigenvector of L corresponding to μ_l ,then

$$\begin{aligned}(U \otimes L)(X_k \otimes Y_l) &= UX_k \otimes LY_l \\ &= \lambda_k X_k \otimes \mu_l Y_l = \lambda_k \mu_l (X_k \otimes Y_l). \qquad \square\end{aligned}$$

Lemma 3.2: The scheme 3.42 is unconditionally stable (i.e. for any r_κ).

Proof:

Recall that the matrices A and B defined in (3.41) (c.f. Appendix A) have the eigenvalues

$$\begin{aligned}\lambda_k &= 2(1-\cos(\tfrac{k\pi}{M})), \\ \mu_k &= 6 + 2\cos(\tfrac{k\pi}{M}),\end{aligned}$$

respectively with associated eigenvectors

$$v^k = \big[\cos\big(\tfrac{(j-\frac{1}{2})k\pi}{M}\big)\big]_{1\le j\le M}, \qquad k = 0, 1, ..., M-1.$$

Therefore, by Theorem 3.2 the eigenvalues of the matrices E, A_x and A_y are given by

$$\begin{aligned}\mu_k\mu_l &= \big[6 + 2\cos(\tfrac{k\pi}{M})\big]\big[6 + 2\cos(\tfrac{l\pi}{M})\big], \\ \mu_k\lambda_l &= 2\big[6 + 2\cos(\tfrac{k\pi}{M})\big]\big[1 - \cos(\tfrac{l\pi}{M})\big], \\ \lambda_k\mu_l &= 2\big[1 - \cos(\tfrac{k\pi}{M})\big]\big[6 + 2\cos(\tfrac{l\pi}{M})\big],\end{aligned}$$

respectively with associated eigenvectors $v^k \otimes v^l, k, l = 0, 1, ...M-1$. Hence, the eigenvalues of the matrix $(\frac{1}{64}E + \frac{1}{16}r_\kappa(A_x + A_y))$ are given by

$$\begin{aligned}\tfrac{1}{64}\mu_k\mu_l + \tfrac{1}{16}r_\kappa(\mu_k\lambda_l + \lambda_k\mu_l) &= \tfrac{1}{64}\Big[36 + 12\big[\cos(\tfrac{k\pi}{M}) + \cos(\tfrac{l\pi}{M})\big] + 4\cos(\tfrac{k\pi}{M})\cos(\tfrac{l\pi}{M})\Big] \\ &+ \tfrac{1}{8}r_\kappa\Big[12 - 4(\cos(\tfrac{k\pi}{M}) + \cos(\tfrac{l\pi}{M}) + \cos(\tfrac{k\pi}{M})\cos(\tfrac{l\pi}{M}))\Big] \\ &= \tfrac{9}{16} + (\tfrac{3}{16} - \tfrac{1}{2}r_\kappa)[\cos(\tfrac{k\pi}{M}) + \cos(\tfrac{l\pi}{M})] \\ &+ (\tfrac{1}{16} - \tfrac{1}{2}r_\kappa)\cos(\tfrac{k\pi}{M})\cos(\tfrac{l\pi}{M}) + \tfrac{3}{2}r_\kappa, \qquad k, l = 0, 1, ..., M-1.\end{aligned}$$

It is not hard to see that these eigenvalues satisfy the inequalities

$$\frac{1}{64}\mu_k\mu_l+\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l) \geq \begin{cases} \frac{9}{16}+2(\frac{3}{16}-\frac{1}{2}r_\kappa)+(\frac{1}{16}-\frac{1}{2}r_\kappa)+\frac{3}{2}r_\kappa, & r_\kappa > \frac{3}{8} \\ \frac{9}{16}-2(\frac{3}{16}-\frac{1}{2}r_\kappa)+(\frac{1}{16}-\frac{1}{2}r_\kappa)+\frac{3}{2}r_\kappa, & \frac{1}{8} < r_\kappa < \frac{3}{8}, \\ \frac{9}{16}-2(\frac{3}{16}-\frac{1}{2}r_\kappa)-(\frac{1}{16}-\frac{1}{2}r_\kappa)+\frac{3}{2}r_\kappa, & r_\kappa < \frac{1}{8} \end{cases}$$

$$= \begin{cases} 1, & r_\kappa > \frac{3}{8} \\ \frac{1}{4}+2r_\kappa & \frac{1}{8} < r_\kappa < \frac{3}{8} \\ \frac{1}{8}+3r_\kappa & r_\kappa < \frac{1}{8} \end{cases}$$

i.e non of these eigenvalues equals zero. Hence the matrix $(\frac{1}{64}E+\frac{1}{16}r_\kappa(A_x+A_y))$ is invertible and the solution U^{n+1} of the scheme (3.42) is well defined.

For convenience of notation we set

$$Q_0=\frac{1}{64}E+\frac{1}{16}r_\kappa(A_x+A_y), \quad Q_1=\frac{1}{64}E-\frac{1}{16}r_\kappa(A_x+A_y).$$

Let $X^{kl}=v^k\otimes v^l$ be an eigenvector of Q_0 and Q_1 corresponding to the eigenvalues $[\frac{1}{64}\mu_k\mu_l\mp\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l)]$, then

$$Q_0^{-1}Q_1X^{kl}=\frac{\frac{1}{64}\mu_k\mu_l-\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l)}{\frac{1}{64}\mu_k\mu_l+\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l)}X^{kl}.$$

Therefore, the eigenvalues of the iteration matrix $Q_0^{-1}Q_1$ are given by

$$\frac{\frac{1}{64}\mu_k\mu_l-\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l)}{\frac{1}{64}\mu_k\mu_l+\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l)}.$$

Since

$$\left|\frac{\frac{1}{64}\mu_k\mu_l-\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l)}{\frac{1}{64}\mu_k\mu_l+\frac{1}{16}r_\kappa(\mu_k\lambda_l+\lambda_k\mu_l)}\right|\leq 1,$$

for any $r_\kappa=\frac{\kappa\Delta t}{(\Delta x)^2}$ then the scheme (3.42) is unconditionally stable. □

3.3.2 Stability of the perturbed scheme ($\nabla v \neq 0$)

Using the same notation defined above, the scheme (3.37) has the following matrix form

$$\begin{aligned} &\big[Q_0+\frac{1}{8}H^{n+1}+\frac{1}{32}(F_x^{n+1}+F_y^{n+1})-\frac{1}{8}r_\kappa(D_x^{n+1}+D_y^{n+1})\big]U^{n+1} \\ &\qquad = \big[Q_1+\frac{1}{8}H^n+\frac{1}{32}(F_x^n+F_y^n)-\frac{1}{8}r_\kappa(D_x^n+D_y^n)\big]U^n, \end{aligned} \tag{3.43}$$

where

$$H^{n+1}=\big[H^{n+1}_{i,j}-H^{n+1}_{i-1,j}+H^{n+1}_{i,j-1}-H^{n+1}_{i-1,j-1}\big],\qquad H^{n}=\big[H^{n}_{i,j}-H^{n}_{i-1,j}+H^{n}_{i,j-1}-H^{n}_{i-1,j-1}\big],$$

$$F^{n+1}_{x}=\big[3\omega^{x,n+1}_{i,j}+\tilde{\omega}^{x,n+1}_{i,j}\big],\qquad F^{n}_{x}=\big[3\omega^{x,n}_{i,j}+\tilde{\omega}^{x,n}_{i,j}\big]$$

$$F^{n+1}_{y}=\big[3\omega^{y,n+1}_{i,j}+\tilde{\omega}^{y,n+1}_{i,j}\big],\qquad F^{n}_{x}=\big[3\omega^{y,n}_{i,j}+\tilde{\omega}^{y,n}_{i,j}\big],$$

$$D^{n+1}_{x}=\big[-\omega^{x,n+1}_{i,j}+\tilde{\omega}^{x,n+1}_{i,j}\big],\qquad D^{n}_{x}=\big[-\omega^{x,n}_{i,j}+\tilde{\omega}^{x,n}_{i,j}\big],$$

$$D^{n+1}_{y}=\big[-\omega^{y,n+1}_{i,j}+\tilde{\omega}^{y,n+1}_{i,j}\big],\qquad D^{n}_{y}=\big[-\omega^{y,n}_{i,j}+\tilde{\omega}^{y,n}_{i,j}\big],$$

are $M^2\times M^2$ matrices.

Lemma 3.3: Assuming Δx and Δy are fixed, then the scheme 3.43 is conditionally stable.

Proof:

Notice that the left hand matrix of the scheme (3.43) can be written as

$$M^{n+1}=Q_0\Big[I+Q_0^{-1}\big[\frac{1}{8}H^{n+1}+\frac{1}{32}(F^{n+1}_{x}+F^{n+1}_{y})-\frac{1}{8}r_\kappa(D^{n+1}_{x}+D^{n+1}_{y})\big]\Big].$$

We set

$$Q^{n+1}_2=Q_0^{-1}\big[\frac{1}{8}H^{n+1}+\frac{1}{32}(F^{n+1}_{x}+F^{n+1}_{y})-\frac{1}{8}r_\kappa(D^{n+1}_{x}+D^{n+1}_{y})\big],$$
$$Q^{n}_2=Q_1^{-1}\big[\frac{1}{8}H^{n}+\frac{1}{32}(F^{n}_{x}+F^{n}_{y})-\frac{1}{8}r_\kappa(D^{n}_{x}+D^{n}_{y})\big].$$

If $\|Q^{n+1}_2\|\leq 1$ as we show in the following, then the matrix $(I+Q^{n+1}_2)$ is invertible and hence M^{n+1} is invertible.

For clarity purpose we rewrite the following definition

$$\begin{bmatrix}\Delta x\Theta^{x,n+1}_{i,j}\\ \Delta y\Theta^{y,n+1}_{i,j}\end{bmatrix}=\tfrac{1}{4}\Delta t\begin{bmatrix}[\tilde{V}^{n}_{x}]_{i,j}+[\tilde{V}^{n+1}_{x}]_{i,j}\\ [\tilde{V}^{n}_{y}]_{i,j}+[\tilde{V}^{n+1}_{y}]_{i,j}\end{bmatrix}-\tfrac{1}{8}(\Delta t)^2[\tilde{V}^{n+1}_{xy}]_{i,j}\begin{bmatrix}[\tilde{V}^{n}_{y}]_{i,j}+[\tilde{V}^{n+1}_{y}]_{i,j}\\ [\tilde{V}^{n}_{x}]_{i,j}+[\tilde{V}^{n+1}_{x}]_{i,j}\end{bmatrix},$$

where

$$[\tilde{V}^{n}_{x}]_{i,j}=\frac{1}{\Delta x}[V^{n}_{i+1,j}-V^{n}_{i,j}+V^{n}_{i+1,j+1}-V^{n}_{i,j+1}],$$
$$[\tilde{V}^{n}_{y}]_{i,j}=\frac{1}{\Delta y}[V^{n}_{i,j+1}-V^{n}_{i,j}+V^{n}_{i+1,j+1}-V^{n}_{i+1,j}],$$
$$[\tilde{V}^{n}_{xy}]_{i,j}=\frac{1}{\Delta x\Delta y}[V^{n}_{i+1,j+1}-V^{n}_{i+1,j}-V^{n}_{i,j+1}+V^{n}_{i,j}].$$

Assuming $[\tilde{V}_x^n]_{i,j}, [\tilde{V}_y^n]_{i,j}$, and $[\tilde{V}_{xy}^n]_{i,j}$ are uniformly bounded and let α, β and γ be their bounds respectively i.e

$$\left|[\tilde{V}_x^n]_{i,j}\right| \leq \alpha, \qquad \left|[\tilde{V}_y^n]_{i,j}\right| \leq \beta, \qquad \left|[\tilde{V}_{xy}^n]_{i,j}\right| \leq \gamma,$$

then, for fixed Δx and Δy we obtain the estimates

$$\begin{aligned}
\left|\Theta_{i,j}^{x,n+1}\right| &\leq \frac{1}{4}\frac{\Delta t}{\Delta x}\left[\left|[\tilde{V}_x^n]_{i,j}\right| + \left|[\tilde{V}_x^{n+1}]_{i,j}\right|\right] + \frac{1}{8}\frac{(\Delta t)^2}{\Delta x}\left|[\tilde{V}_{xy}^{n+1}]_{i,j}\right|\left[\left|[\tilde{V}_y^n]_{i,j}\right| + \left|[\tilde{V}_y^{n+1}]_{i,j}\right|\right] \\
&\leq \frac{1}{2}\frac{\Delta t}{\Delta x}\alpha + \frac{1}{4}\frac{(\Delta t)^2}{\Delta x}\gamma\beta \\
&\leq \left[\frac{1}{2}\frac{1}{\Delta x}\alpha + \frac{1}{4}\frac{\Delta t}{\Delta x}\gamma\beta\right]\Delta t \\
&\leq \left[\frac{1}{2}\frac{1}{\Delta x}\alpha + \frac{1}{4}\frac{\Delta t_0}{\Delta x}\gamma\beta\right]\Delta t \leq K_x\Delta t \leq K_x\Delta t_0
\end{aligned}$$

for some $\Delta t_0 \geq \Delta t$ where

$$K_x = \left[\frac{1}{2}\frac{1}{\Delta x}\alpha + \frac{1}{4}\frac{\Delta t_0}{\Delta x}\gamma\beta + \beta\right].$$

Similarly, we get $|\Theta_{i,j}^{y,n+1}| \leq K_y\Delta t$ for all i, j where

$$K_y = \left[\frac{1}{2}\frac{1}{\Delta x}\beta + \frac{1}{4}\frac{\Delta t_0}{\Delta x}\gamma\alpha\right].$$

Now we can use the above estimates to seek the norms of the left hand matrices of the scheme (3.43). We assume that $\min\{K_x, K_y\}\Delta t_0 \leq \frac{1}{2}$. Notice that $|a_{i,j}| + |A_{i,j}| = 2|\Theta_{i,j}^x|$ and $|b_{i,j}| + |B_{i,j}| = 2|\Theta_{i,j}^y|$. Hence we obtain (we omit the super scripts $n+1$ of the entries)

$$
\begin{aligned}
\|H^{n+1}\|_\infty \leq & |a_{i,j}||b_{i,j}| + |A_{i-1,j}||b_{i-1,j}| + |a_{i,j-1}||B_{i,j-1}| + |A_{i-1,j-1}||B_{i-1,j-1}| + |a_{i-1,j}||b_{i-1,j}| \\
& + |a_{i-1,j-1}||B_{i-1,j-1}| + |A_{i,j}||b_{i,j}| + |A_{i,j-1}||B_{i,j-1}| + |a_{i-1,j-1}||b_{i-1,j-1}| \\
& + |A_{i,j-1}||b_{i,j-1}| + |a_{i,j-1}||b_{i,j-1}| + |A_{i-1,j-1}||b_{i-1,j-1}| + |a_{i-1,j}||B_{i-1,j}| \\
& + |A_{i,j}||B_{i,j}| + |a_{i,j}||B_{i,j}| + |A_{i-1,j}||B_{i-1,j}| \\
& = \big(|a_{i,j}| + |A_{i,j}|\big)\big(|b_{i,j}| + |B_{i,j}|\big) + \big(|a_{i-1,j}| + |A_{i-1,j}|\big)\big(|b_{i-1,j}| + |B_{i-1,j}|\big) \\
& + \big(|a_{i,j-1}| + |A_{i,j-1}|\big)\big(|b_{i,j-1}| + |B_{i,j-1}|\big) \\
& + \big(|a_{i-1,j-1}| + |A_{i-1,j-1}|\big)\big(|b_{i-1,j-1}| + |B_{i-1,j-1}|\big) \\
& = (2|\Theta^x_{i,j}|)(2|\Theta^y_{i,j}|) + (2|\Theta^x_{i-1,j}|)(2|\Theta^y_{i-1,j}|) + (2|\Theta^x_{i,j-1}|)(2|\Theta^y_{i,j-1}|) \\
& + (2|\Theta^x_{i-1,j-1}|)(2|\Theta^y_{i-1,j-1}|) \\
& \leq 16K_xK_y(\Delta t)^2
\end{aligned}
$$

Also, the 1-norm $\|H^{n+1}\|_1$ is bounded above by

$$
\begin{aligned}
\|H^{n+1}\|_1 & \leq 4\big[|a_{i,j}||b_{i,j}| + |a_{i,j-1}||B_{i,j-1}| + |A_{i-1,j}||b_{i-1,j}| + |A_{i-1,j-1}||B_{i-1,j-1}\big] \\
& \leq 16(K_x^{n+1}\Delta t + (K_x^{n+1}\Delta t)^2)(K_y^{n+1}\Delta t + (K_y^{n+1}\Delta t)^2) \\
& \leq 16K_x(1 + K_x\Delta t_0)K_y(1 + K_y\Delta t_0)\Delta t^2.
\end{aligned}
$$

For simplicity, we set

$$
\Gamma_x(K_x) = K_x(1 + K_x\Delta t_0) \qquad \text{and} \qquad \Gamma_y(K_y) = K_y(1 + K_y\Delta t_0).
$$

Therefore,

$$
\|H^{n+1}\|_2 \leq \sqrt{\|H^{n+1}\|_1\|H^{n+1}\|_\infty} \leq 16(\Delta t)^2\sqrt{K_xK_y\Gamma_x(K_x)\Gamma_y(K_y)}.
$$

Next we estimate the norm of the matrix $\mathit{\Gamma}_x^{n+1}$. The upper bound of the sup-norm is given by

$$\begin{aligned}
\|\mathcal{F}_x^{n+1}\|_\infty \leq & 3\big[(|a_{i,j}| + |A_{i,j}|) + (|a_{i-1,j}| + |A_{i-1,j}|) + (|a_{i,j-1}| + |A_{i,j-1}|) \\
& + (|a_{i-1,j-1}| + |A_{i-1,j-1}|)\big] \\
& + (|a_{i,j}| + |A_{i,j}|) + (|a_{i-1,j}| + |A_{i-1,j}|) + (|a_{i,j-1}| + |A_{i,j-1}|) \\
& + (|a_{i-1,j-1}| + |A_{i-1,j-1}|) \\
& = 4\big[(|a_{i,j}| + |A_{i,j}|) + (|a_{i-1,j}| + |A_{i-1,j}|) + (|a_{i,j-1}| + |A_{i,j-1}|) \\
& + (|a_{i-1,j-1}| + |A_{i-1,j-1}|)\big] \\
& = 8\big[|\Theta^x_{i,j}| + |\Theta^x_{i-1,j}| + |\Theta^x_{i,j-1}| + |\Theta^x_{i-1,j-1}|\big] \\
& \leq 32K_x\Delta t.
\end{aligned}$$

However, the 1-norm $\|\mathcal{F}_x^{n+1}\|_1$ is bounded above by

$$\|\mathcal{F}_x^{n+1}\|_1 \leq 8\big[|a_{i,j}| + |a_{i,j-1}| + |A_{i-1,j}| + |A_{i-1,j-1}|\big] \leq 32(K_x\Delta t + (K_x\Delta t)^2).$$

Then

$$\|\mathcal{F}_x^{n+1}\|_2 \leq 32\Delta t\sqrt{K_x\Gamma_x(K_x)}.$$

The norm of D_x^{n+1} will be given as follows

$$\begin{aligned}
\|D_x^{n+1}\|_\infty \leq & 2\big[|a_{i,j}| + |A_{i,j}| + |a_{i,j-1}| + |A_{i,j-1}| + |a_{i-1,j}| + |A_{i-1,j}| + |a_{i-1,j-1}| + |A_{i-1,j-1}|\big] \\
= & 4\big[|\Theta^x_{i,j}| + |\Theta^x_{i,j-1}| + |\Theta^x_{i-1,j}| + |\Theta^x_{i-1,j-1}|\big] \\
& \leq 16K_x\Delta t
\end{aligned}$$

Similarly, we have

$$\|D_x^{n+1}\|_1 \leq 4\big[|a_{i,j}| + |a_{i,j-1}| + |A_{i-1,j}| + |A_{i-1,j-1}|\big] \leq 16(K_x\Delta t + (K_x\Delta t)^2).$$

Therefore,

$$\|D_x^{n+1}\|_2 \leq 16\Delta t\sqrt{K_x\Gamma_x(K_x)}.$$

By the same argument, we get

$$\|F_y^{n+1}\|_2 \leq 32\Delta t\sqrt{K_y\Gamma_y(K_y)},$$
$$\|D_y^{n+1}\|_2 \leq 16\Delta t\sqrt{K_y\Gamma_y(K_y)}.$$

It is clear that $\|Q_0^{-1}\|_2 = [\min\{1, \frac{1}{4}+2r_\kappa, \frac{1}{8}+3r_\kappa\}]^{-1}$. To simplify the notation, we set

$$\varphi_x = \sqrt{K_x\Gamma_x(K_x)},$$
$$\varphi_y = \sqrt{K_y\Gamma_y(K_y)}.$$

Combining all of the above, we get that

$$\begin{aligned}
\|Q_2^{n+1}\|_2 &\leq \|Q_0^{-1}\|_2\left[\frac{1}{32}\big[\|F_x^{n+1}\|_2 + \|F_y^{n+1}\|_2\big] + \frac{1}{8}\|H^{n+1}\|_2 + \frac{1}{8}r_\kappa\big[\|D_x^{n+1}\|_2 + \|D_y^{n+1}\|_2\big]\right] \\
&\leq \frac{\Delta t(\varphi_x+\varphi_y) + 2(\Delta t)^2\varphi_x\varphi_y + 2r_\kappa\Delta t(\varphi_x+\varphi_y)}{\min\{1, \frac{1}{4}+2r_\kappa, \frac{1}{8}+3r_\kappa\}} \\
&\leq \frac{\Delta t\big[\varphi_x+\varphi_y+2\Delta t_0\varphi_x\varphi_y+2r_\kappa(\varphi_x+\varphi_y)\big]}{\min\{1, \frac{1}{4}+2r_\kappa, \frac{1}{8}+3r_\kappa\}}.
\end{aligned} \tag{3.44}$$

Bounding the right hand side of the above inequality by 1 gives that

$$\Delta t < \frac{\min\{1, \frac{1}{4}+2r_\kappa, \frac{1}{8}+3r_\kappa\}}{\big[\varphi_x+\varphi_y+2\Delta t_0\varphi_x\varphi_y+2r_\kappa(\varphi_x+\varphi_y)\big]}.$$

Under this restriction we get $\|Q_2^{n+1}\|_2 \leq \tilde{C}\Delta t < 1$ where

$$\tilde{C} = \frac{\big[\varphi_x+\varphi_y+2\Delta t_0\varphi_x\varphi_y+2r_\kappa(\varphi_x+\varphi_y)\big]}{\min\{1, \frac{1}{4}+2r_\kappa, \frac{1}{8}+3r_\kappa\}}.$$

Hence, the Neumann series

$$(I+Q_2^{n+1})^{-1} = I - Q_2^{n+1} + (-Q_2^{n+1})^2 \ldots = \sum_{k=0}^{\infty}(-Q_2^{n+1})^k$$

is convergent. Therefore,

$$\|(I+Q_2^{n+1})^{-1}\|_2 \leq \frac{1}{1-\|Q_2^{n+1}\|_2} = 1 + \frac{\|Q_2^{n+1}\|_2}{1-\|Q_2^{n+1}\|_2} \leq 1 + \frac{\tilde{C}\Delta t}{1-\tilde{C}\Delta t_0} = 1 + C_1\Delta t$$

where $C_1 = \frac{\tilde{C}}{1-\tilde{C}\Delta t_0}$. By the same argument we get the estimate of the norm of Q_2^n, i.e $\|Q_2^n\|_2 \leq \tilde{C}\Delta t$.

Notice that the iteration matrix of the scheme (3.43) is given by

$$\begin{aligned} M^n &= (I + Q_2^{n+1})^{-1} Q_0^{-1} \big[Q_1 + \frac{1}{8} H^n + \frac{1}{32} (F_x^n + F_y^n) - \frac{1}{8} r_\kappa (D_x^n + D_y^n) \big] \\ &= (I + Q_2^{n+1})^{-1} [Q_0^{-1} Q_1 + Q_2^n]. \end{aligned}$$

So, we have

$$\begin{aligned} \|M^n\|_2 &\le \|(I + Q_2^{n+1})^{-1}\|_2 \big[\|Q_0^{-1} Q_1\|_2 + \|Q_2^n\|_2 \big] \\ &\le (1 + C_1 \Delta t)(1 + \tilde{C} \Delta t) \le 1 + (C_1 + \tilde{C}) \Delta t + C_1 \tilde{C}_2 (\Delta t)^2 \\ &\le 1 + (C_1 + \tilde{C} + C_1 \tilde{C} \Delta t_0) \Delta t = 1 + C_2 \Delta t \end{aligned}$$

where $C_2 = C_1 + \tilde{C} + C_1 \tilde{C}_2 \Delta t_0$. Then, for T and N to be the final time and total time steps (i.e. $\Delta t = \frac{T}{N}$) respectively, we get by iteration

$$\|U^{n+1}\|_2 \le (1 + C_2 \Delta t)^n \|U_0\|_2 \le (1 + C_1 \tfrac{T}{N})^N \|U_0\|_2 \le e^{C_2 T} \|U_0\|_2.$$

Therefore, the scheme (3.43) is conditionally stable. $\square$

3.4 Consistency

In this section we are going to discuss the consistency of the scheme. Let $u(x, y, t)$ be a smooth solution of (3.1) and let $u_{i,j}^n = u(\bar{x}_i, \bar{y}_j, t_n)$ denotes its values at the space-time nodal points $(\bar{x}_i, \bar{y}_j, t_n)$. The local truncation error $\tau_{i,j}^{n+1}$ is defined by

$$\begin{aligned} \tau_{i,j}^{n+1} = \frac{1}{\Delta t} \Big\{ & \tfrac{1}{64} \big[I_{i,j}^{n+1} - I_{i,j}^n \big] - \tfrac{\kappa}{16} \big[\Delta_{i,j}^x \{u^{n+1}\} + \Delta_{i,j}^y \{u^{n+1}\} + \Delta_{i,j}^x \{u^n\} + \Delta_{i,j}^y \{u^n\} \big] \\ & \tfrac{1}{32} \big[3\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1} + 3\omega_{i,j}^{y,n+1} + \tilde{\omega}_{i,j}^{y,n+1} - 3\omega_{i,j}^{x,n} - \tilde{\omega}_{i,j}^{x,n} - 3\omega_{i,j}^{y,n} - \tilde{\omega}_{i,j}^{y,n} \big] \\ & + \tfrac{1}{8} \big[H_{i,j}^{n+1} - H_{i-1,j}^{n+1} + H_{i,j-1}^{n+1} - H_{i-1,j-1}^{n+1} - H_{i,j}^n + H_{i-1,j}^n - H_{i,j-1}^n + H_{i-1,j-1}^n \big] \\ & - \tfrac{\kappa}{8} \tfrac{\Delta t}{(\Delta y)^2} \big[-\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1} + \omega_{i,j}^{x,n} - \tilde{\omega}_{i,j}^{x,n} \big] \\ & - \tfrac{\kappa}{8} \tfrac{\Delta t}{(\Delta x)^2} \big[-\omega_{i,j}^{y,n+1} + \tilde{\omega}_{i,j}^{y,n+1} + \omega_{i,j}^{y,n} - \tilde{\omega}_{i,j}^{y,n} \big] \Big\} \end{aligned} \tag{3.45}$$

Lemma 3.4: The scheme 3.37 is of second order in both time and space.

Remark 3.5 *All the functions and derivatives in the following expansions are evaluated at the point* $(\bar{x}_i, \bar{y}_j, t_n)$.

Proof:

We are writing $\Delta t = \nu h$ and $\Delta x = \Delta y = h$. By expanding about the point $(\bar{x}_i, \bar{y}_j, t_n)$ we get

$$\begin{aligned} I_{i,j}^n &= (u_{i-1,j+1}^n + 6u_{i,j+1}^n + u_{i-1,j+1}^n) + 6(u_{i-1,j}^n + 6u_{i,j}^n + u_{i-1,j}^n) \\ &+ (u_{i-1,j-1}^n + 6u_{i,j-1}^n + u_{i-1,j-1}^n) \\ &= 64u^n + 8h^2(u_{xx}^n + u_{yy}^n) + O(h^4). \end{aligned} \tag{3.46}$$

Also we notice that $I_{i,j}^{n+1} = I_{i,j}^n + (\nu h)(I_{i,j}^n)_t + \frac{1}{2}(\nu h)^2(I_{i,j}^n)_{tt} + \frac{1}{6}(\nu h)^3(I_{i,j}^n)_{ttt} + O((\nu h)^4)$. Hence

$$\tfrac{1}{64(\nu h)}(I_{i,j}^{n+1} - I_{i,j}^n) = u_t^n + \tfrac{1}{2}(\nu h)u_{tt}^n + \tfrac{1}{8}h^2(u_{xxt}^n + u_{yyt}^n) + \tfrac{1}{6}(\nu h)^2 u_{ttt}^n + O(h^3) + O((\nu h)^3). \tag{3.47}$$

Similarly, we have

$$\begin{aligned} \tfrac{1}{16(\nu h)}(\Delta_{i,j}^x\{u^n\} + \Delta_{i,j}^y\{u^n\}) = \tfrac{1}{16h^2}\Big[&8h^2(u_{xx}^n + u_{yy}^n) + h^4\big(\tfrac{8}{12}(u_{xxxx}^n + u_{yyyy}^n) + 2u_{xxyy}^n\big) \\ &+ O(h^5)\Big], \end{aligned} \tag{3.48}$$

and

$$\begin{aligned} \tfrac{1}{16(\nu h)}(\Delta_{i,j}^x\{u^{n+1}\} + \Delta_{i,j}^y\{u^{n+1}\}) = \tfrac{1}{16h^2}\Big[&8h^2(u_{xx}^n + u_{yy}^n) + h^4\big(\tfrac{8}{12}(u_{xxxx}^n + u_{yyyy}^n) + 2u_{xxyy}^n\big) \\ &+ 8(\nu h)h^2(u_{xx}^n + u_{yy}^n) + 4h^2(\nu h)^2(u_{xxtt}^n + u_{yytt}^n) \\ &+ \tfrac{8}{6}h^2(\nu h)^3(u_{xxttt}^n + u_{yyttt}^n) + O(h^6)\Big]. \end{aligned} \tag{3.49}$$

In the following we are writing the Taylor expansion of the flux terms. For convenience we set

$$C_1^n = -a_{i-1,j}^n + a_{i,j}^n - A_{i-1,j}^n + A_{i,j}^n, \qquad \tilde{C}_1^n = -a_{i-1,j-1}^n + a_{i,j-1}^n - A_{i-1,j-1}^n + A_{i,j-1}^n,$$
$$C_2^n = a_{i-1,j}^n + A_{i,j}^n, \qquad \tilde{C}_2^n = a_{i-1,j-1}^n + A_{i,j-1}^n,$$
$$C_3^n = -a_{i-1,j}^n + A_{i,j}^n, \qquad \tilde{C}_3^n = -a_{i-1,j-1}^n + A_{i,j-1}^n.$$

Then

$$\begin{aligned} F_{i,j}^{x,n} - F_{i-1,j}^{x,n} &= -a_{i-1,j}^n u_{i-1,j}^n + (a_{i,j}^n - A_{i-1,j}^n) u_{i,j}^n + A_{i,j}^n u_{i+1,j}^n \\ &= C_1^n u^n + C_2^n h u_x^n + \tfrac{1}{2} C_3^n h^2 u_{xx}^n + \tfrac{1}{6} C_2^n h^3 u_{xxx} + O(h^4). \end{aligned} \tag{3.50}$$

In the next expansion we used the previous one considered as a function at the point $(\bar{x}_i, \bar{y}_{j-1}, t_n)$ and then expand it about the point $(\bar{x}_i, \bar{y}_j, t_n)$.

$$\begin{aligned} F_{i,j-1}^{x,n} - F_{i-1,j-1}^{x,n} &= -a_{i-1,j-1}^n u_{i-1,j-1}^n + (a_{i,j-1}^n - A_{i-1,j-1}^n) u_{i,j-1}^n + A_{i,j-1}^n u_{i+1,j-1}^n \\ &= \tilde{C}_1^n u^n + h[\tilde{C}_2^n u_x^n - \tilde{C}_1^n u_y^n] + \tfrac{1}{2} h^2 [\tilde{C}_3^n u_{xx}^n - 2\tilde{C}_2^n u_{xy}^n + \tilde{C}_1^n u_{yy}^n] \\ &+ \tfrac{1}{6} h^3 [\tilde{C}_2^n u_{xxx}^n - 3\tilde{C}_3^n u_{xxy}^n + 3\tilde{C}_2^n u_{xyy}^n - \tilde{C}_1^n u_{yyy}^n] + O(h^4). \end{aligned} \tag{3.51}$$

Similarly, we get

$$\begin{aligned} \tilde{F}_{i,j}^{x,n} - \tilde{F}_{i-1,j}^{x,n} &= -a_{i-1,j-1}^n u_{i-1,j}^n + (a_{i,j-1}^n - A_{i-1,j-1}^n) u_{i,j}^n + A_{i,j-1}^n u_{i+1,j}^n \\ &= \tilde{C}_1^n u^n + h \tilde{C}_2^n u_x^n + \tfrac{1}{2} h^2 \tilde{C}_3^n u_{xx}^n + \tfrac{1}{6} h^3 \tilde{C}_2^n u_{xxx}^n + O(h^4), \end{aligned} \tag{3.52}$$

and

$$\begin{aligned} &\tilde{F}_{i,j+1}^{x,n} - \tilde{F}_{i-1,j+1}^{x,n} = -a_{i-1,j}^n u_{i-1,j+1}^n + (a_{i,j}^n - A_{i-1,j}^n) u_{i,j+1}^n + A_{i,j}^n u_{i+1,j+1}^n \\ &= C_1^n u^n + h(C_2^n u_x^n + C_1^n u_y^n) + \tfrac{1}{2} h^2 \left[C_3^n u_{xx}^n + 2C_2^n u_{xy}^n \right] + \tfrac{1}{6} h^3 \left[C_2^n u_{xxx}^n + 3C_3^n u_{xxy}^n \right] + O(h^4). \end{aligned} \tag{3.53}$$

At time t_{n+1} we have the following expansions

$$\begin{aligned}
F^{x,n+1}_{i,j} - F^{x,n+1}_{i-1,j} &= -a^{n+1}_{i-1,j}u^{n+1}_{i-1,j} + (a^{n+1}_{i,j} - A^{n+1}_{i-1,j})u^{n+1}_{i,j} + A^{n+1}_{i,j}u^{n+1}_{i+1,j} \\
&= C^{n+1}_1 u^n + h\big[C^{n+1}_2 u^n_x + \nu C^{n+1}_1 u^n_t\big] \\
&+ \tfrac{1}{2}h^2\big[C^{n+1}_3 u^n_{xx} + \nu^2 C^{n+1}_1 u^n_{tt} + 2\nu C^{n+1}_2 u^n_{xt}\big] \\
&+ \tfrac{1}{2}h^3\big[\tfrac{1}{3}C^{n+1}_2 u^n_{xxx} + \nu C^{n+1}_3 u^n_{xxt} + \nu^2 C^{n+1}_2 u^n_{xtt} + \tfrac{1}{3}\nu^3 C^{n+1}_1 u^n_{ttt}\big] + O(h^4),
\end{aligned} \tag{3.54}$$

$$\begin{aligned}
F^{x,n+1}_{i,j-1} - F^{x,n+1}_{i-1,j-1} &= -a^{n+1}_{i-1,j-1}u^{n+1}_{i-1,j-1} + (a^{n+1}_{i,j-1} - A^{n+1}_{i-1,j-1})u^{n+1}_{i,j-1} + A^{n+1}_{i,j-1}u^{n+1}_{i+1,j-1} \\
&= \tilde{C}^{n+1}_1 u^n + h[\tilde{C}^{n+1}_2 u^n_x - \tilde{C}^{n+1}_1(u^n_y - \nu u^n_t)] \\
&+ h^2\Big[\nu[\tilde{C}^{n+1}_2 u^n_{xt} - \tilde{C}^{n+1}_1(u^n_{yt} - \tfrac{1}{2}\nu u^n_{tt})] + \tfrac{1}{2}[\tilde{C}^{n+1}_3 u^n_{xx} - 2\tilde{C}^{n+1}_2 u^n_{xy} + \tilde{C}^{n+1}_1 u^n_{yy}]\Big] \\
&+ \tfrac{1}{2}h^3\Big[\nu[\tilde{C}^{n+1}_3 u^n_{xxt} - 2\tilde{C}^{n+1}_2 u^n_{xyt} + \tilde{C}^{n+1}_1 u^n_{yyt}] + \nu^2[\tilde{C}^{n+1}_2 u^n_{xtt} - \tilde{C}^{n+1}_1 u^n_{ytt}] \\
&+ \tfrac{1}{3}[\tilde{C}^{n+1}_2 u^n_{xxx} - 3\tilde{C}^{n+1}_3 u^n_{xxy} + 3\tilde{C}^{n+1}_2 u^n_{xyy} - \tilde{C}^{n+1}_1 u_{yyy} + \nu^3\tilde{C}^{n+1}_1 u^n_{ttt}]\Big] \\
&+ O(h^4),
\end{aligned} \tag{3.55}$$

$$\begin{aligned}
\tilde{F}^{x,n+1}_{i,j} - \tilde{F}^{x,n+1}_{i-1,j} &= -a^{n+1}_{i-1,j-1}u^{n+1}_{i-1,j} + (a^{n+1}_{i,j-1} - A^{n+1}_{i-1,j-1})u^{n+1}_{i,j} + A^{n+1}_{i,j-1}u^{n+1}_{i+1,j} \\
&= \tilde{C}^{n+1}_1 u^n + h\Big[\tilde{C}^{n+1}_2 u^n_x + \nu\tilde{C}^{n+1}_1 u^n_t\Big] \\
&+ \tfrac{1}{2}h^2\Big[\tilde{C}^{n+1}_3 u^n_{xx} + \nu^2\tilde{C}^{n+1}_1 u^n_{tt} + 2\nu\tilde{C}^{n+1}_2 u^n_{xt}\Big] \\
&+ \tfrac{1}{2}h^3\Big[\nu\tilde{C}^{n+1}_3 u^n_{xxt} + \nu\tilde{C}^{n+1}_2 u^n_{xtt} + \tfrac{1}{3}\nu^3\tilde{C}_1 u^n_{ttt} + \tfrac{1}{3}\tilde{C}^{n+1}_2 u^n_{xxx}\Big] + O(h^4),
\end{aligned} \tag{3.56}$$

and

$$\begin{aligned}
\tilde{F}^{x,n+1}_{i,j+1} - \tilde{F}^{x,n+1}_{i-1,j+1} &= -a^{n+1}_{i-1,j}u^{n+1}_{i-1,j+1} + (a^{n+1}_{i,j} - A^{n+1}_{i-1,j})u^{n+1}_{i,j+1} + A^{n+1}_{i,j}u^{n+1}_{i+1,j+1} \\
&= C^{n+1}_1 u^n + h\Big[C^{n+1}_2 u^n_x + \nu C^{n+1}_1 u^n_t\Big] \\
&+ \tfrac{1}{2}h^2\Big[C^{n+1}_3 u^n_{xx} + 2C^{n+1}_2 u^n_{xy} + \nu C^{n+1}_1 u^n_{tt} + 2\nu C^{n+1}_2 u^n_{xt}\Big] \\
&+ \tfrac{1}{2}h^3\Big[\nu(C^{n+1}_3 u^n_{xxt} + 2C^{n+1}_2 u^n_{xyt}) + \nu^2 C^{n+1}_2 u^n_{xtt} + \tfrac{1}{3}\nu^3 C_1 u^n_{ttt} \\
&+ \tfrac{1}{3}(C^{n+1}_2 u^n_{xxx} + 3C^{n+1}_2 u^n_{xyy} + 3C^{n+1}_2 u^n_{yyy})\Big] + O(h^4).
\end{aligned} \tag{3.57}$$

Then, the addition of (3.50) and (3.52) gives

$$\begin{aligned}
\omega^{x,n}_{i,j} = F^{x,n}_{i,j} - F^{x,n}_{i-1,j} + \tilde{F}^{x,n}_{i,j} - \tilde{F}^{x,n}_{i-1,j} = (C^n_1 + \tilde{C}^n_1)u^n + h(C^n_2 + \tilde{C}^n_2)u^n_x \\
+ \tfrac{1}{2}h^2(C^n_3 + \tilde{C}^n_3)u^n_{xx} + \tfrac{1}{6}h^3(C^n_2 + \tilde{C}^n_2)u^n_{xxx} + O(h^4).
\end{aligned} \tag{3.58}$$

Similarly, adding (3.51) and (3.53) leads to

$$\begin{aligned}
\tilde{\omega}^{x,n}_{i,j} &= F^{x,n}_{i,j-1} - F^{x,n}_{i-1,j-1} + \tilde{F}^{x,n}_{i,j+1} - \tilde{F}^{x,n}_{i-1,j+1} \\
&= (C^n_1 + \tilde{C}^n_1)u^n + h\Big[(C^n_2 + \tilde{C}^n_2)u^n_x + (C^n_1 - \tilde{C}^n_1)u^n_y\Big] \\
&+ \tfrac{1}{2}h^2\Big[(C^n_3 + \tilde{C}^n_3)u^n_{xx} + 2(C^n_2 - \tilde{C}^n_2)u^n_{xy} + \tilde{C}^n_1 u^n_{yy}\Big] \\
&+ \tfrac{1}{6}h^3\Big[(C^n_2 + \tilde{C}^n_2)u^n_{xxx} + 3(C^n_3 - \tilde{C}^n_3)u^n_{xxy} + 3\tilde{C}^n_2 u^n_{xyy} - \tilde{C}^n_1 u^n_{yyy}\Big] + (h^4).
\end{aligned} \tag{3.59}$$

From the equations (3.58) and (3.59) we get

$$\begin{aligned}
3\omega^{x,n}_{i,j} + \tilde{\omega}^{x,n}_{i,j} = 4(C^n_1 + \tilde{C}^n_1)u^n + h\Big[4(C^n_2 + \tilde{C}^n_2)u^n_x + (C^n_1 - \tilde{C}^n_1)u^n_y\Big] \\
+ \tfrac{1}{2}h^2\Big[4(C^n_3 + \tilde{C}^n_3)u^n_{xx} + 2(C^n_2 - \tilde{C}^n_2)u^n_{xy} + \tilde{C}^n_1 u^n_{yy}\Big] \\
+ \tfrac{1}{6}h^3\Big[4(C^n_2 + \tilde{C}^n_2)u^n_{xxx} + 3(C^n_3 - \tilde{C}^n_3)u^n_{xxy} + 3\tilde{C}^n_2 u^n_{xyy} - \tilde{C}^n_1 u^n_{yyy}\Big] + O(h^4).
\end{aligned} \tag{3.60}$$

Similarly, we have

$$
\begin{aligned}
&3\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1} \\
&\quad = 4(C_1^{n+1} + \tilde{C}_1^{n+1})u^n \\
&\quad +h\Big[4(C_2^{n+1} + \tilde{C}_2^{n+1})u_x^n + (C_1^{n+1} - \tilde{C}_1^{n+1})u_y^n + 4\nu(C_1^{n+1} + \tilde{C}_1^{n+1})u_t^n\Big] \\
&\quad +h^2\Big[\nu\big[4(C_2^{n+1} + \tilde{C}_2^{n+1})u_{xt}^n + 3(C_1^{n+1} - \tilde{C}_1^{n+1})u_{yt}^n\big] \\
&\qquad + \tfrac{1}{2}\big[4(C_3^{n+1} + \tilde{C}_3^{n+1})u_{xx}^n + 2(C_2^{n+1} - \tilde{C}_2^{n+1})u_{xy}^n + \tilde{C}_1^{n+1}u_{yy}^n \\
&\qquad + 4\nu^2(C_1^{n+1} + \tilde{C}_1^{n+1})u_{tt}^n\big]\Big] \\
&\quad +\tfrac{1}{2}h^3\Big[\nu\big[4(C_3^{n+1} + \tilde{C}_3^{n+1})u_{xxt}^n + 2(C_2^{n+1} - \tilde{C}_2^{n+1})u_{xyt}^n + \tilde{C}_1^{n+1}u_{yyt}^n\big] \\
&\qquad + \nu^2\big[4(C_2^{n+1} + \tilde{C}_2^{n+1})u_{xtt}^n + 3(C_1^{n+1} - \tilde{C}_1^{n+1})u_{ytt}^n\big] + \tfrac{4}{3}\nu^3(C_1^{n+1} + \tilde{C}_1^{n+1})u_{ttt}^n \\
&\qquad + \tfrac{1}{3}\big[4(C_2^{n+1} + \tilde{C}_2^{n+1})u_{xxx}^n + 3(C_3^{n+1} - \tilde{C}_3^{n+1})u_{xxy}^n + 3\tilde{C}_2^{n+1}u_{xyy}^n - \tilde{C}_1^{n+1}u_{yyy}^n\big]\Big] \\
&\qquad + O(h^4).
\end{aligned}
\tag{3.61}
$$

Since the coefficients in the above equations involve $\Theta_{i,j}^{x,n}$ and $\Theta_{i,j}^{y,n}$ we are writing the expansions of these terms. For clarity purpose, we repeat the definition of $\Theta_{i,j}^{x,n}$ and $\Theta_{i,j}^{y,n}$ here. $\Theta_{i,j}^{x,n}$ and $\Theta_{i,j}^{y,n}$ were defined in (3.11) by

$$
\Theta_{i,j}^{x,n} = -\tfrac{1}{4}\frac{\nu}{h}\Big\{[\tilde{v}_x]_{i,j}^n + [\tilde{v}_x]_{i,j}^{n+1}\Big\} + \tfrac{1}{8}\frac{\nu^2}{h}[\tilde{v}_{xy}]_{i,j}^n\Big\{[\tilde{v}_y]_{i,j}^n + [\tilde{v}_y]_{i,j}^{n+1}\Big\},
$$

$$
\Theta_{i,j}^{y,n} = -\tfrac{1}{4}\frac{\nu}{h}\Big\{[\tilde{v}_y]_{i,j}^n + [\tilde{v}_y]_{i,j}^{n+1}\Big\} + \tfrac{1}{8}\frac{\nu^2}{h}[\tilde{v}_{xy}]_{i,j}^n\Big\{[\tilde{v}_x]_{i,j}^n + [\tilde{v}_x]_{i,j}^{n+1}\Big\}.
$$

where

$$
[\tilde{v}_x]_{i,j}^n = v_{i+1,j}^n - v_{i,j}^n + v_{i+1,j+1}^n - v_{i,j+1}^n, \qquad [\tilde{v}_y]_{i,j}^n = v_{i,j+1}^n - v_{i,j}^n + v_{i+1,j+1}^n - v_{i+1,j}^n,
$$

and

$$
[\tilde{v}_{xy}]_{i,j}^n = v_{i+1,j+1}^n - v_{i+1,j}^n + v_{i,j}^n - v_{i,j+1}^n.
$$

Define $v_{i,j}^n = v(\bar{x}_i, \bar{y}_j, t_n)$, then

$$v_{i+1,j}^n - v_{i,j}^n = \sum_{m=1}^{p} \frac{(\partial_x^m v)_{i,j}^n}{m!} h^m + O(h^{p+1}),$$

where $(\partial_x^m v)_{i,j}^n = \dfrac{\partial^m v}{\partial x^m}(\bar{x}_i, \bar{y}_j, t_n)$, and

$$v_{i,j}^n - v_{i-1,j}^n = -(v_{i-1,j}^n - v_{i,j}^n) = \sum_{m=1}^{p} \frac{(-1)^{m+1}}{m!} (\partial_x^m v)_{i,j}^n h^m + O(h^{p+1}).$$

Hence

$$\sum_{m=1}^{p} \frac{(i_0)^{m+1}}{m!} (\partial_x^m v)_{i,j}^n h^m + O(h^{p+1}) = \begin{cases} v_{i+1,j}^n - v_{i,j}^n, & i_0 = 1 \\ v_{i,j}^n - v_{i-1,j}^n, & i_0 = -1 \end{cases}.$$

By setting

$$E_{i,j}^n(i_0, j_0) = \sum_{m=1}^{p} \left[\sum_{k=0}^{m-1} \frac{j_0^k}{k!} \frac{(i_0)^{m-k+1}}{(m-k)!} (\partial_y^k \partial_x^{m-k} v)_{i,j} \right] h^m + O(h^{p+1}), \qquad i_0 = -1, 1, \quad j_0 = -1, 0, 1$$

we have

$$E_{i,j}^n(i_0, 0) = \begin{cases} v_{i+1,j}^n - v_{i,j}^n, & i_0 = 1 \\ v_{i,j}^n - v_{i-1,j}^n, & i_0 = -1 \end{cases}.$$

We set $[\tilde{v}_x^n]_{i,j}(i_0, j_0) = E_{i,j}^n(i_0, 0) + E_{i,j}^n(i_0, j_0)$. By setting

$$\varepsilon_{i,j}^n(i_0, j_0) = E_{i,j}^n(i_0, 0) + E_{i,j}^n(i_0, j_0),$$

we can write the expansion at time t_{n+1} as

$$\begin{aligned} [\tilde{v}_x]_{i,j}^{n+1}(i_0, j_0) = \varepsilon_{i,j}^{n+1}(i_0, j_0) = \sum_{m=1}^{p} \frac{(i_0)^{m+1}}{m!} \Bigg\{ & \sum_{l=0}^{m-1} (i_0)^l \binom{m}{l} \nu^l \Bigg[2 \big(\partial_t^l \partial_x^{m-l} v \big)_{i,j}^n \\ & + \sum_{k=1}^{m-k-l} (i_0)^k (j_0)^k \binom{m-l}{k} \big(\partial_t^l \partial_y^k \partial_x^{m-k-l} v \big)_{i,j}^n \Bigg] \Bigg\} h^m + O(h^{p+1}). \end{aligned}$$

The mix derivatives are given by

$$[\tilde{v}_{xy}]_{i,j}^n(i_0, j_0) = \sum_{m=2}^{p} \frac{1}{m!} \left[\sum_{k=1}^{m-1} (i_0)^{m-k+1} (j_0)^{k+1} \binom{m}{k} (\partial_y^k \partial_x^{m-k} v)_{i,j}^n \right] h^m + O(h^{p+1}),$$

and

$$[\tilde{v}_{xy}]^{n+1}_{i,j}(i_0,j_0)$$
$$= \sum_{m=2}^{p} \frac{1}{m!}\left\{ \sum_{l=0}^{m-1} (i_0)^l \binom{m}{l} \nu^l \left[\sum_{k=1}^{m-k-l} (i_0)^{m-k+1}(j_0)^{k+1} \binom{m-l}{k} (\partial_t^l \partial_y^k \partial_x^{m-k-l} v)^n_{i,j} \right] \right\} h^m$$
$$+ O(h^{p+1})$$

Using the above expansions we get

$$[\tilde{v}_x]^n_{i,j}(i_0,j_0) + [\tilde{v}_x]^{n+1}_{i,j}(i_0,j_0)$$
$$= 4hv^n_x + 2h^2(i_0 v^n_{xx} + j_0 v^n_{xy} + \nu v^n_{tx})$$
$$+ \tfrac{1}{6}h^3(4v^n_{xxx} + 6i_0 j_0 v^n_{xxy} + 6v^n_{xyy} + 6i_0\nu v^n_{txx} + 6j_0\nu v^n_{txy} + 6\nu^2 v^n_{ttx})$$
$$+ \tfrac{1}{24}h^4(i_0 4v^n_{xxxx} + 8j_0 v^n_{xxxy} + 8i_0 v^n_{xxyy} + 8j_0 v^n_{xyyy} + 8\nu v^n_{txxx} + 12i_0 j_0 \nu v^n_{txxy}$$
$$+ 12\nu v^n_{txyy} + 12i_0\nu^2 v^n_{ttxx} + 12i_0\nu^2 v^n_{ttxy} + 8\nu^3 v^n_{tttx}) + O(h^5),$$

and

$$[\tilde{v}_y]^n_{i,j}(i_0,j_0) + [\tilde{v}_y]^{n+1}_{i,j}(i_0,j_0)$$
$$= 4hv^n_y + 2h^2(i_0 v^n_{yy} + j_0 v^n_{xy} + \nu v^n_{ty})$$
$$+ \tfrac{1}{6}h^3(4v^n_{yyy} + 6i_0 j_0 v^n_{xxy} + 6v^n_{xyy} + 6i_0\nu v^n_{tyy} + 6j_0\nu v^n_{txy} + 6\nu^2 v^n_{tty})$$
$$+ \tfrac{1}{24}h^4(i_0 4v^n_{yyyy} + 8j_0 v^n_{xxxy} + 8i_0 v^n_{xxyy} + 8j_0 v^n_{xyyy} + 8\nu v^n_{tyyy} + 12i_0 j_0 \nu v^n_{txyy}$$
$$+ 12\nu v^n_{txxy} + 12i_0\nu^2 v^n_{ttyy} + 12i_0\nu^2 v^n_{ttxy} + 8\nu^3 v^n_{ttty}) + O(h^5).$$

Also, we have

$$[\tilde{v}_{xy}]^n_{i,j}(i_0,j_0) = h^2 v^n_{xy} + \tfrac{1}{6}h^3(3i_0 v^n_{xxy} + 3j_0 v^n_{xyy}) + O(h^4),$$

$$[\tilde{v}_{xy}]^{n+1}_{i,j}(i_0,j_0) = h^2 v^n_{xy} + \tfrac{1}{6}h^3(3i_0 v^n_{xxy} + 3j_0 v^n_{xyy} + 6\nu v^n_{txy}) + O(h^4).$$

Hence the expansions of $\Theta^{x,n}_{i,j}$ are given as follows

$$\Theta^{x,n}_{i,j} = -\tfrac{1}{4}\frac{\nu}{h}\Big\{[\tilde{v}_x]^n_{i,j}(1,1) + [\tilde{v}_x]^{n+1}_{i,j}(1,1)\Big\} + \tfrac{1}{8}\frac{\nu^2}{h}[\tilde{v}_{xy}]^n_{i,j}(1,1)\Big\{[\tilde{v}_y]^n_{i,j}(1,1) + [\tilde{v}_y]^{n+1}_{i,j}(1,1)\Big\}$$

$$\begin{aligned}\Theta^{x,n}_{i-1,j} &= -\tfrac{1}{4}\frac{\nu}{h}\Big\{[\tilde{v}_x]^n_{i,j}(-1,1) + [\tilde{v}_x]^{n+1}_{i,j}(-1,1)\Big\}\\ &\quad + \tfrac{1}{8}\frac{\nu^2}{h}[\tilde{v}_{xy}]^n_{i,j}(-1,1)\Big\{[\tilde{v}_y]^n_{i,j}(-1,1) + [\tilde{v}_y]^{n+1}_{i,j}(-1,1)\Big\},\end{aligned}$$

$$\begin{aligned}\Theta^{x,n}_{i,j-1} &= -\tfrac{1}{4}\frac{\nu}{h}\Big\{[\tilde{v}_x]^n_{i,j}(1,-1) + [\tilde{v}_x]^{n+1}_{i,j}(1,-1)\Big\}\\ &\quad + \tfrac{1}{8}\frac{\nu^2}{h}[\tilde{v}_{xy}]^n_{i,j}(1,-1)\Big\{[\tilde{v}_y]^n_{i,j}(1,-1) + [\tilde{v}_y]^{n+1}_{i,j}(1,-1)\Big\},\end{aligned}$$

$$\begin{aligned}\Theta^{x,n}_{i-1,j-1} &= -\tfrac{1}{4}\frac{\nu}{h}\Big\{[\tilde{v}_x]^n_{i,j}(-1,-1) + [\tilde{v}_x]^{n+1}_{i,j}(-1,-1)\Big\}\\ &\quad + \tfrac{1}{8}\frac{\nu^2}{h}[\tilde{v}_{xy}]^n_{i,j}(-1,-1)\Big\{[\tilde{v}_y]^n_{i,j}(-1,-1) + [\tilde{v}_y]^{n+1}_{i,j}(-1,-1)\Big\}.\end{aligned}$$

Analogously, we can write the expansions of $\Theta^y_{i,j}$'s.

We now expand the constants in terms of the nodal values of $v(x,y,t_n)$. So, we have

$$\begin{aligned}
C^n_1 &= 2(\Theta^{x,n}_{i,j} - \Theta^{x,n}_{i-1,j}) = -2\nu h v^n_{xx} - \nu h^2\big[v^n_{xxy} + \nu v^n_{txx}\big]\\
&\quad - h^3\big[\tfrac{1}{6}\nu v^n_{xxx} + \tfrac{1}{3}\nu v^n_{xxyy} + \tfrac{1}{2}\nu^2 v^n_{txxy} + \tfrac{1}{2}\nu^3 v^n_{ttxy} + \tfrac{1}{2}\nu^2 v^n_{ttxx} - \nu^2 v^n_{xy}v^n_{xx} - \nu^2 v^n_x v^n_{xxy}\big] + O(h^4),\\
\tilde{C}^n_1 &= 2(\Theta^{x,n}_{i,j-1} - \Theta^{x,n}_{i-1,j-1}) = -2\nu h v^n_{xx} - \nu h^2\big[-v^n_{xxy} + \nu v^n_{txx}\big]\\
&\quad - h^3\big[\tfrac{1}{6}\nu v^n_{xxx} + \tfrac{1}{3}\nu v^n_{xxyy} - \tfrac{1}{2}\nu^2 v^n_{txxy} + \tfrac{1}{2}\nu^3 v^n_{ttxy} + \tfrac{1}{2}\nu^2 v^n_{ttxx} - \nu^2 v^n_{xy}v^n_{xx} - \nu^2 v^n_x v^n_{xxy}\big] + O(h^4),\\
C^n_2 &= (\Theta^{x,n}_{i,j} + \Theta^{x,n}_{i-1,j}) + (\Theta^{x,n}_{i,j})^2 - (\Theta^{x,n}_{i-1,j})^2 = -2\nu v^n_x + \nu h\big[-v^n_{xy} - \nu v^n_{tx} - 2v^n_x v^n_{xx}\big]\\
&\quad - h^2\big[\tfrac{1}{3}\nu v^n_{xxx} + \tfrac{1}{2}\nu v^n_{xyy} + \tfrac{1}{2}\nu^2 v^n_{txy} + \tfrac{1}{2}\nu^3 v^n_{tty} - \nu^2 v^n_x v^n_{xy} - \nu^2 v^n_x(v^n_{xxy} + v^n_{txx})\\
&\quad - \nu^2 v^n_{xx}(v^n_{xy} + \nu v^n_{tx})\big] + O(h^3),\\
\tilde{C}^n_2 &= (\Theta^{x,n}_{i,j-1} + \Theta^{x,n}_{i-1,j-1}) + (\Theta^{x,n}_{i,j-1})^2 - (\Theta^{x,n}_{i-1,j-1})^2 = -2\nu v^n_x + \nu h\big[v^n_{xy} - \nu v^n_{tx} + 2v^n_x v^n_{xx}\big]\\
&\quad - h^2\big[\tfrac{1}{3}\nu v^n_{xxx} + \tfrac{1}{2}\nu v^n_{xyy} - \tfrac{1}{2}\nu^2 v^n_{txy} + \tfrac{1}{2}\nu^3 v^n_{tty} - \nu^2 v^n_x v^n_{xy} - \nu^2 v^n_x(-v^n_{xxy} + v^n_{txx})\\
&\quad - \nu^2 v^n_{xx}(-v^n_{xy} + \nu v^n_{tx})\big] + O(h^3),
\end{aligned}$$

$$C_3^n = 2(\nu v_x^n)^2 + h\big[\nu v_{xx}^n + \nu^2 v_x^n(v_{xy}^n + \nu v_{tx}^n)\big] + O(h^2),$$

$$\tilde{C}_3^n = 2(\nu v_x^n)^2 + h\big[\nu v_{xx}^n + \nu^2 v_x^n(-v_{xy}^n + \nu v_{tx}^n)\big] + O(h^2).$$

By subtracting (3.60) from (3.61) we get

$$\begin{aligned}
&3\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1} - 3\omega_{i,j}^{x,n} - \tilde{\omega}_{i,j}^{x,n} = 4(C_1^{n+1} + \tilde{C}_1^{n+1} - C_1^n - \tilde{C}_1^n)u^n \\
&+ h\left[4(C_2^{n+1} + \tilde{C}_2^{n+1} - C_2^n - \tilde{C}_2^n)u_x^n + (C_1^{n+1} - \tilde{C}_1^{n+1} - C_1^n + \tilde{C}_1^n)u_y^n\right] \\
&+ 4\nu h(C_1^{n+1} + \tilde{C}_1^{n+1})u_t^n + \nu h^2\left[4(C_2^{n+1} + \tilde{C}_2^{n+1})u_{xt} + 3(C_1^{n+1} - \tilde{C}_1^{n+1})u_{yt}^n\right] \\
&+ \tfrac{1}{2}h^2\left[4(C_3^{n+1} + \tilde{C}_3^{n+1} - C_3^n - \tilde{C}_3^n)u_{xx}^n + 2(C_2^{n+1} - \tilde{C}_2^{n+1} - C_2^n + \tilde{C}_2^n)u_{xy}^n + (\tilde{C}_1^{n+1} - \tilde{C}_1^n)u_{yy}^n\right] \\
&+ \tfrac{1}{2}\nu h^3\left[4(C_3^{n+1} + \tilde{C}_3^{n+1})u_{xxt}^n + 2(C_2^{n+1} - \tilde{C}_2^{n+1})u_{xyt}^n + \tilde{C}_1^{n+1}u_{yyt}^n\right] \\
&+ 2(\nu h)^2(C_1^{n+1} + \tilde{C}_1^{n+1})u_{tt}^n + \tfrac{1}{2}\nu^2 h^3\left[4(C_2^{n+1} + \tilde{C}_2^{n+1})u_{xtt}^n + 3(C_1^{n+1} - \tilde{C}_1^{n+1})u_{ytt}^n\right] \\
&+ \tfrac{1}{6}h^3\Big[4(C_2^{n+1} + \tilde{C}_2^{n+1} - C_2^n - \tilde{C}_2^n)u_{xxx}^n + 3(C_3^{n+1} - \tilde{C}_3^{n+1} - C_3^n + \tilde{C}_3^n)u_{xxy}^n \\
&+ 3(\tilde{C}_2^{n+1} - \tilde{C}_2^n)u_{xyy}^n - (\tilde{C}_1^{n+1} + \tilde{C}_1^n)u_{yyy}^n\Big] + \tfrac{4}{6}(\nu h)^3(C_1^{n+1} + \tilde{C}_1^{n+1})u_{ttt}^n + O(h^4).
\end{aligned} \tag{3.62}$$

The expansion of the first brackets in the above equation is given by

$$\begin{aligned}
&4(C_1^{n+1} + \tilde{C}_1^{n+1} - C_1^n - \tilde{C}_1^n)u^n \\
&\qquad = 4\big[8\nu h v_{xx}^n + 4\nu^2 h^2 v_{txx}^n + h^3(\tfrac{2}{3}\nu v_{xxx}^n + \tfrac{4}{3}\nu v_{xxyy}^n + 2\nu^3 v_{ttxy}^n + 2\nu^2 v_{ttxx}^n) + O(h^4)]u^n
\end{aligned}$$

Similarly, we expand the second brackets in (3.62) to get

$$\begin{aligned}
&h\left[4(C_2^{n+1} + \tilde{C}_2^{n+1} - C_2^n - \tilde{C}_2^n)u_x^n + (C_1^{n+1} - \tilde{C}_1^{n+1} - C_1^n + \tilde{C}_1^n)u_y^n\right] \\
&\quad = h\left[4[8\nu v_x^n + 4\nu^2 h v_{tx}^n + h^2(\tfrac{4}{3}\nu v_{xxx}^n + 2\nu v_{xyy}^n + 2\nu^3 v_{tty}^n)]u_x^n + [4\nu h^2 v_{xxy}^n + 2\nu^2 h^3 v_{txxy}^n]u_y^n\right] \\
&\quad + O(h^4),
\end{aligned}$$

Next we expand the third brackets that gives

$$
\begin{aligned}
4\nu h(C_1^{n+1}+\tilde{C}_1^{n+1})u_t^n &= 4\nu h(4\nu h v_{xx}^n + 2\nu h^2 v_{txx}^n)u_t^n + O(h^4) \\
&= 16(\nu h)^2 u_t^n v_{xx}^n + 8\nu^2 h^3 u_t^n v_{txx}^n + O(h^4).
\end{aligned}
$$

The expansion of the fourth brackets in (3.62) leads to

$$
\begin{aligned}
&\nu h^2\left[4(\tilde{C}_2^{n+1}+C_2^{n+1})u_{xt}^n + 3(C_1^{n+1}-\tilde{C}_1^{n+1})u_{yt}^n\right] \\
&\qquad = \nu h^2\left[4(4\nu v_x^n + 2\nu^2 h v_{tx}^n + O(h^2))u_{tx}^n + 3(2\nu h^2 v_{xxy}^n + O(h^3))u_{ty}^n\right] \\
&\qquad = 16\nu^2 h^2 u_{tx}^n v_x^n + 8\nu^3 h^3 u_{tx}^n v_{tx}^n + O(h^4).
\end{aligned}
$$

It is easy to see that the remaining terms in (3.62) are higher order terms. The combination of these equations gives

$$
\begin{aligned}
&\tfrac{1}{32\nu h}\left\{3\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1} - 3\omega_{i,j}^{x,n} - \tilde{\omega}_{i,j}^{x,n}\right\} \\
&\quad = u^n v_{xx}^n + u_x^n v_x^n + \tfrac{1}{2}\nu h(u^n v_{txx}^n + u_x^n v_{tx}^n + u_t^n v_{xx}^n + u_{tx}^n v_x^n) \\
&\quad + h^2\left[u^n(\tfrac{2}{3}v_{xxx}^n + \tfrac{4}{3}v_{xxyy}^n + 2\nu v_{ttxx}^n) + u_x^n(\tfrac{4}{3}v_{xxx}^n + 2v_{xyy}^n + 2\nu^2 v_{tty}^n) + 4u_y^n v_{xxy}^n + 8\nu u_t^n v_{txx}^n\right. \\
&\quad \left. + 8\nu^2 u_{tx}^n v_{tx}^n\right] + O(h^3).
\end{aligned}
\tag{3.63}
$$

The fluxes in the y-direction are expanded in the same way which gives

$$
\begin{aligned}
&\tfrac{1}{32\nu h}\left\{3\omega_{i,j}^{y,n+1} + \tilde{\omega}_{i,j}^{y,n+1} - 3\omega_{i,j}^{y,n} - \tilde{\omega}_{i,j}^{y,n}\right\} \\
&\quad = u^n v_{yy}^n + u_y^n v_y^n + \tfrac{1}{2}\nu h(u^n v_{tyy}^n + u_y^n v_{ty}^n + u_t^n v_{yy}^n + u_{ty}^n v_y^n) \\
&\quad + h^2\left[u^n(\tfrac{2}{3}v_{yyy}^n + \tfrac{4}{3}v_{xxyy}^n + 2\nu v_{ttyy}^n) + u_y^n(\tfrac{4}{3}v_{yyy}^n + 2v_{xxy}^n + 2\nu^2 v_{ttx}^n) + 4u_x^n v_{xyy}^n + 8\nu u_t^n v_{tyy}^n\right. \\
&\quad \left. + 8\nu^2 u_{ty}^n v_{ty}^n\right] + O(h^3)
\end{aligned}
\tag{3.64}
$$

Now we compute the expansion of the fluxes from the diffusion term. These expansions are given by

$$
-\omega_{i,j}^{x,n} + \tilde{\omega}_{i,j}^{x,n} = h(C_1^n - \tilde{C}_1^n)u_y^n + \tfrac{1}{2}h^2\left[2(C_2^n - \tilde{C}_2^n)u_{xy}^n + \tilde{C}_1^n u_{yy}^n\right] + O(h^3), \tag{3.65}
$$

and

$$-\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1}$$
$$= h(C_1^{n+1} - \tilde{C}_1^{n+1})u_y^n + \tfrac{1}{2}h^2\Big[2\nu(C_1^{n+1} - \tilde{C}_1^{n+1})u_{yt}^n + 2(C_2^{n+1} - \tilde{C}_2^{n+1})u_{xy}^n + \tilde{C}_1^{n+1}u_{yy}^n\Big]$$
$$+ O(h^3). \tag{3.66}$$

Subtracting (3.65) from (3.66) to get

$$\tfrac{\kappa}{8h^2}\Big[-\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1} + \omega_{i,j}^{x,n} - \tilde{\omega}_{i,j}^{x,n}\Big] = \tfrac{1}{8h^2}\Big\{h(C_1^{n+1} - \tilde{C}_1^{n+1} + C_1^n - \tilde{C}_1^n)u_y^n$$
$$+\tfrac{1}{2}h^2\Big[2\nu(C_1^{n+1} - \tilde{C}_1^{n+1} + C_1^n - \tilde{C}_1^n)u_{yt}^n + 2(C_2^{n+1} - \tilde{C}_2^{n+1} + C_2^n - \tilde{C}_2^n)u_{xy}^n + (C_1^{n+1} + C_1^n)u_{yy}^n\Big]$$
$$+ O(h^3)\Big\} \tag{3.67}$$

The expansion of the first term in the above equation gives

$$h[C_1^{n+1} - \tilde{C}_1^{n+1} + C_1^n - \tilde{C}_1^n]u_y^n = O(h^5).$$

It is easy to see that the remaining terms in (3.67) are higher order terms. Hence

$$\tfrac{\kappa}{8h^2}\Big[-\omega_{i,j}^{x,n+1} + \tilde{\omega}_{i,j}^{x,n+1} + \omega_{i,j}^{x,n} - \tilde{\omega}_{i,j}^{x,n}\Big] = O(h^3). \tag{3.68}$$

Analogously, we have

$$\tfrac{\kappa}{8h^2}\Big[-\omega_{i,j}^{y,n+1} + \tilde{\omega}_{i,j}^{y,n+1} + \omega_{i,j}^{y,n} - \tilde{\omega}_{i,j}^{y,n}\Big] = O(h^3). \tag{3.69}$$

The last expansion will be for the corner fluxes.

Set $D_{i,j}^n = a_{i,j}^n b_{i,j}^n + a_{i,j}^n B_{i,j}^n + A_{i,j}^n b_{i,j}^n + A_{i,j}^n B_{i,j}^n$. Since $a_{i,j}^n + A_{i,j}^n = 2\Theta_{i,j}^{x,n}$ and

$b^n_{i,j} + B^n_{i,j} = 2\Theta^{y,n}_{i,j}$, then we can write the sum of the corner fluxes as

$$\begin{aligned}
&H^n_{i,j} - H^n_{i-1,j} + H^n_{i,j-1} - H^n_{i-1,j-1} \\
&\qquad = (D^n_{i,j} - D^n_{i-1,j} + D^n_{i,j-1} - D^n_{i-1,j-1})u^n \\
&\qquad + 2h\Big[(A^n_{i,j}\Theta^{y,n}_{i,j} - a^n_{i-1,j}\Theta^{y,n}_{i-1,j} + A^n_{i,j-1}\Theta^{y,n}_{i,j-1} - a^n_{i-1,j-1}\Theta^{y,n}_{i-1,j-1})u^n_x \\
&\qquad + (B^n_{i,j}\Theta^{x,n}_{i,j} - b^n_{i,j-1}\Theta^{x,n}_{i,j-1} + B^n_{i-1,j}\Theta^{x,n}_{i-1,j} - b^n_{i-1,j-1}\Theta^{x,n}_{i-1,j-1})u^n_y\Big] + O(h^2).
\end{aligned} \tag{3.70}$$

Similarly, we have

$$\begin{aligned}
&H^{n+1}_{i,j} - H^{n+1}_{i-1,j} + H^{n+1}_{i,j-1} - H^{n+1}_{i-1,j-1} \\
&\qquad = (D^{n+1}_{i,j} - D^{n+1}_{i-1,j} + D^{n+1}_{i,j-1} - D^{n+1}_{i-1,j-1})u^n \\
&\qquad + 2h\Big[\tfrac{1}{2}\nu(D^{n+1}_{i,j} - D^{n+1}_{i-1,j} + D^{n+1}_{i,j-1} - D^{n+1}_{i-1,j-1})u^n_t \\
&\qquad + (A^{n+1}_{i,j}\Theta^{y,n+1}_{i,j} - a^{n+1}_{i-1,j}\Theta^{y,n+1}_{i-1,j} + A^{n+1}_{i,j-1}\Theta^{y,n+1}_{i,j-1} - a^{n+1}_{i-1,j-1}\Theta^{y,n+1}_{i-1,j-1})u^n_x \\
&\qquad + (B^{n+1}_{i,j}\Theta^{x,n+1}_{i,j} - b^{n+1}_{i,j-1}\Theta^{x,n+1}_{i,j-1} + B^{n+1}_{i-1,j}\Theta^{x,n+1}_{i-1,j} - b^{n+1}_{i-1,j-1}\Theta^{x,n+1}_{i-1,j-1})u^n_y\Big] + O(h^2).
\end{aligned} \tag{3.71}$$

The subtraction of the first terms on the right in (3.71)-(3.70) is

$$\Big[(D^{n+1}_{i,j} - D^n_{i,j}) - (D^{n+1}_{i-1,j} - D^n_{i-1,j}) + (D^{n+1}_{i,j-1} - D^n_{i,j-1}) - (D^{n+1}_{i-1,j-1} - D^n_{i-1,j-1})\Big]u^n \tag{3.72}$$

The coefficient can be simplified as

$$\begin{aligned}
D^{n+1}_{i,j} - D^n_{i,j} &= a^{n+1}_{i,j}(b^{n+1}_{i,j} + B^{n+1}_{i,j}) - a^n_{i,j}(b^n_{i,j} + B^n_{i,j}) + A^{n+1}_{i,j}(b^{n+1}_{i,j} + B^{n+1}_{i,j}) - A^n_{i,j}(b^n_{i,j} + B^n_{i,j}) \\
&= 2\big[a^{n+1}_{i,j}\Theta^{y,n+1}_{i,j} - a^n_{i,j}\Theta^{y,n}_{i,j} + A^{n+1}_{i,j}\Theta^{y,n+1}_{i,j} - A^n_{i,j}\Theta^{y,n}_{i,j}\big] \\
&= 2\big[(a^{n+1}_{i,j} + A^{n+1}_{i,j})\Theta^{y,n+1}_{i,j} - (a^n_{i,j} + A^n_{i,j})\Theta^{y,n}_{i,j}\big] \\
&= 4\big[\Theta^{x,n+1}_{i,j}\Theta^{y,n+1}_{i,j} - \Theta^{x,n}_{i,j}\Theta^{y,n}_{i,j}\big].
\end{aligned}$$

The product of $\Theta^{x,n}_{i,j}$ and $\Theta^{y,n}_{i,j}$ can be written as

$$\begin{aligned}
\Theta^{x,n}_{i,j}\Theta^{y,n}_{i,j} &= \tfrac{1}{16}\frac{\nu^2}{h^2}([\tilde{v}_x]^n_{i,j}+[\tilde{v}_x]^{n+1}_{i,j})([\tilde{v}_y]^n_{i,j}+[\tilde{v}_y]^{n+1}_{i,j})\\
&\quad-\tfrac{1}{32}\frac{\nu^3}{h^2}[\tilde{v}_{xy}]^n_{i,j}\Big[([\tilde{v}_x]^n_{i,j}+[\tilde{v}_x]^{n+1}_{i,j})^2+([\tilde{v}_y]^n_{i,j}+[\tilde{v}_y]^{n+1}_{i,j})^2\Big]\\
&\quad+\tfrac{1}{64}\frac{\nu^4}{h^2}([\tilde{v}_{xy}]^n_{i,j})^2([\tilde{v}_x]^n_{i,j}+[\tilde{v}_x]^{n+1}_{i,j})([\tilde{v}_y]^n_{i,j}+[\tilde{v}_y]^{n+1}_{i,j}).
\end{aligned}$$

Therefore, we get

$$\begin{aligned}
&\Theta^{x,n+1}_{i,j}\Theta^{y,n+1}_{i,j}-\Theta^{x,n}_{i,j}\Theta^{y,n}_{i,j}\\
&\quad=\tfrac{1}{32}\frac{\nu^3}{h^2}([\tilde{v}_{xy}]^{n+1}_{i,j}+[\tilde{v}_{xy}]^n_{i,j})\Big[([\tilde{v}_x]^{n+1}_{i,j}+[\tilde{v}_x]^n_{i,j})^2+([\tilde{v}_y]^{n+1}_{i,j}+[\tilde{v}_y]^{n+1}_{i,j})^2\Big]\\
&\quad+\tfrac{1}{64}\frac{\nu^4}{h^2}\Big[([\tilde{v}_{xy}]^{n+1}_{i,j})^2-([\tilde{v}_{xy}]^n_{i,j})^2\Big]\Big[\big([\tilde{v}_x]^{n+1}_{i,j}+[\tilde{v}_x]^n_{i,j}\big)\big([\tilde{v}_y]^{n+1}_{i,j}+[\tilde{v}_y]^{n+1}_{i,j}\big)\Big]\\
&\quad=\tfrac{1}{32}\frac{\nu^3}{h^2}\Big[\big[2h^2v^n_{xy}+h^3(v^n_{xxy}+v^n_{xyy}+\nu v^n_{txy})\big]\big[16h^2(v^n_x+v^n_y)+O(h^3)\big]\Big]\\
&\quad+\tfrac{1}{64}\tfrac{\nu^4}{h^2}\Big[(2\nu h^5v^n_{xy})\big[16h^2v^n_xv^n_y+O(h^3)\big]\Big]\\
&\quad=\nu^2h^2v^n_{xy}(v^n_x+v^n_y)+\tfrac{1}{2}\nu^3h^3(v^n_{xxy}+v^n_{xyy}+\nu v^n_{txy})+O(h^4).
\end{aligned}$$

Similarly we obtain the other product terms. Therefore we have

$$\begin{aligned}
&\Big[(D^{n+1}_{i,j}-D^n_{i,j})-(D^{n+1}_{i-1,j}-D^n_{i-1,j})+(D^{n+1}_{i,j-1}-D^n_{i,j-1})-(D^{n+1}_{i-1,j-1}-D^n_{i-1,j-1})\Big]u\\
&\qquad=8\nu^3h^3u^nv^n_{xxy}(v^n_x+v^n_y)+O(h^4).
\end{aligned}$$

The subtraction of the second terms on the right in (3.71)-(3.70) is given by

$$\begin{aligned}
2h\Big\{&\tfrac{1}{2}\nu(D^{n+1}_{i,j}-D^{n+1}_{i-1,j}+D^{n+1}_{i,j-1}-D^{n+1}_{i-1,j-1})u^n_t\\
&+\Big[(A^{n+1}_{i,j}\Theta^{y,n+1}_{i,j}-A^n_{i,j}\Theta^{y,n}_{i,j})-(a^{n+1}_{i-1,j}\Theta^{y,n+1}_{i-1,j}-a^n_{i-1,j}\Theta^{y,n}_{i-1,j})\\
&+(A^{n+1}_{i,j-1}\Theta^{y,n}_{i,j-1}-A^n_{i,j-1}\Theta^{y,n}_{i,j-1})-(a^{n+1}_{i-1,j-1}\Theta^{y,n+1}_{i-1,j-1}-a^n_{i-1,j-1}\Theta^{y,n}_{i-1,j-1})\Big]u^n_x\\
&+\Big[(B^{n+1}_{i,j}\Theta^{x,n+1}_{i,j}-B^n_{i,j}\Theta^{x,n}_{i,j})-(b^{n+1}_{i-1,j}\Theta^{x,n+1}_{i-1,j}-b^n_{i-1,j}\Theta^{x,n}_{i-1,j})\\
&+(B^{n+1}_{i,j-1}\Theta^{x,n}_{i,j-1}-B^n_{i,j-1}\Theta^{x,n}_{i,j-1})-(b^{n+1}_{i-1,j-1}\Theta^{x,n+1}_{i-1,j-1}-b^n_{i-1,j-1}\Theta^{x,n}_{i-1,j-1})\Big]u^n_y\Big\}.
\end{aligned}\tag{3.73}$$

From the previous argument, we see that

$\frac{1}{2}\nu(D^{n+1}_{i,j} - D^{n+1}_{i-1,j} + D^{n+1}_{i,j-1} - D^{n+1}_{i-1,j-1})u^n_t = O(h^3).$

For the coefficient of u_x, we will seek for the order of the first brackets and the remaining terms are treated in the same way. By ignoring the higher order terms we get

$$\begin{aligned}
A^{n+1}_{i,j}\Theta^{y,n+1}_{i,j} - A^n_{i,j}\Theta^{y,n}_{i,j} &= (A^{n+1}_{i,j} - A^n_{i,j})\Theta^{y,n+1}_{i,j} + A^n_{i,j}(\Theta^{y,n+1}_{i,j} - \Theta^{y,n}_{i,j}) \\
&\approx (\Theta^{x,n+1}_{i,j} - \Theta^{x,n}_{i,j})\Theta^{y,n+1}_{i,j} + \Theta^{x,n}_{i,j}(\Theta^{y,n+1}_{i,j} - \Theta^{y,n}_{i,j}) \\
&\approx \tfrac{1}{16}\frac{\nu^3}{h^2}\Big[[\tilde{v}_{xy}]^{n+1}_{i,j}\big([\tilde{v}_x]^n_{i,j} + [\tilde{v}_x]^{n+1}_{i,j}\big)^2 + [\tilde{v}_{xy}]^n_{i,j}\big([\tilde{v}_y]^n_{i,j} + [\tilde{v}_y]^{n+1}_{i,j}\big)^2\Big] \\
&+ \tfrac{1}{32}\frac{\nu^3}{h^2}\big([\tilde{v}_{xy}]^{n+1}_{i,j} - [\tilde{v}_{xy}]^n_{i,j}\big)\Big[\big([\tilde{v}_y]^{n+1}_{i,j} + [\tilde{v}_y]^n_{i,j}\big)^2 - \big([\tilde{v}_x]^n_{i,j} + [\tilde{v}_x]^{n+1}_{i,j}\big)^2\Big] \\
&+ \tfrac{1}{64}\frac{\nu^4}{h^2}\big([\tilde{v}_{xy}]^{n+1}_{i,j} + [\tilde{v}_{xy}]^n_{i,j}\big)\big([\tilde{v}_{xy}]^{n+1}_{i,j} - [\tilde{v}_{xy}]^n_{i,j}\big)\big([\tilde{v}_x]^{n+1}_{i,j} + [\tilde{v}_x]^n_{i,j}\big)\big([\tilde{v}_y]^{n+1}_{i,j} + [\tilde{v}_y]^n_{i,j}\big) \\
&= \nu^2h^2\big((v^n_x)^2 + (v^n_y)^2\big) + O(h^3).
\end{aligned}$$

Similarly we get from the coefficient of u^n_y

$$B^{n+1}_{i,j}\Theta^{x,n+1}_{i,j} - B^n_{i,j}\Theta^{x,n}_{i,j} = \nu^2h^2\big((v^n_x)^2 + (v^n_y)^2\big) + O(h^3).$$

Combining all of these results, we see that

$$\begin{aligned}
2h\Big\{&\tfrac{1}{2}\nu(D^{n+1}_{i,j} - D^{n+1}_{i-1,j} + D^{n+1}_{i,j-1} - D^{n+1}_{i-1,j-1})u^n_t \\
&+ \Big[(A^{n+1}_{i,j}\Theta^{y,n+1}_{i,j} - A^n_{i,j}\Theta^{y,n}_{i,j}) - (a^{n+1}_{i-1,j}\Theta^{y,n+1}_{i-1,j} - a^n_{i-1,j}\Theta^{y,n}_{i-1,j}) \\
&+ (A^{n+1}_{i,j-1}\Theta^{y,n}_{i,j-1} - A^n_{i,j-1}\Theta^{y,n}_{i,j-1}) - (a^{n+1}_{i-1,j-1}\Theta^{y,n+1}_{i-1,j-1} - a^n_{i-1,j-1}\Theta^{y,n}_{i-1,j-1})\Big]u^n_x \\
&+ \Big[(B^{n+1}_{i,j}\Theta^{x,n+1}_{i,j} - B^n_{i,j}\Theta^{x,n}_{i,j}) - (b^{n+1}_{i-1,j}\Theta^{x,n+1}_{i-1,j} - b^n_{i-1,j}\Theta^{x,n}_{i-1,j}) \\
&+ (B^{n+1}_{i,j-1}\Theta^{x,n}_{i,j-1} - B^n_{i,j-1}\Theta^{x,n}_{i,j-1}) - (b^{n+1}_{i-1,j-1}\Theta^{x,n+1}_{i-1,j-1} - b^n_{i-1,j-1}\Theta^{x,n}_{i-1,j-1})\Big]u^n_y\Big\} = O(h^4).
\end{aligned}$$

Similarly, the other terms in the difference between (3.70) and (3.71) are higher order terms. Hence

$$\begin{aligned}
&\tfrac{1}{8(\nu h)}\Big\{H^{n+1}_{i,j} - H^{n+1}_{i-1,j} + H^{n+1}_{i,j-1} - H^{n+1}_{i-1,j-1} - H^n_{i,j} + H^n_{i-1,j} - H^n_{i,j-1} + H^n_{i-1,j-1}\Big\} \\
&\qquad = 8\nu^2h^2u^nv^n_{xxy}(v^n_x + v^n_y) + O(h^3).
\end{aligned} \tag{3.74}$$

Now we combine these results to write the truncation error. The sum of (3.48) and (3.49) leads to

$$\begin{aligned}
&\tfrac{\kappa}{16(\nu h)}\big[\Delta^x_{i,j}\{u^{n+1}\}+\Delta^y_{i,j}\{u^{n+1}\}+\Delta^x_{i,j}\{u^n\}+\Delta^y_{i,j}\{u^n\}\big]\\
&\qquad\qquad = (u^n_{xx}+u^n_{yy})+\tfrac{1}{2}(\nu h)(u^n_{xxt}+u^n_{yyt})+\tfrac{1}{4}(\nu h)^2(u^n_{xxtt}+u^n_{yytt})\\
&\qquad\qquad + h^2\big(\tfrac{1}{12}(u^n_{xxxx}+u^n_{yyyy})+\tfrac{1}{4}u^n_{xxyy}\big)+O(h^3).
\end{aligned} \tag{3.75}$$

By subtracting (3.75) from (3.47) we get

$$\begin{aligned}
&\tfrac{1}{64(\nu h)}\big[I^{n+1}_{i,j}-I^n_{i,j}\big]-\tfrac{\kappa}{16(\nu h)}\big[\Delta^x_{i,j}\{u^{n+1}\}+\Delta^y_{i,j}\{u^{n+1}\}+\Delta^x_{i,j}\{u^n\}+\Delta^y_{i,j}\{u^n\}\big]\\
&\quad = u^n_t-\kappa(u^n_{xx}+u^n_{yy})+\tfrac{1}{2}(\nu h)\tfrac{\partial}{\partial t}\big[u^n_t-\kappa(u^n_{xx}+u^n_{yy})\big]\\
&\quad + h^2\big[\tfrac{1}{8}(u^n_{xxt}+u^n_{yyt})-\tfrac{1}{12}\kappa(u^n_{xxxx}+u^n_{yyyy})-\tfrac{1}{4}\kappa u^n_{xxyy}+\nu^2(\tfrac{1}{6}u^n_{ttt}-\tfrac{1}{4}\kappa(u^n_{xxtt}+u^n_{yytt}))\big]\\
&\quad + O(h^3).
\end{aligned} \tag{3.76}$$

From the combination of the equations (3.63), (3.64), (3.68), (3.69), (3.74) and (3.76) we get

$$\begin{aligned}
\tau^{n+1}_{i,j} &= \Big[u^n_t-\kappa(u^n_{xx}+u^n_{yy})+u^n(v^n_{xx}+v^n_{yy})+u^n_xv^n_x+u^n_yv^n_y\Big]\\
&+\tfrac{1}{2}\nu h\tfrac{\partial}{\partial t}\Big[u^n_t-\kappa(u^n_{xx}+u^n_{yy})+u^n(v^n_{xx}+v^n_{yy})+u^n_xv^n_x+u^n_yv^n_y\Big]\\
&+h^2\Big[u^n(\tfrac{2}{3}v^n_{xxx}+\tfrac{4}{3}v^n_{xxyy}+2\nu v^n_{ttxx}+\tfrac{2}{3}v^n_{yyy}+\tfrac{4}{3}v^n_{xxyy}+2\nu v^n_{ttyy})\\
&+u^n_x(\tfrac{4}{3}v^n_{xxx}+2v^n_{xyy}+2\nu^2v^n_{tty})+u^n_y(\tfrac{4}{3}v^n_{yyy}+2v^n_{xxy}+2\nu^2v^n_{ttx})+4u^n_xv^n_{xyy}+8\nu u^n_tv^n_{tyy}\\
&+8\nu^2u^n_{ty}v^n_{ty}+4u^n_yv^n_{xxy}+8\nu u^n_tv^n_{txx}+8\nu^2u^n_{tx}v^n_{tx}+8\nu^2u^nv^n_{xxy}(v^n_x+v^n_y)+\tfrac{1}{8}(u^n_{xxt}+u^n_{yyt})\\
&-\tfrac{1}{12}\kappa(u^n_{xxxx}+u^n_{yyyy})-\tfrac{1}{4}\kappa u^n_{xxyy}+\nu^2(\tfrac{1}{6}u^n_{ttt}-\tfrac{1}{4}\kappa(u^n_{xxtt}+u^n_{yytt}))\Big]+O(h^3).
\end{aligned} \tag{3.77}$$

Since the first two brackets are zero being the left hand side of the differential equation, the scheme (3.37) is of second order. □

3.5 Comparing the unperturbed scheme with Crank-Nicolson method

This section is devoted to compare the unperturbed scheme (3.42) (when $\nabla v = 0$) with the Crank-Nicolson method. It is well known that the Crank-Nicolson method is second order. Also, it is not hard to see that the unperturbed scheme is second order too (c.f. (3.76)). Both of the methods are implicit and unconditionally stable. To compare the accuracy, we will apply the two schemes on the heat equation

$$u_t = \kappa \nabla u + g(x, y, t)$$
$$\frac{\partial u}{\partial n} = 0, \qquad u(x, y, 0) = 1,$$

where the source term $g(x, y, t)$ is chosen so that the equation has the exact solution

$$u(x, y, t) = e^{2t\cos(3x\pi)\cos(2y\pi)}.$$

The local error term of the unperturbed scheme in absence of the source term is given by

$$\tau_{i,j}^{n+1} = \tfrac{1}{8}(u_{xxt}^n + u_{yyt}^n) - \tfrac{1}{12}\kappa(u_{xxxx}^n + u_{yyyy}^n) - \tfrac{1}{4}\kappa u_{xxyy}^n + \nu^2(\tfrac{1}{6}u_{ttt}^n - \tfrac{1}{4}\kappa(u_{xxtt}^n + u_{yytt}^n)),$$

where the derivatives are evaluated at the point $(\bar{x}_i, \bar{y}_j, t_n)$ and the increments are $\Delta x = \Delta y = h$ and $\Delta t = \nu h$. However, the C-N method has the local error term

$$\tilde{\tau}_{i,j}^{n+1} = -\kappa\left[\tfrac{1}{12}(u_{xxxx}^n + u_{yyyy}^n) + \tfrac{1}{4}\nu^2(u_{xxtt}^n + u_{yytt}^n)\right]$$

It is clear that the local error term $\tau_{i,j}^{n+1}$ can be written as

$$\tau_{i,j}^{n+1} = \tilde{\tau}_{i,j}^{n+1} + \tfrac{1}{8}(u_{xxt}^n + u_{yyt}^n) - \tfrac{1}{4}\kappa u_{xxyy}^n + \nu^2 \tfrac{1}{6}u_{ttt}^n.$$

Tables (3.1) and (3.2) show that the C-N method is slightly more accurate than the unperturbed scheme. The induced linear systems in both methods were solved using conjugate gradient method and the errors UE_k were computed at time $t = 1$ using the Frobenius norm.

Table 3.1 Rates of Convergence Using Unperturbed Scheme

k	N	M	UE_k	UE_k/UE_{k+1}
1	50	20	8.87e-02	0.00e+00
2	100	40	2.13e-02	4.17e+00
3	200	80	5.26e-03	4.04e+00

Table 3.2 Rates of Convergence Using C-N Method

k	N	M	UE_k	UE_k/UE_{k+1}
1	50	20	5.91e-02	0.00e+00
2	100	40	1.42e-02	4.16e+00
3	200	80	3.52e-03	4.04e+00

CHAPTER 4. NUMERICAL EXPERIMENTS

In this chapter, numerical experiments are given to confirm the facts shown analytically in the previous chapters. All the numerical experiments concerning the 1D chemotaxis system are given first. The 2D system experiments will be given in the second section.

4.1 Numerical experiments for the 1D chemotaxis system

4.1.1 Positivity preserving property

To confirm the positivity preserving property of the scheme (2.20), we applied the method on the equation

$$u_t + (v_x u)_x = 0.6 u_{xx},$$
$$u_x(0,t) = u_x(1,t) = 0$$

with a function $v(x,t)$ known in advance. We choose different cases so that we can cover many situations. In the sense of chemotaxis, the population density $u(x,t)$ is expected to leave the low concentration area toward the area where there is a high concentration of chemotactant $v(x,t)$. We used this fact in our experiments. In these experiments, the total time steps (N = 40000), space steps (M =40) and final time (T = 100) were used.

Case 1: Mass moving away from the centre

Let $v(x,t)$ be given by

$$v(x,t) = 2(-1 + \tfrac{t}{t+1} + \cos(\tfrac{2t}{t+1} x(x-1)\pi)).$$

It is clear that $v(x,t)$ satisfies the conditions: $v_x(0,t) = v_x(1,t) = v(x,0) = 0$. Also, it converges to the steady state

$$v(x) = 2\cos(2x(x-1)\pi).$$

In this case, the concentration of the chemotactant moves toward the boundaries of the domain and as a consequence the population density is moving away from the centre. Figure 4.1 and 4.2, in which we used initial condition $u(x,0) = 1$, show this phenomenon.

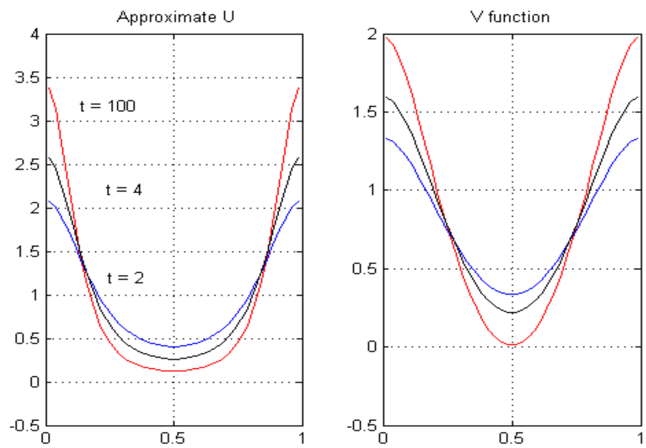

Figure 4.1 Response of the cell density, $u(x,t)$, to the chemoattractant.

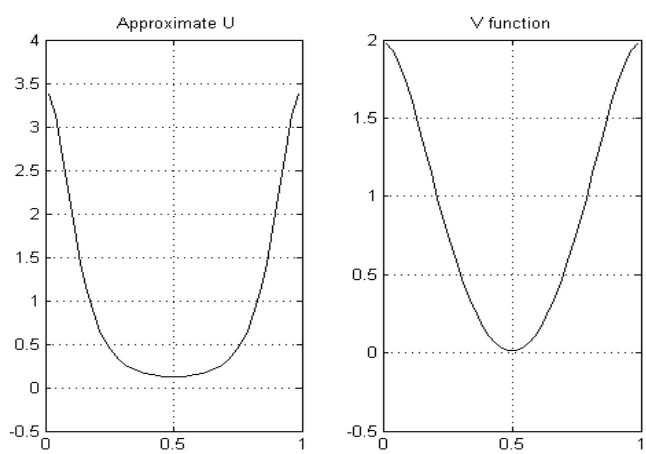

Figure 4.2 Final time $T = 100$ and initial condition $u(x,0) = 1$.

Case 2: Mass moving to the right edge

Let $v(x,t)$ be given by

$$v(x,t) = \begin{cases} 0 & x \le 0.5 \\ \frac{2t}{t+1}\cos(2x(x-1)\pi). \end{cases}$$

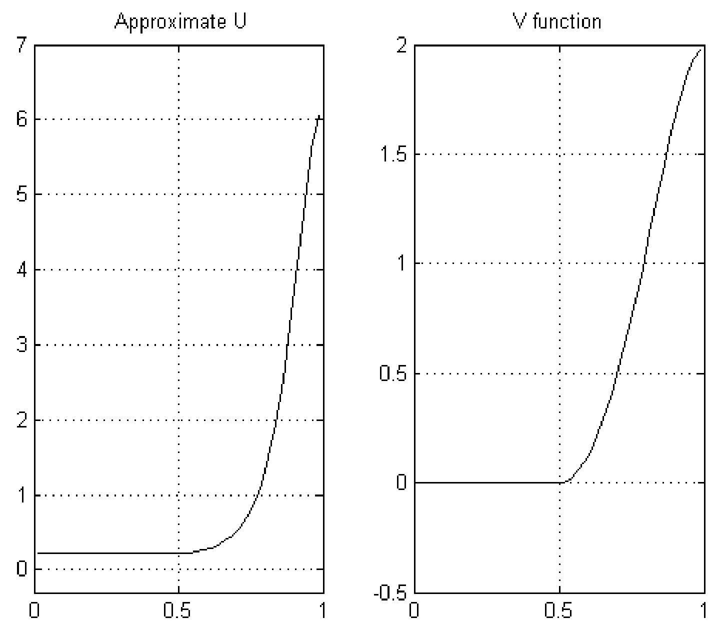

Figure 4.3 Right edge concentration: final time $T = 100$ and initial condition $u(x,0) = 1$.

The required conditions, $v_x(0,t) = v_x(1,t) = v(x,0) = 0$, are satisfied by this function. Obviously, this function converges to

$$v(x) = \begin{cases} 0 & x \le 0.5 \\ 2\cos(2x(x-1)\pi). \end{cases}$$

In this case we let the chemical substance move to the right edge of the domain. It follows that the population $u(x,t)$ is moving to the right edge (Figure 4.3). In Figure 4.3, the initial condition $u(x,0) = 1$ was used.

Case 3: Dead spot, $v(x,t) = 0$ for some range $0 < x_1 \leq x \leq x_2 < 1$

$$v(x,t) = \begin{cases} \frac{1.5t}{t+1}\cos(\frac{x\pi}{0.6}) & x \leq 0.3 \\ 0 \quad 0.3 \leq x \leq 0.7 \\ \frac{1.5t}{t+1}\cos(\frac{(x-1)\pi}{0.6}) \end{cases}$$

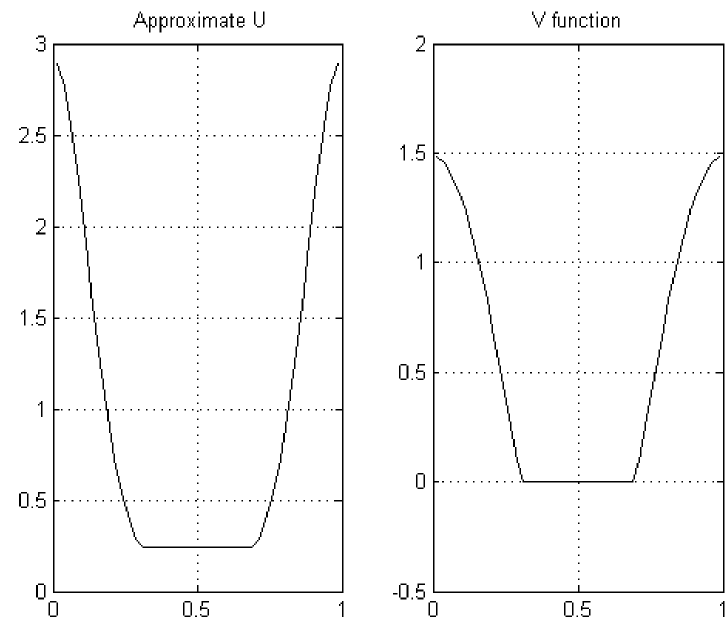

Figure 4.4 Dead spot: final time $T = 100$ and initial condition $u(x,0) = 1$.

An interesting situation is that when there is a dead spot $v(x,t) = 0$ for some range $0 < x_1 \leq x \leq x_2 < 1$. This is the case we have now. Experimenting with this case is given in Figure 4.4 with initial condition $u(x,0) = 1$.

Case 4 : Clustering in the middle of the domain

The last case in our experiment is when the chemical substances are clustering in the middle of the domain. Due to instability in this case with $N = 40000$ for $v(x,t)$ given by

$$v(x,t) = \begin{cases} 0 & x \leq 0.3 \\ \frac{1.5t}{t+1}(1-\cos\left(\frac{(x-0.3)(x-0.7)\pi}{(0.4)^2}\right)) & 0.3 \leq x \leq 0.7 \\ 0 & x \geq 0.7 \end{cases}$$

we shrink the time steps by choosing $N = 70000$. Figure 4.5 shows that the mass is moving toward the centre making a pile.

Conclusion: It is clear that non of these results gave negative values even with small initial conditions. Therefore, the scheme (2.20) might be positivity preserving.

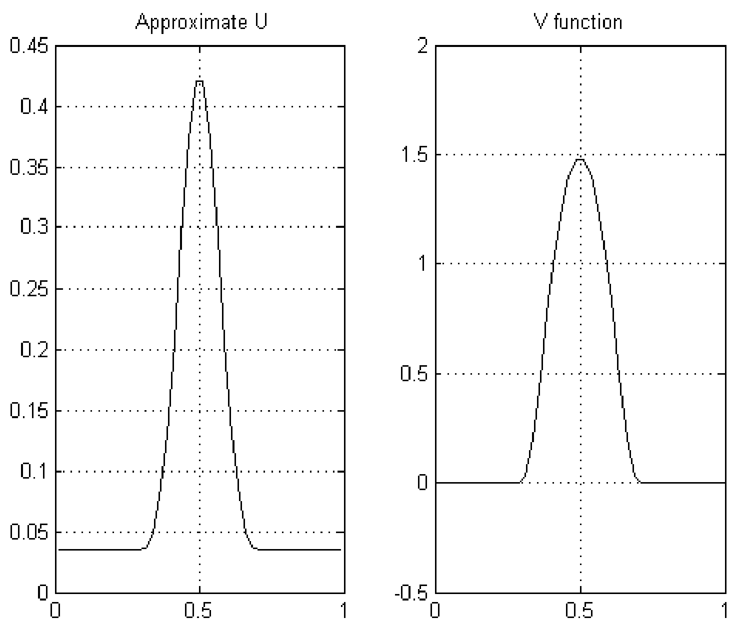

Figure 4.5 Centre clustering: final time $T = 100$ and initial condition $u(x,0) = 0.1$.

4.1.2 Studying the stability numerically

It was shown in Section 2.6 that the scheme (2.20) with fixed Δx is conditionally stable and the stability depends on the interaction between two parameters (i.e r_κ and θ_i^n). In order to confirm this situation, we tested the scheme with different values of

$r_\kappa = \frac{\kappa \Delta t}{(\Delta x)^2}$. The scheme was applied on the equation

$$u_t + (uv_x)_x = \kappa u_{xx}$$
$$u_x(0,t) = u_x(1,t) = 0, \quad x \in (0,1), \quad t \geq 0$$

with $v(x,t) = t(1+\cos(3\pi x))$ and initial condition $u(x,0) = 1$. The parameter θ_i^n is given by

$$\theta_i^n = \frac{x(t_n) - x_i}{\Delta x} = -\frac{1}{\Delta x} \int\limits_{t_n}^{t_{n+1}} v_x(x,s)ds \approx -\frac{\Delta t}{2\Delta x}[v_x(x,t_n) + v_x(x,t_{n+1})]$$

where $v_x = -3\pi t \sin(3\pi x)$. Hence, we have that

$$\begin{aligned} |\theta_i^n| \leq \frac{3\pi}{2\Delta x}[t_{n+1} + t_n]\Delta t = \frac{3\pi \Delta t}{2\Delta x}(2n+1)(\Delta t)^2 &\leq \frac{3\pi}{2\Delta x}(2N-1)(\Delta t)^2 \\ &< \frac{3\pi}{2\Delta x}(2N)(\Delta t)^2 = \frac{3\pi T}{\Delta x}\Delta t \end{aligned}$$

where T and N are the final time and total time steps respectively ($\Delta t = \frac{T}{N}$). To satisfy the condition $|\theta_i^n| \leq \frac{1}{2}$ we let $\Delta t \leq \frac{\Delta x}{6\pi T}$.

In our experiment we used $T = 1$ and $M = 40$. Hence, we need $\Delta t \leq 1.3263e-03$ i.e. $N \geq 754$ to ensure that the condition $|\theta_i^n| \leq \frac{1}{2}$ is satisfied for all n and all i. It was shown that for $r_\kappa \geq \frac{1}{4}$ no restriction is needed for θ_i^n and the scheme is stable. We can see this in Figure 4.6 where $r_\kappa = 0.249$ (here we used $\kappa = 0.1178$). However, the scheme is supposed to have instability when $r_\kappa < \frac{1}{4}$ if we keep $|\theta_i^n| \leq \frac{1}{2}$. It was shown that to have stability in this case we let $|\theta_i^n| \in (0,\alpha)$ where $\alpha \leq \frac{\frac{1}{2}+2r_\kappa}{2} \leq \frac{1}{2}$. With the condition $|\theta_i^n| \leq \frac{1}{2}$, the first instability appears when $r_\kappa = 0.16$ (here we used $\kappa = 0.0754$) Figure 4.7. With this value of r_κ, we should have $|\theta_i^n| \in (0, 0.41)$ which indicates that $N \geq 920$ should be used. However, the numerical results show some negative values when $N < 950$ so we need to use $N \geq 950$. We are keeping $r_\kappa = 0.16$ fixed by choosing $\kappa = 0.095$ and N = 950. Figure 4.8 shows that the instability in the previous graph was removed completely. Therefore the stability was recovered by using smaller time steps i.e smaller bound for $|\theta_i^n|$.

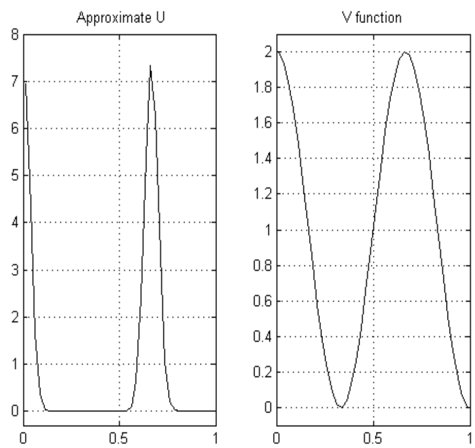

Figure 4.6 $N = 754, M = 40,\ r_\kappa = 0.249.$

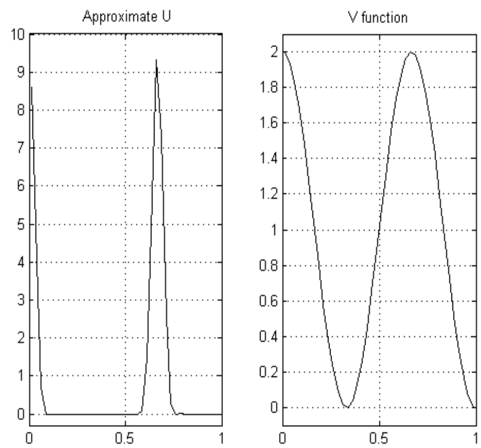

Figure 4.7 $N = 754, M = 40,\ r_\kappa = 0.16.$

4.1.3 A numerical experiment for convergence of the 1D scheme

This part is devoted to confirm the order of the accuracy of the scheme (2.20). We consider the system

$$u_t + (uv_x)_x = \kappa u_{xx} + g(x,t), \qquad 0 < x < 1, \qquad t > 0 \tag{4.1}$$

$$v_t = \sigma v_{xx} - \lambda v + f(x,t), \qquad 0 < x < 1, \qquad t > 0 \tag{4.2}$$

$$u(x,0) = 1, \quad v(x,0) = 0 \qquad in\ \Omega \tag{4.3}$$

$$\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = 0 \qquad on\ \partial\Omega. \tag{4.4}$$

The constants used in the simulation were $\kappa = 1, \sigma = 1$, and $\lambda = 5$. The source terms are

$$f(x) = 20 + 20\cos(6\pi x),$$

$$g(x,t) = e^v(wv_t - (v_x w_x + 5w) + 5 + 2\cos(4\pi x))$$

where

$$w = w(x,t) = 1 + \tfrac{2}{5+16\pi^2}(1 - e^{-(5+16\pi^2)t})\cos(4\pi x).$$

and $v = v(x,t)$ is the exact solution of the second equation (4.2) given by

$$v(x,t) = 4(1 - e^{-5t}) + \frac{20}{5+36\pi^2}(1 - e^{(-5+36\pi^2)t})\cos(6\pi x).$$

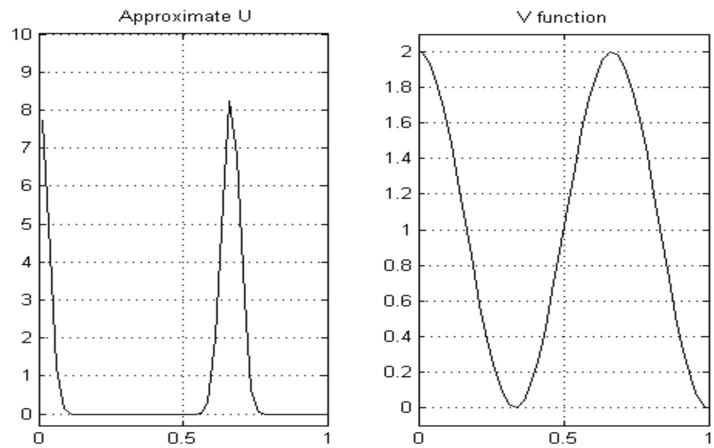

Figure 4.8 $N = 950, M = 40,\ r_\kappa = 0.16.$

Equation (4.1) has exact solution $u(x, t) = e^v w$.

Table 4.1 shows the rate of convergence which is second order as we proved analytically. The parameters which are used in the table are defined as follows. The first column k represent the doubling of the time and space partitioning while the second and third columns (N and M) represent the time and space subdivisions respectively. The errors in the chemoattractant concentration ,$v(x, t)$ and the cell density,$u(x, t)$, are given by the parameters VE_k and UE_k respectively. The errors were computed in the final time $T = 1$ using the sup-norm $|| \cdot ||_\infty$. Finally, the convergence rates are given by the parameters VE_k/VE_{k+1} and UE_k/UE_{k+1}. Figures 4.9 and 4.10 show the response of the cell density $u(x, t)$ to the chemoattractant $v(x, t)$.

Table 4.1 Rates of Convergence for 1D Chemotaxis System

k	N	M	VE_k	VE_k/VE_{k+1}	UE_k	UE/UE_{k+1}
1	100	20	4.21e-03	0.00e+00	2.30e-01	0.00e+00
2	200	40	1.03e-03	4.10e+00	5.97e-02	3.86e+00
3	400	80	2.55e-04	4.02e+00	1.51e-02	3.95e+00

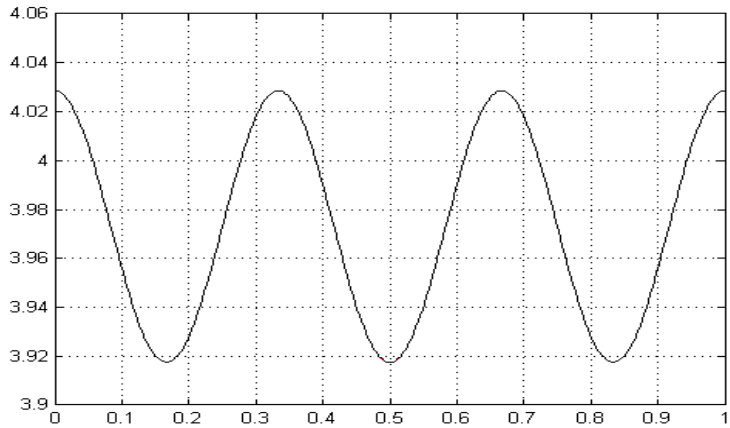

Figure 4.9 1D Chemoattractant $v(x,t)$

4.2 Numerical experiments for the 2D chemotaxis system

4.2.1 Positivity preserving property

The scheme (3.37) being so complicated makes it challenging to prove the positivity preserving property. Therefore, we study this property numerically. The method will be applied on the equation

$$u_t + \nabla(u\nabla v) = \nabla \cdot (\kappa \nabla u), \qquad (x,y) \in \Omega = (0,1)\times(0,1), \qquad t \geq 0$$
$$\frac{\partial u}{\partial n} = 0, \qquad (x,y) \in \partial\Omega$$

with a function $v(x,y,t)$ known in advance. It was shown that when r_κ is large enough then the method is stable. Therefore, we will be testing the method with $\kappa = 0.5$ so that we don't need to worry about any restrictions on $|\theta^{x,n}_{i,j}|$ and $|\theta^{y,n}_{i,j}|$. In the following first three cases we used M = 20, N = 7500 and final time $T = 30$. With these values, it is guaranteed that the conditions $|\theta^{x,n}_{i,j}| \leq \frac{1}{2}$ and $|\theta^{y,n}_{i,j}| \leq \frac{1}{2}$ are satisfied.

Case 1: Mass moving away from the centre

Let

$$v(x,y,t) = \frac{2t}{t+1}\cos(2x(x-1)\pi)\cos(2y(y-1)\pi).$$

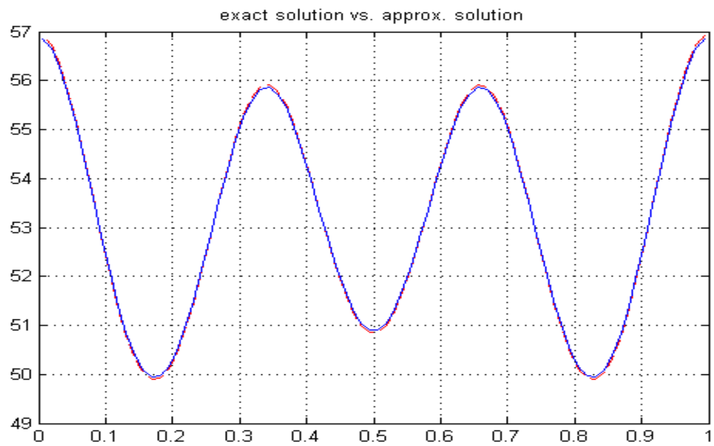

Figure 4.10 1D Cell density $u(x, t)$

In this case we let the chemical substance move to the corners of the domain and, hence, the cell density $u(x, y, t)$ will be leaving the centre (Figure 4.11 and 4.12). In Figure 4.12, $\kappa = 0.2$ was used.

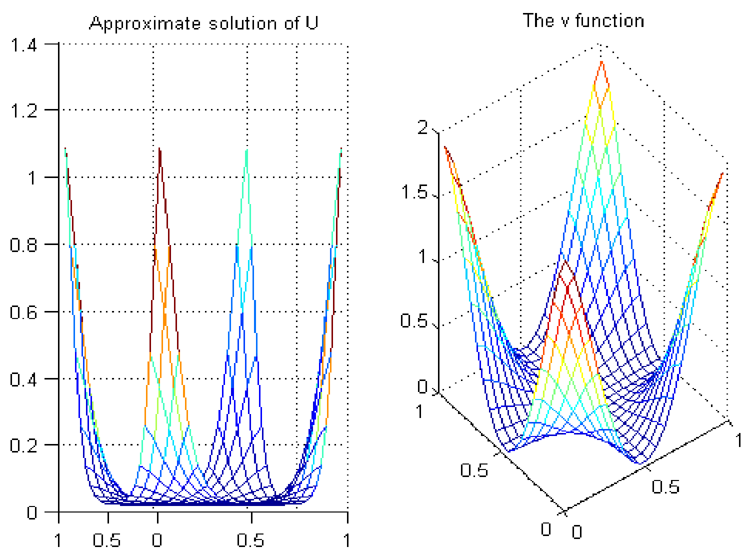

Figure 4.11 Final time $T = 30$, initial condition $u(x, y, 0) = 0.1$, $r_\kappa = 0.8$

Case 2: Mass moving to one half of the domain

In this case we will test the method when the chemoattractant, $v(x, y, t)$, is concentrated

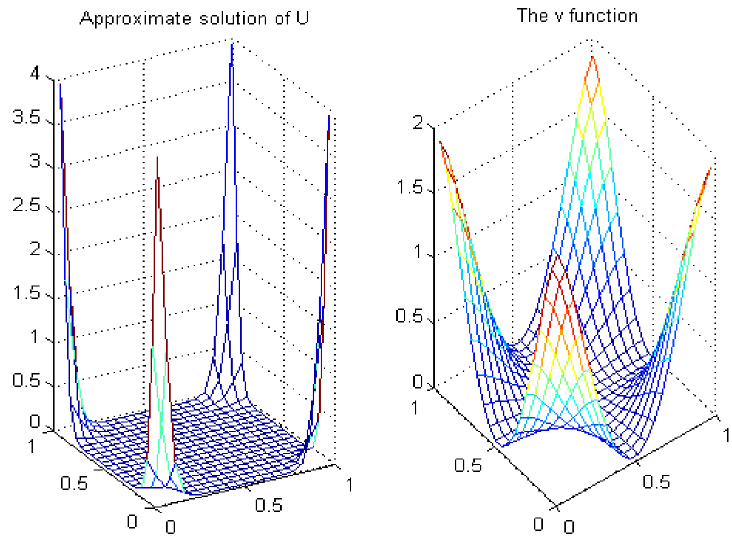

Figure 4.12 Final time $T = 30$, initial condition $u(x, y, 0) = 0.1$, $r_\kappa = 0.32$

in one half of the domain. Let

$$v(x, y, t) = \begin{cases} 0 \quad x \leq 0.5 \\ \frac{2t}{t+1} \cos(2x(x-1)\pi) \cos(2y(y-1)\pi), \quad \text{elsewhere}, \end{cases}$$

then the mass is moving to the half of the domain as a response to the chemical substances. Figure 4.13 shows the sectional view of this response.

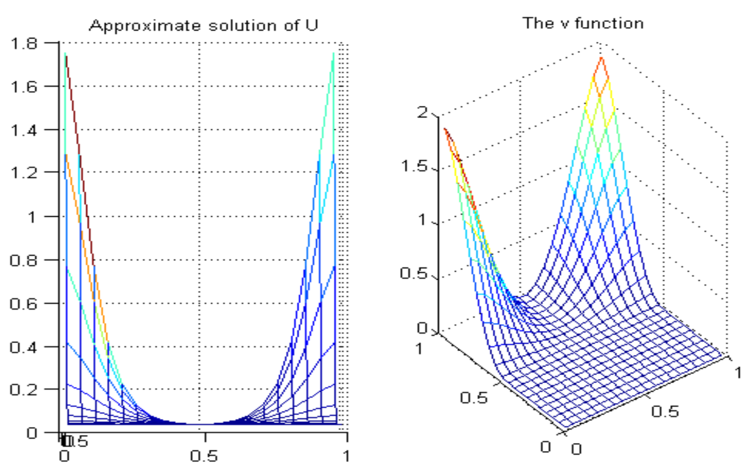

Figure 4.13 Final time $T = 30$, initial condition $u(x, y, 0) = 0.1$, $r_\kappa = 0.8$

Case 3: Mass moving to one corner

Let

$$v(x,y,t) = \begin{cases} 0 \quad x \le 0.5, \quad y \le 0.5 \\ \frac{2t}{t+1}\cos(2x(x-1)\pi)\cos(2y(y-1)\pi), \quad \text{elsewhere.} \end{cases}$$

This function describes the case when the chemical substances move toward one corner of the domain. The result of this case is shown by Figure 4.14.

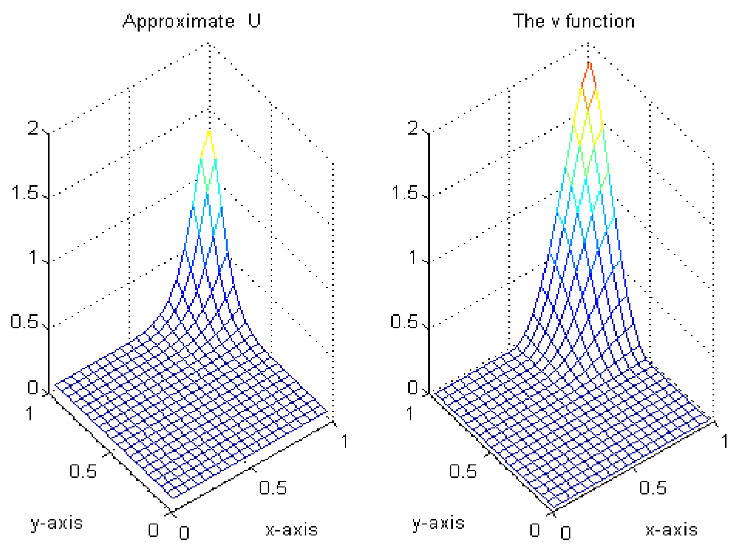

Figure 4.14 Final time $T = 30$, initial condition $u(x, y, 0) = 0.1$, $r_\kappa = 0.8$

Case 4: Mass clustering at the centre

The last case in our experiment will be when the mass is clustering in the middle of the domain. Let $v(x, y, t)$ be given by

$$v(x,y,t) = \begin{cases} 0 \quad x,y \le 0.3, \\ \frac{t}{t+1}(1-\cos(2\pi\frac{(x-0.3)(x-0.7)}{0.4^2}))(1-\cos(2\pi\frac{(y-0.3)(y-0.7)}{0.4^2})), \quad 0.3 \le x,y \le 0.7, \\ 0 \quad x,y \ge 0.7 \end{cases}$$

In this case, we are keeping $M = 20$ and $T = 30$ but letting $N = 30000$ so that the conditions $|\theta^{x,n}_{i,j}| \le 0.5$ and $|\theta^{y,n}_{i,j}| \le 0.5$ are satisfied for all i, j and all n. Figure 4.15 shows that the mass is moving toward the centre and making a pile.

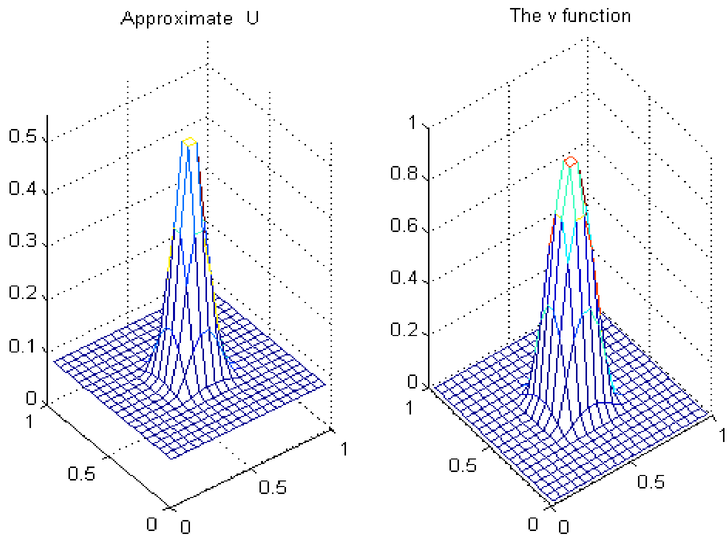

Figure 4.15 Final time $T = 30$, initial condition $u(x, y, 0) = 0.1$, $r_\kappa = 0.2$

Conclusion: Since none of these results gave negative values, we can say that the scheme (3.37) might be positivity preserving.

4.2.2 Studying the stability numerically

We devote this section to study the stability of the scheme (3.37) numerically. It was shown in Section 3.3 that the scheme is conditionally stable. We confirm this fact by applying the scheme on the equation

$$\begin{aligned} u_t + \nabla \cdot (u\nabla v) &= \nabla \cdot (\kappa \nabla u), && (x, y) \in \Omega, t > 0 \\ \frac{\partial u}{\partial n} &= 0, && (x, y) \in \partial\Omega \\ u(x, y, 0) &= 1 \end{aligned}$$

where $v(x, y, t)$ is given by

$$v(x, y, t) = 2t(1 + \cos(3x\pi)\cos(3y\pi)).$$

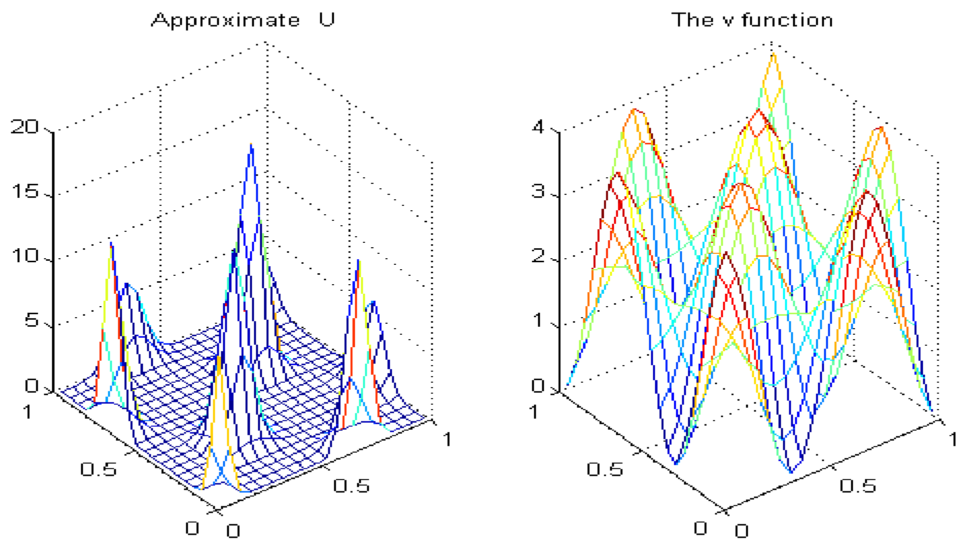

Figure 4.16 Final time $T = 1, N = 754$ and $r_\kappa = 0.2122$

It is clear that this function satisfies the conditions $v(x, y, 0) = 0$ and $\dfrac{\partial v}{\partial n} = 0$ on $\partial\Omega$. The parameters $\theta_{i,j}^{x,n}$ and $\theta_{i,j}^{y,n}$ can be found by solving the following differential equations

$$\begin{bmatrix} x \\ y \end{bmatrix}' = \begin{bmatrix} v_x(x(t), y(t), t) \\ v_y(x(t), y(t), t) \end{bmatrix}, \qquad \begin{bmatrix} x \\ y \end{bmatrix}(t_{n+1}) = \begin{bmatrix} x_i \\ y_j \end{bmatrix}.$$

Hence, we have

$$\begin{bmatrix} \theta_{i,j}^{x,n} \\ \theta_{i,j}^{y,n} \end{bmatrix} \approx \begin{bmatrix} \frac{\Delta t}{2\Delta x}[v_x(x, y, t_n) + v_x(x, y, t_{n+1})] \\ \frac{\Delta t}{2\Delta y}[v_y(x, y, t_n) + v_y(x, y, t_{n+1})] \end{bmatrix},$$

and similarly

$$\begin{bmatrix} \theta_{i,j}^{x,n+1} \\ \theta_{i,j}^{y,n+1} \end{bmatrix} \approx - \begin{bmatrix} \frac{\Delta t}{2\Delta x}[v_x(x, y, t_n) + v_x(x, y, t_{n+1})] \\ \frac{\Delta t}{2\Delta y}[v_y(x, y, t_n) + v_y(x, y, t_{n+1})] \end{bmatrix}.$$

where

$$v_x(x, y, t) = -6\pi t \sin(3\pi x)\cos(3\pi y), \qquad v_y(x, y, t) = -6\pi t \cos(3\pi x)\sin(3\pi y).$$

Therefore, we get that

$$|\theta^{x,n+1}_{i,j}| \le \frac{3\pi\Delta t}{\Delta x}(t_{n+1}+t_n) = \frac{3\pi}{\Delta x}(2n+1)(\Delta t)^2$$
$$< \frac{3\pi}{\Delta x}(2N-1)(\Delta t)^2,$$

where N is the total time steps $\Delta t = \frac{T}{N}$. Similarly, we have

$$|\theta^{y,n+1}_{i,j}| < \frac{3\pi}{\Delta y}(2N-1)(\Delta t)^2.$$

Notice that $\frac{3\pi}{\Delta x}(2N-1)(\Delta t)^2 < \frac{3\pi}{\Delta x}(2N)(\Delta t)^2 = \frac{6\pi T}{\Delta x}\Delta t$. So to satisfy the conditions $|\theta^{x,n+1}_{i,j}| \le \frac{1}{2}$ and $|\theta^{y,n+1}_{i,j}| \le \frac{1}{2}$ we let $\Delta t \le \frac{\Delta x}{12\pi T}$. In our experiment, we choose $M = 20$ and $T = 1$. Thus we need $\Delta t \le 0.001326$ (i,e $N = 754$).

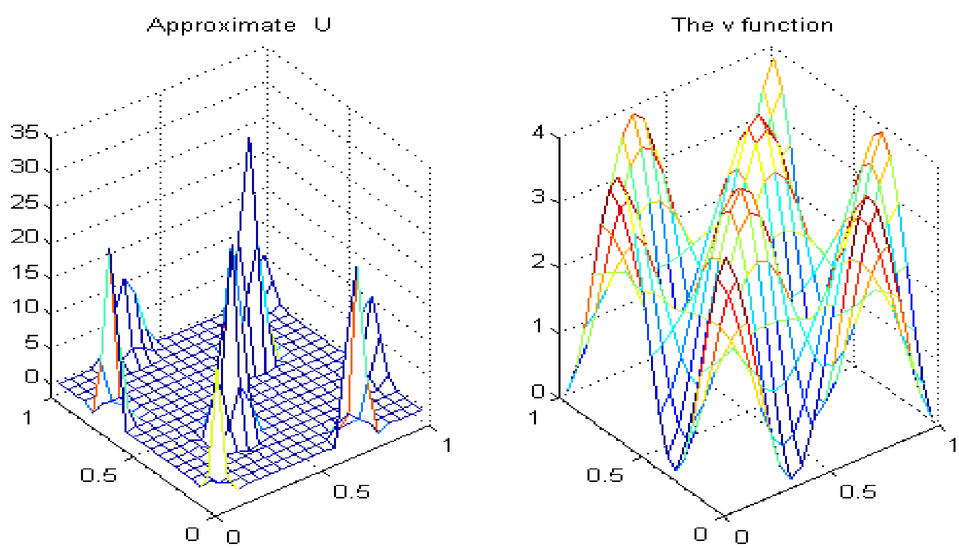

Figure 4.17 Final time $T = 1, N = 800$ and $r_\kappa = 0.12$

Our numerical results show that the scheme under the conditions $|\theta^{x,n}_{i,j}| \le \frac{1}{2}$ and $|\theta^{x,n}_{i,j}| \le \frac{1}{2}$ is stable when $r_\kappa \ge 0.21$ (Figure 4.16 in which we used $\kappa = 0.4$).Under the assumption that the scheme is positivity preserving, we consider any negative value shown in the results is a sign of non-stability. The first negative values appear when $r_\kappa = 0.2$ (here $\kappa = 0.377$). In order to remove these negative values we shrink Δt. Thus, we use N $= 800$ and let $r_\kappa = 0.2$ by choosing $\kappa = 0.4$. Hence we have $|\theta^{x,n}_{i,j}|$, $|\theta^{y,n}_{i,j}| \in (0, 0.471)$.

Figure 4.17 shows instability in the method when $r_\kappa = 0.12$ ($\kappa = 0.244$) and N = 800 were used. To recover the stability we let N = 1350 while keeping $r_\kappa = 0.12$ by choosing $\kappa = 0.40586$ (Figure 4.18). In this case we have $|\theta^{x,n}_{i,j}|$, $|\theta^{y,n}_{i,j}| \in (0, 0.279)$.

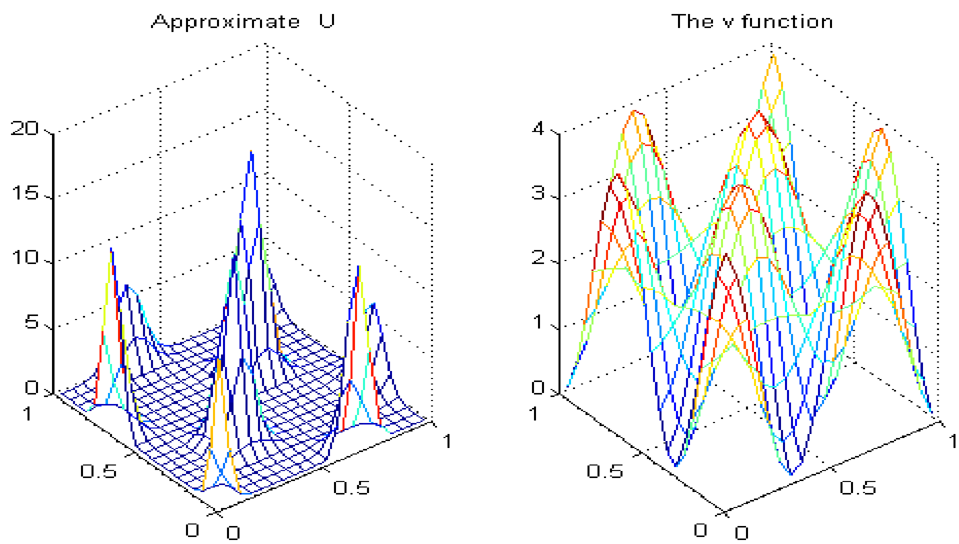

Figure 4.18 Final time $T = 1, N = 1350$ and $r_\kappa = 0.12$

4.2.3 A Numerical experiment for convergence of the 2D scheme

As a simple illustration of the scheme (3.37) we consider the system

$$u_t + \nabla \cdot (u \nabla v) = \nabla \cdot (\kappa \nabla u) + g(u, v, x, y, t), \tag{4.5}$$

$$v_t = \nabla \cdot (\sigma \nabla v) - \lambda v + f(u, v, x, y, t) \qquad (x, y) \in \Omega, t > 0 \tag{4.6}$$

$$\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = 0, \qquad (x, y) \in \partial\Omega, t > 0, \tag{4.7}$$

$$u(x, y, 0) = 1, \qquad v(x, y, 0) = 0 \tag{4.8}$$

where $\Omega = (0, 1) \times (0, 1)$ is a rectangle in the plane. The constants used in the simulation are $\kappa = 5$, $\sigma = 0.1$ and $\lambda = 5$. The source terms are given by

$$f(x, y) = 20 + 10 \cos(3\pi x) \cos(2\pi y),$$

$$g(x, y, t) = \exp(v)(v_t + (1 - \kappa)((v_x)^2 + (v_y)^2 + v_{xx} + v_{yy})),$$

where $v = v(x, y, t)$ is the exact solution of (4.6) given by

$$v(x,y,t) = \frac{20}{\lambda}(1-\exp(-\lambda t)) + \frac{10}{13\sigma\pi^2+\lambda}(1-\exp(-(13\sigma\pi^2+\lambda)t))\cos(3\pi x)\cos(2\pi y).$$

The exact solution of (4.5) is given by $u(x, y, t) = e^v$. In this experiment we used the alternating direction implicit method (ADI) to find the approximate solution of $v(x, y, t)$. The method is given by

$$\begin{aligned}\left(1-\frac{r_{\kappa,x}}{2}\delta_x^2\right)v_{j,k}^{n+\frac{1}{2}} &= \left(1+\frac{r_{\kappa,y}}{2}\delta_x^2\right)v_{j,k}^{n} + \frac{\Delta t}{2}F_{j,k}^{n}\\ \left(1-\frac{r_{\kappa,y}}{2}\delta_y^2\right)v_{j,k}^{n+1} &= \left(1+\frac{r_{\kappa,x}}{2}\delta_x^2\right)v_{j,k}^{n+\frac{1}{2}} + \frac{\Delta t}{2}F_{j,k}^{n+1},\end{aligned}$$

where $F = F(x, y, v, t) = -\lambda v + f(u, v, x, y, t)$.

Table 4.2 shows second order accuracy as it was proved analytically. The parameters which are used in the table are defined as follows. The first column k represent the doubling of the time and space partitioning whereas the second and third columns (N and M) represent the time and space subdivisions respectively. The errors in the chemoattractant concentration ,$v(x, y, t)$ and the cell density,$u(x, y, t)$, are given by the parameters VE_k and UE_k respectively and their relative errors are given by $VRel$ and $URel$. The errors were computed in the final time $T = 1$ using the Frobenius norm $||\cdot||_{frob}$. Finally, the convergence rates are given by the parameters VE_k/VE_{k+1} and UE_k/UE_{k+1}. Figures 4.19 and 4.20 show response of the cell density $u(x, y, t)$ to the chemoattractant $v(x, y, t)$. It is clear from the graphs that the cells move toward the high concentration areas.

Table 4.2 Rates of Convergence for 2D Chemotaxis System

k	N	M	VE_k	VE_k/VE_{k+1}	$VRel$	UE_k	UE/UE_{k+1}	$URel$
1	50	10	1.26e-02	0.00e+00	3.16e-04	2.46e+00	0.00e+00	4.23e-03
2	100	20	3.11e-03	4.05e+00	3.90e-05	5.64e-01	4.36e+00	4.90e-04
3	200	40	7.75e-04	4.01e+00	4.86e-06	1.38e-01	4.08e+00	6.01e-05

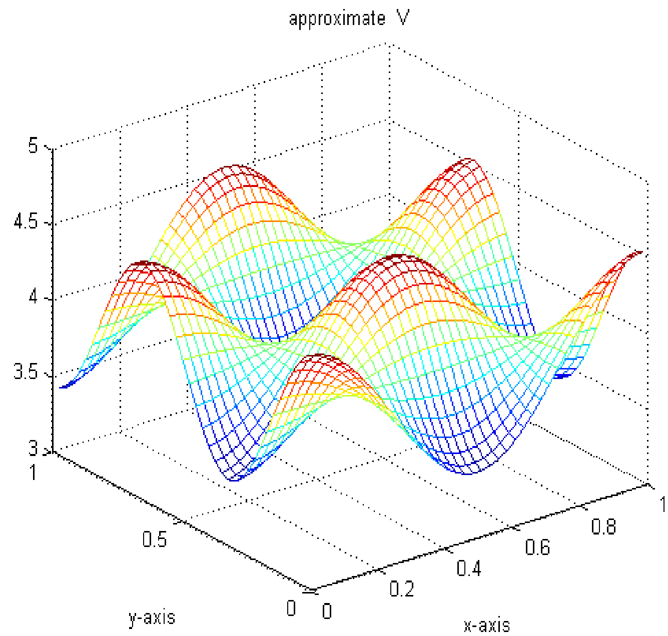

Figure 4.19 2D Chemoattractant

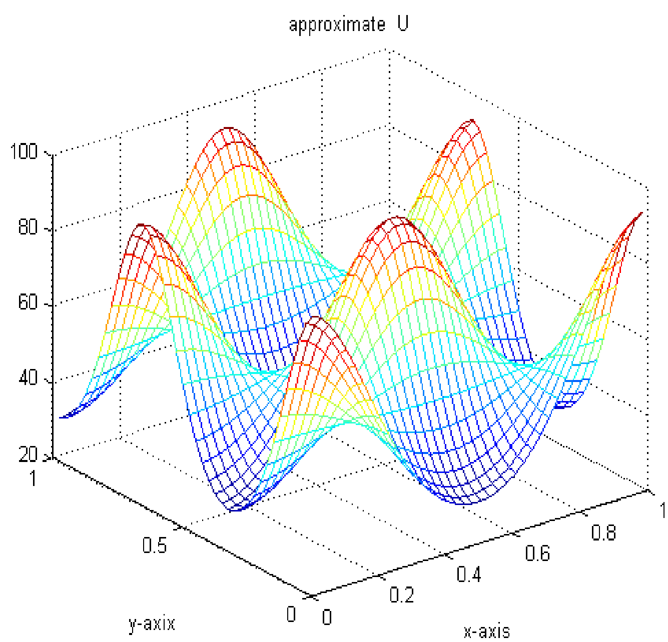

Figure 4.20 2D Cell density

CHAPTER 5. SUMMARY AND DISCUSSION

In this thesis we succeeded in deriving second order finite volume schemes. The order of the accuracy was proved analytically in both cases (i.e 1D and 2D). Attaching to this analysis, numerical experiments were given to support the results. It was shown that the method is mass conservative. However, more work is needed to prove that the method preserves the non-negative property. We have not been able to prove this property and it is not clear whether is satisfied by the method or not. Therefore, some numerical experiments, with $v(x, y, t)$ known in advance, were made to test the scheme. The numerical results never gave any negative values which indicate that the method might satisfy the property.

One drawback of the method is that it is very complicated which makes it hard to implement and harder to analyse. Also the coefficients in the method were changing with time. This problem in the method raised some difficulties in proving the stability and hence very restrictive conditions were induced. As a consequence of this restriction on the stability, a desirable feature, which is using large time steps, of numerical methods was not possible to be used. Numerical tests were given in the context of studying the stability of the methods.

In the discussion of consistency of the 2D scheme, it was shown that the diffusion flux terms are higher order terms. Therefore, these terms might be removed from the scheme without affecting the accuracy. This helps in reducing the scheme and simplifies things.

Because of the convective flux terms, the linear systems obtained from the methods in both cases are non-symmetric. The matrix in the 1D case is tridiagonal whereas it is block tridiagonal matrix in the 2D case. In both cases, the matrices are not fixed constants where they are needed to be updated for each time step. This affects the efficiency of the schemes since more computation effort should be made in each time step. Since the systems are sparse, some ideas of sparse matrices (e.g. sparse storage) may be applied. Even though we used a fixed time step in our work, it is possible to use different time steps depending on where and when the restrictions of the method are satisfied (i.e. $|\theta_{i,j}^{x,n}| \leq 0.5$ and $|\theta_{i,j}^{y,n}| \leq 0.5$).

Future work: It is straight forward to extended the method to three space dimensions but the scheme will become more complicated. This is might be done in the future. One thing that can be done to improve the method is the use of the idea of alternating direction method. Since the 1D scheme method is second order, it might be possible to apply the method on the two dimensional case by alternating the 1D scheme. First apply the method in the x-direction and then in the y-direction.

APPENDIX A. EIGENVALUES OF TRIDIAGONAL MATRICES

In this appendix we are going to calculate the eigenvalues of the matrices A and B given in Section 2.5. We will find the eigenvalues and eigenvectors of the matrix B. The eigenvalues of A can be found in the same way. Recall that the matrix B is given by

$$B = \begin{bmatrix} 7 & 1 & & & \\ 1 & 6 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 6 & 1 \\ & & & 1 & 7 \end{bmatrix}_{M\times M} .$$

We solve the eigenvalue problem

$$B_1 v = \lambda v,$$

where $\lambda \in R$ and $v = [v_1, ..., v_M]^T \neq 0$. From the eigenvalue problem we obtain the following difference equation

$$\begin{aligned} & v_{j-1} + 6v_j + v_{j+1} = \lambda v_j \qquad j = 1, ..., M \\ & v_0 = v_1, \qquad v_M = v_{M+1} \end{aligned}$$

which is equivalent to

$$\begin{aligned} & v_{j-1} + (6-\lambda)v_j + v_{j+1} = 0 \qquad j = 1, ..., M \\ & v_0 = v_1, \qquad v_M = v_{M+1} \end{aligned} \tag{A.1}$$

The solution of such equation can be written in terms of the roots of the characteristic polynomial, which in our case is given by

$$P(r) = r^2 + (6 - \lambda)r + 1.$$

Let r_1 and r_2 be the roots of $P(r)$, then the solution of (A.1) is given as

$$v_j = C_1 r_1^j + C_2 r_2^j, \qquad j = 0, 1, ..., M + 1.$$

By the boundary condition $v_0 = v_1$, we have

$$C_1 + C_2 = C_1 r_1 + C_2 r_2 \Rightarrow C_1(1 - r_1) + C_2(1 - r_2) = 0.$$

Since $v \neq 0$, then at least one of the constants C_1 and C_2 is not zero.The possible cases are

i) $C_1 = 0$ which leads to $r_2 = 1$

ii) $C_2 = 0$ that gives $r_1 = 1$,

iii) neither of the constants C_1 and C_2 is zero. In this case, there are two possibilities: $r_1 = r_2 = 1$ or neither of the roots is 1. If non of them is 1, then we have

$$C_2 = -C_1 \frac{1 - r_1}{1 - r_2}$$

which gives

$$v_j = C_1 \big(r_1^j - \frac{1 - r_1}{1 - r_2} r_2^j \big).$$

Furthermore we have $v_M = v_{M+1}$ which leads to

$$C_1 \left(r_1^M - \frac{1 - r_1}{1 - r_2} r_2^M \right) = C_1 \left(r_1^{M+1} - \frac{1 - r_1}{1 - r_2} r_2^{M+1} \right)$$

$$\Rightarrow$$

$$C_1(1 - r_1)(r_1^M - r_2^M) = 0 \Rightarrow (\frac{r_1}{r_2})^M = 1.$$

From the identity

$$\begin{aligned} r_1 r_2 &= \left(\frac{(6 - \lambda) + \sqrt{(6 - \lambda)^2 - 4}}{2} \right) \left(\frac{(6 - \lambda) - \sqrt{(6 - \lambda)^2 - 4}}{2} \right) \\ &= \frac{(6 - \lambda)^2 - (6 - \lambda)^2 + 4}{4} = 1, \end{aligned}$$

we get that

$$(\frac{r_1}{r_2})^M = (r_1^2)^M = 1.$$

Since the roots of a quadratic polynomial are in general complex, the above equation can be written in the form

$$r_1^2 = e^{2\pi i(\frac{k}{M})}, \qquad k = 1, ..., M-1.$$

It is easy to see that the possible roots are

$$\begin{aligned} r_{1,k} &= e^{\pi i(\frac{k}{M})}, \\ r_{2,k} &= e^{-\pi i(\frac{k}{M})}, \qquad k = 1, ..., M-1. \end{aligned}$$

For every $k = 1, ..., M-1$ there is an eigenvalue λ_k given by the equation

$$\lambda_k - 6 = r_{1,k} + r_{2,k} = e^{\pi i(\frac{k}{M})} + e^{-\pi i(\frac{k}{M})} = 2\cos(\pi(\tfrac{k}{M})).$$

If $r_1 = r_2 = 1$ then $\lambda - 6 = 2 \Rightarrow \lambda = 8$ which can be obtained form the above equation by letting $k = 0$. Therefore, the eigenvalues of the matrix B are given by

$$\lambda_k = 6 + 2\cos(\pi(\tfrac{k}{M})), \qquad k = 0, 1, ..., M-1.$$

The associated eigenvectors $v_{k,j}$ ($k = 1, ..., M-1$) are given by

$$\begin{aligned} v_{k,j} &= C_1\left(r_{1,k}^j - (\frac{1-r_{1,k}}{1-r_{2,k}})r_{2,k}^j\right) = C_1\left(r_{1,k}^j - (\frac{1-r_{1,k}}{r_{1,k}-1})r_{1,k}r_{2,k}^j\right) = C_1(r_{i,k}^j + r_{2,k}^{j-1}) \\ &= C_1\left(e^{i(\frac{jk\pi}{M})} + e^{-i(\frac{(j-1)k\pi}{M})}\right) \\ &= C1\left[\left(\cos(\tfrac{jk\pi}{M}) + \cos\left(\tfrac{(j-1)k\pi}{M}\right)\right) + i\left(\sin\left(\tfrac{jk\pi}{M}\right) - \sin\left(\tfrac{(j-1)k\pi}{M}\right)\right)\right] \\ &= 2C_1\left[\cos\left(\tfrac{(j-\frac{1}{2})k\pi}{M}\right)\cos\left(\tfrac{\frac{1}{2}k\pi}{M}\right) + i\cos\left(\tfrac{(j-\frac{1}{2})k\pi}{M}\right)\sin\left(\frac{\frac{1}{2}k\pi}{M}\right)\right] \\ &= 2C_1\cos\left(\frac{(j-\frac{1}{2})k\pi}{M}\right)\left[\cos\left(\tfrac{\frac{1}{2}k\pi}{M}\right) + i\sin\left(\tfrac{\frac{1}{2}k\pi}{M}\right)\right]. \end{aligned}$$

By choosing $2C_1 = \left[\cos(\frac{\frac{1}{2}k\pi}{M}) + i\sin(\frac{\frac{1}{2}k\pi}{M})\right]^{-1}$ we get $v_{k,j} = \cos\left(\frac{(j-\frac{1}{2})k\pi}{M}\right)$. When $k = 0$, then $v_{0,j} = 1$ for all j.

APPENDIX B. MATLAB CODES

The following MATLAB programs are used to generate the data needed to do the numerical experiments given in Chapter 4. The main code in 1d system is *Chemotax1d.m*. This code and its related source functions *testFfcn.m* and *testGfcn.m* were written by Smiley. The main code in 2D case is *Chemotax2d.m* and the source functions are given in the codes *test2DFfn.m* and *test2DGfn.m* The code *ExactV.m* returns the exact solution $v(x, y, t)$.

Chemotax1d

```
% Script File : Order 2 1d Model System Convergence
% The model system parameters
xi = 1;
kappa = 5;
sigma = 1;
lambda = 5;
% Parameters determining the source terms and known solutions
% of a test model system
f0 = 20; fn = 20; nn = 6;
h0 = 5; hm = 2; mm = 4; gamma4w = 5;
parm4f = [f0,fn,nn];
parm4v = [f0,fn,nn,lambda,sigma];
parm4w = [h0,hm,mm,gamma4w,kappa];
```

```
% set final time and print table heading
t_final = 1.0;                     % pre-chosen final time
disp('  ')
disp('   A Table Verifying Rates of Convergence ')
disp('  ')
disp('  k   N   M     VE_k    VE_k/VE_(k+1) UE_k    UE_k/UE_(k+1)')
disp('-------------------------------------------------------------')
M = 20;        % M = number of subdivisions in space
N = 50;        % N = number of time steps
nUerror = 0;  % n*Error = new error, set to 0 for looping
nVerror = 0;
for k = 1:3
  oUerror = nUerror;             % new error becomes old error
  oVerror = nVerror;             % new error becomes old error
  M = 2*M; dx = 1/M;           % set new M and new N
  N = 2*N; dt = t_final/N;
  x = linspace(0,1,M+1);       % set space grid
  xbar = 0.5*(x(2:M+1) + x(1:M));
  rk = kappa*dt/(dx*dx);    % u diffusion
  rs = sigma*dt/(dx*dx);    % v diffusion
  rt = dt/(dx*dx);          % theta multiplier
% factor matrix for V system
  Lv = zeros(1,M);     % lower sub-diagonal
  Dv = zeros(1,M);     % diagonal
  Uv = zeros(1,M);     % upper sub-diagonal
  a = -rs/2;                   % a(i) = a, all i
```

```
  b = 1 + rs + 0.5*lambda*dt;  % b(i) = b, 1 < i < M
  c = -rs/2;                   % c(i) = c, all i
% LU factorization
  Dv(1) = 1 + 0.5*rs + 0.5*lambda*dt;
  Uv(1) = c;
  for i = 2:(M-1)
  Lv(i) = a/Dv(i-1);
  Dv(i) = b - Lv(i)*c;
  Uv(i) = c;
  end
  Lv(M) = a/Dv(M-1);
  Dv(M) = (1 + 0.5*rs + 0.5*lambda*dt) - Lv(M)*c;
% factor matrix for U system
  Lu = zeros(1,M);     % lower sub-diagonal
  Du = zeros(1,M);     % diagonal
  Uu = zeros(1,M);     % upper sub-diagonal
  a = (1/8) - rk/2;                  % a(i) = a, all i
  b = (6/8) + rk;                  % b(i) = b, 1 < i < M
  c = (1/8) - rk/2;                  % c(i) = c, all i
% LU factorization
  Du(1) = (7/8) + 0.5*rk;
  Uu(1) = c;
  for i = 2:(M-1)
  Lu(i) = a/Du(i-1);
  Du(i) = b - Lu(i)*c;
  Uu(i) = c;
  end
```

```
  Lu(M) = a/Du(M-1);
  Du(M) = ((7/8) + 0.5*rk) - Lu(M)*c;
% Declare characteristic method variables
  theta = zeros(1,M-1);
  Amin = zeros(1,M-1);
  Amax = zeros(1,M-1);
  Flx = zeros(1,M-1);
% variables used in solving Lw = b, Uz = w so LUz = b
  w = zeros(1,M);
  z = zeros(1,M);
% variables used in setting up the explicit terms in the scheme
  rhs = zeros(1,M);     % right hand side made up from
  del = zeros(1,M-1);   % differences of v,u
  diff = zeros(1,M);    % diffusion terms
  oterms = zeros(1,M);  % other terms
% variables used in setting up the system matrix
  A = zeros(1,M);
  B = zeros(1,M);
  C = zeros(1,M);

% The initialization
  t = 0;
  u = ones(1,M);      % u = u(x,t_{n+1})
  u0 = ones(1,M);     % u0 = u(x,t_n)
  v = zeros(1,M);     % v = v(x,t_{n+1})
  v0 = zeros(1,M);    % v0 = v(x,t_n)
```

```
%------------ Simulation loops ------------------------

  for j = 1:N

    t0 = t;    % variables at t = t_{n+1}

    v0 = v;    % become variable at t = t_n

    u0 = u;

    t = t0 + dt;

  % first solve for v^{n+1}

    del = v0(2:M) - v0(1:M-1);

    diff(1) = rs*del(1);

    diff(2:M-1) = rs*(del(2:M-1) - del(1:M-2));

    diff(M) = -rs*del(M-1);

    oterms = testFfcn(xbar,t0,parm4f) + testFfcn(xbar,t,parm4f);

    oterms = dt*(oterms - lambda*v0);

    rhs = v0 + 0.5*(diff + oterms);

    w(1) = rhs(1);                    % solve Lw = rhs

    for i = 2:M

      w(i) = rhs(i) - Lv(i)*w(i-1);

    end

    z(M) = w(M)/ Dv(M);               % solve Uz = w

    for i = M-1:-1:1

      z(i) = (w(i) - Uv(i)*z(i+1))/Dv(i);

    end

    v = z;     % now v = v^{n+1}

  % now solve for u^{n+1}

    theta = -0.5*rt*xi*((v(2:M)-v(1:M-1)) + (v0(2:M)-v0(1:M-1)));

    Amin = theta.*(1 - theta);

    Amax = theta.*(1 + theta);
```

```
  Flx = 0.5*(Amin.*u0(1:M-1) + Amax.*u0(2:M));
  A(2:M) = (1/8) - 0.5*rk + 0.25*Amax(1:M-1);
  B(1) = (7/8) + 0.5*rk - 0.25*Amax(1);
  B(2:M-1) = (6/8) + rk + 0.25*(Amin(1:M-2) - Amax(2:M-1));
  B(M) = (7/8) + 0.5*rk + 0.25*Amin(M-1);
  C(1:M-1) = (1/8) - (0.5*rk + 0.25*Amin(1:M-1));
% LU factorization
  Du(1) = B(1);
  for i = 2:M
    Lu(i) = A(i)/Du(i-1);
    Du(i) = B(i) - Lu(i)*C(i-1);
  end
  Uu = C;
  del = u0(2:M) - u0(1:M-1);
  diff(1) = rk*del(1);
  diff(2:M-1) = rk*(del(2:M-1) - del(1:M-2));
  diff(M) = -rk*del(M-1);
  oterms = dt*(testGfcn(xbar,t0,parm4v,parm4w) ...
              + testGfcn(xbar,t,parm4v,parm4w));
  rhs(1) = 0.125*(7*u0(1) + u0(2));
  rhs(1) = rhs(1) + 0.5*(Flx(1) + diff(1) + oterms(1));
  rhs(2:M-1) = 0.125*(u0(1:M-2) + 6*u0(2:M-1) + u0(3:M));
  rhs(2:M-1) = rhs(2:M-1) + 0.5*(Flx(2:M-1) - Flx(1:M-2));
  rhs(2:M-1) = rhs(2:M-1) + 0.5*(diff(2:M-1) + oterms(2:M-1));
  rhs(M) = 0.125*(u0(M-1) + 7*u0(M));
  rhs(M) = rhs(M) + 0.5*(diff(M) + oterms(M) - Flx(M-1));
  w(1) = rhs(1);                    % solve Lw = rhs
```

```
    for i = 2:M
      w(i) = rhs(i) - Lu(i)*w(i-1);
    end
    z(M) = w(M)/ Du(M);                 % solve Uz = w
    for i = M-1:-1:1
      z(i) = (w(i) - Uu(i)*z(i+1))/Du(i);
    end
    u = z;                              % now u = u^{n+1}
  end
  exactv = testVsoln(xbar,t,parm4v);
  exactu = testUsoln(xbar,t,parm4v,parm4w);
  nVerror = max(abs(v - exactv));
  nUerror = max(abs(u - exactu));
  disp(sprintf(' %1d %5d %4d  %4.2e   %4.2e %4.2e   %4.2e '...
                 ,k,N,M,nVerror,oVerror/nVerror,nUerror,oUerror/nUerror))
end
  exactv = testVsoln(xbar,t,parm4v);
  plot(xbar,v,'r',xbar,exactv,'k-.')
  exactu = testUsoln(xbar,t,parm4v,parm4w);
  figure
  plot(xbar,u,'b')
 % plot(xbar,exactu,'g-.')
  pause(1)
```

testFfcn

```
% This function defines the source term f(x,t) in the
% model system equation for v(x,t) with known solutions
```

```
function f = testFfcn(xbar,t,parm4F)
f0 = parm4F(1); fn = parm4F(2); n = parm4F(3);
f = f0 + fn*cos(n*pi*xbar);
```

testGfcn

```
% This function defines the source term g(x,t) in the
% model system equation for u(x,t) with known solutions
function g = testGfcn(xbar,t,parm4V,parm4W)
f0 = parm4V(1); fn = parm4V(2); n = parm4V(3);
lambda = parm4V(4); sigma = parm4V(5);
a0 = lambda;
an = lambda + sigma*n*n*pi*pi;
v = (f0/a0)*(1 - exp(-a0*t));
v = v + (fn/an)*(1 - exp(-an*t))*cos(n*pi*xbar);
h0 = parm4W(1); hm = parm4W(2); m = parm4W(3);
gamma = parm4W(4); kappa = parm4W(5);
b0 = gamma;
bm = gamma + kappa*m*m*pi*pi;
w = (h0/b0) + (1 - (h0/b0))*exp(-b0*t);
w = w + (hm/bm)*(1 - exp(-bm*t))*cos(m*pi*xbar);
h = h0 + hm*cos(m*pi*xbar);
vt = f0*exp(-a0*t) + fn*exp(-an*t)*cos(n*pi*xbar);
vx = -n*pi*(fn/an)*(1 - exp(-an*t))*sin(n*pi*xbar);
wx = -m*pi*(hm/bm)*(1 - exp(-bm*t))*sin(m*pi*xbar);
g = exp(v/kappa).*( w.*(vt/kappa) + h - (vx.*wx + gamma*w));
```

(Chemotax2d)

```
% This script solves 2D chemotaxis system
% first solves the equation
% v_t = sigma*laplace v -lambda*V + f(x,y,t) using ADI method
% where f(x,y,t) is given in the function test2DFfcn
% second solves the equation
% u_t + div(u*div(v)) = kappa*laplace u + G(x,y,t)
% the outputs: A Table showing the rate of convergence , graphs of the
%             exact and approximate solutions

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% The model system parameters
sigma = 0.1;
lambda = 5;
kappa = 5;
% Parameters determining the source terms and known solutions
% of a test model system
 nn = 3;    mm = 2;  a1 = 2;
 f0 = 20;    fn = 10;
parm4f = [lambda, sigma,nn, mm,f0, fn];
parm4v = parm4f;
t_final = 1.0;                       % pre-chosen final time
disp('  ')
disp('   A Table showing the rate of convergence ')
disp('  ')
```

```
disp('  k   N   M     VE_{k}    V_rate      UE_{k}    U_rate')
disp('--------------------------------------------------------')
M = 5;        % M = number of subdivisions in space
N = 25;        % N = number of time steps
v_err(1)= 0 ;
U_error(1) = 0;
 for k =1:3
     M =2*M ; N = 2*N;
dx = 1/M;    dy = 1/M;
dt = 1/N;
x = linspace(0,1,M+1);       % set space grid
y = linspace(0,1,M+1);
xbar = 0.5*(x(2:M+1) + x(1:M));
ybar = 0.5*(y(2:M+1) + y(1:M));
dx2 = dx*dx;   dy2 = dy*dy;
mx = 0.5/dx;
my = 0.5/dy;
% the constant multilpiers
rsx = sigma*dt/(dx2);
rsy = sigma*dt/(dy2);
rkx = kappa*dt/dx2;
rky = kappa*dt/dy2;
R = 0.125;
S = 1/32;
V_old = zeros(M+2,M+2);           %v(t_n)
V_new = zeros(M+2,M+2);           % v(t_n+1)
U_old = ones(M+2,M+2);            % u(t_n)
```

```
U_new = zeros(M+2,M+2);           % u(t_n+1)
Mass = zeros(M*M,M*M);           % the mass matrix
XDiff = zeros(M*M,M*M);          % the diffusion matrix in x direction
YDiff = zeros(M*M,M*M);          % the diffusion matrix in y direction
RHS = zeros(M,M);                % the right hand side  matrix
Theta = zeros(M+1,M+1);          % the Theta's matrix
% multiplier at time t_n
Amin_old = zeros(M+1,M+1);
Amax_old= zeros(M+1,M+1);
Bmin_old = zeros(M+1,M+1);
Bmax_old = zeros(M+1,M+1);
% multiplier atr time t_n+1
Amin_new = zeros(M+1,M+1);
Amax_new= zeros(M+1,M+1);
Bmin_new = zeros(M+1,M+1);
Bmax_new = zeros(M+1,M+1);
globalflux = zeros(M*M,M*M);    % global matrix for the flux
A_flux = zeros(M,M);            % diagonal block matrix
B_flux = zeros(M,M);            % upper sub-diagonal block matrix
C_flux = zeros(M,M);            % lower sub-diagonal block matrix
A_global = zeros(M*M,M*M);      % global matrix of the system
X_diffflux = zeros(M*M,M*M);   % flux diffusion matrix in x direction
Y_diffflux = zeros(M*M,M*M);   % flux diffusion matrix in y direction
Rb = zeros(M*M,1);             % right hand side long vector
% factor matrix for V system
Lv = zeros(1,M);                % lower sub-diagonal
Dv = zeros(1,M);                % diagonal
```

```
Uv = zeros(1,M);                % upper sub-diagonal
ax = -rsx/2;                    % a(i) = a, all i    ,   rsx = rsy
bx = 1 + rsx ;                  % b(i) = b, 1 < i < M
cx = -rsx/2;                    % c(i) = c, all i
% LU factorization  at time  t_n
Dv(1) = 1 + 0.5*rsx ;
Uv(1) = cx;
for i = 2:(M-1)
  Lv(i) = ax/Dv(i-1);
  Dv(i) = bx - Lv(i)*cx;
  Uv(i) = cx;
end
Lv(M) = ax/Dv(M-1);
Dv(M) = (1 + 0.5*rsx ) - Lv(M)*cx;
ay = -rsy/2;
by = 1 + rsy + 0.5*lambda*dt;
cy = -rsy/2;
% LU factorization at time t_n+1
Lvt_n1 = zeros(1,M);     % lower sub-diagonal
Dvt_n1 = zeros(1,M);     % diagonal
Uvt_n1 = zeros(1,M);     % upper sub-diagonal
Dvt_n1(1) = 1 + 0.5*rsy + 0.5*lambda*dt;
Uvt_n1(1) = cy;
for i = 2:(M-1)
  Lvt_n1(i) = ay/Dvt_n1(i-1);
  Dvt_n1(i) = by - Lvt_n1(i)*cy;
  Uvt_n1(i) = cy;
```

```
end
Lvt_n1(M) = ay/Dvt_n1(M-1);
Dvt_n1(M) = (1 + 0.5*rsy + 0.5*lambda*dt) - Lvt_n1(M)*cy;
% variables used in solving Lw = b, Uz = w so LUz = b
w = zeros(1,M);
z = zeros(1,M);
% variables used in setting up the explicit terms in the scheme
Xrhs = zeros(M,1);     % right hand side made up for x-direction
Yrhs = zeros(M,1);     % right hand side made up for y-direction
% The initialization
t = 0;
v = zeros(M,M);     % v = v(x,y,t_{n+1})
v0 = zeros(M,M);    % v = v(x,y,t_{n})
v1 = zeros(M,M);    % this is to be used in solving the y diretion

% ---------------------- constructing the constant matrices ---------
  b = zeros(1,M);
  bx = zeros(1,M);
  a = ones(1,M-1);
  b(2:M-1) = 6*ones(1,M-2);
  b(1) = 7;     b(M) = 7;
  bx(2:M-1) = -2*ones(1,M-2);
  bx( 1) = -1 ;  bx(M) = -1;
% block matrix for the mass matrix
  AMass = diag(b) + diag(a,1) + diag(a,-1);
% block matrix for the X diffusion matrix
  X_block = diag(bx) + diag(a,1) + diag(a,-1);
```

```
% boundary blocks
  Mass(1:M,1:M) = 7*AMass;
  Mass(1:M, M+1:2*M) = AMass;
  Mass((M-1)*M+1:M*M,(M-2)*M+1:(M-1)*M)= AMass;
  Mass((M-1)*M+1:M*M,(M-1)*M+1:M*M)= 7*AMass;
  XDiff(1:M,1:M) = 7*X_block;
  XDiff(1:M, M+1:2*M) = X_block;
  XDiff((M-1)*M+1:M*M,(M-2)*M+1:(M-1)*M)= X_block;
  XDiff((M-1)*M+1:M*M,(M-1)*M+1:M*M)= 7*X_block;
  YDiff(1:M,1:M) = -AMass;
  YDiff(1:M, M+1:2*M) = AMass;
  YDiff((M-1)*M+1:M*M,(M-2)*M+1:(M-1)*M)= AMass;
  YDiff((M-1)*M+1:M*M,(M-1)*M+1:M*M)= -AMass;
  for ii = 2:M-1
  % diagonal blocks
    Mass((ii-1)*M+1:ii*M,(ii-1)*M+1:ii*M) = 6*AMass;
  % off diagonal blocks
    Mass((ii-1)*M+1:ii*M,(ii-1)*M+1+M:ii*M+M) = AMass;
    Mass((ii-1)*M+1:ii*M,(ii-1)*M+1-M:ii*M-M) = AMass;
  % diagonal blocks
    XDiff((ii-1)*M+1:ii*M,(ii-1)*M+1:ii*M) = 6*X_block;
  % off diagonal blocks
    XDiff((ii-1)*M+1:ii*M,(ii-1)*M+1+M:ii*M+M) = X_block;
    XDiff((ii-1)*M+1:ii*M,(ii-1)*M+1-M:ii*M-M) = X_block;
  % diagonal blocks
    YDiff((ii-1)*M+1:ii*M,(ii-1)*M+1:ii*M) = -2*AMass;
  % off diagonal blocks
```

```
    YDiff((ii-1)*M+1:ii*M,(ii-1)*M+1+M:ii*M+M) = AMass;
    YDiff((ii-1)*M+1:ii*M,(ii-1)*M+1-M:ii*M-M) = AMass;
  end
% -----------Simulation loops ------------------------------
  for kk = 1:N
      t1 = t + dt;
% ---------------- solve for v(:,j)^{n+1}----------------------------
% -------------------- solving in the x direction -------------------
 for j = 1:M     % fix j   to solve v in the x direction
% computing the right hand side
  for i = 1:M
   if (j== 1)
     Yrhs(i) = (1-0.5*rsy)*v0(i,1) + 0.5*rsy *v0(i,2)+ ...
      0.5*dt*(-lambda*v0(i,1)+ test2DFfcn(xbar(i),ybar(1),t,parm4f)) ;
   elseif (j < M)
      Yrhs(i) = 0.5*rsy*v0(i,j-1)+(1-rsy)*v0(i,j) + 0.5*rsy *v0(i,j+1)+ ...
      0.5*dt*(-lambda*v0(i,j)+ test2DFfcn(xbar(i),ybar(j),t,parm4f)) ;
   else
      Yrhs(i) = (1-0.5*rsy)*v0(i,M) + 0.5*rsy *v0(i,M-1)+ ...
      0.5*dt*(-lambda*v0(i,M)+ test2DFfcn(xbar(i),ybar(M),t,parm4f)) ;
   end
  end
 w(1) = Yrhs(1);                    % solve Lw = rhs
 for ii = 2:M
   w(ii) = Yrhs(ii) - Lv(ii)*w(ii-1);
 end
z(M) = w(M)/ Dv(M);               % solve Uz = w
```

```
  for ii = M-1:-1:1
    z(ii) = (w(ii) - Uv(ii)*z(ii+1))/Dv(ii);
  end


  for s = 1:M
   v(s,j) = z(s);
  end                  % now v(:,j) = v(:,j)^{n+0.5}
end
 v1 = v;


 % --------------------- solving in the y direction --------------------
for i = 1: M    % fix i to solve in the y direction
 for j = 1:M
   if (i == 1)
     Xrhs(j) = (1-0.5*rsx)*v1(1,j) + 0.5*rsx *v1(2,j)+ ...
              0.5*dt*(test2DFfcn(xbar(1),ybar(j),t1,parm4f)) ;
   elseif ( i < M)
     Xrhs(j) = 0.5*rsx*v1(i-1,j)+(1-rsx)*v1(i,j) + 0.5*rsx *v1(i+1,j)+ ...
              0.5*dt*(test2DFfcn(xbar(i),ybar(j),t1,parm4f)) ;
   else
     Xrhs(j) =  0.5*rsx *v1(M-1,j)+(1-0.5*rsx)*v1(M,j) + ...
              0.5*dt*(test2DFfcn(xbar(M),ybar(j),t1,parm4f)) ;
   end
 end
 w(1) = Xrhs(1);                    % solve Lw = rhs
 for ii = 2:M
     w(ii) = Xrhs(ii) - Lvt_n1(ii)*w(ii-1);
```

```
 end
 z(M) = w(M)/ Dvt_n1(M);               % solve Uz = w
 for jj = M-1:-1:1
   z(jj) = (w(jj) - Uvt_n1(jj)*z(jj+1))/Dvt_n1(jj);
 end
 for s = 1:M
    v(i,s) = z(s);                        % now v(:,j) = v(:,j)^{n+1}
 end
end

% -------------------- solving for U(t_n+1) ------------------------
V_old(2:M+1,2:M+1) = v0;        % the values at t_n
V_new(2:M+1,2:M+1) = v;          % the values at t_n+1
v0 = v;      % updating the values at t_n
% inserting ghost points
 for j = 1:M+2
    V_old(1,j) = V_old(2,j);       % x-direction
    V_old(M+2,j) = V_old(M+1,j);
    V_new (1,j) = V_new(2,j);
    V_new(M+2,j) = V_new(M+1,j);
    U_old(1,j) = U_old(2,j);       % x-direction
    U_old(M+2,j) = U_old(M+1,j);
    U_new (1,j) = U_new(2,j);
    U_new(M+2,j) = U_new(M+1,j);
 end
 for j = 1:M+2
    V_old(j,1) = V_old(j,2);    % y-direction
```

```
   V_old(j,M+2) = V_old(j,M+1);
   V_new(j,1) = V_new(j,2);
   V_new(j,M+2) = V_new(j,M+1);
   U_old(j,1) = U_old(j,2);    % y-direction
   U_old(j,M+2) = U_old(j,M+1);
   U_new(j,1) = U_new(j,2);
   U_new(j,M+2) = U_new(j,M+1);
 end

%-------------------------- Computing Theta's -------------------------
  for i = 1:M+1
     for j = 1:M+1
       Alpha1_old = V_old(i+1,j)- V_old(i,j);
       Alpha2_old = V_old(i+1,j+1)- V_old(i,j+1) ;
       Beta1_old = V_old(i,j+1)- V_old(i,j);
       Beta2_old = V_old(i+1,j+1)- V_old(i+1,j);
     % the derivative V_x(t_n)
       Vx_old = mx*(Alpha1_old + Alpha2_old);
     % the derivative V_y(t_n)
       Vy_old = my*(Beta1_old + Beta2_old);
     % the mix derivative V_xy(t_n)
       Vxy_old = (Beta2_old - Beta1_old)/(dx*dy);
       Alpha1_new = V_new(i+1,j)- V_new(i,j);
       Alpha2_new = V_new(i+1,j+1)- V_new(i,j+1);
       Beta1_new = V_new(i,j+1)- V_new(i,j);
       Beta2_new = V_new(i+1,j+1)- V_new(i+1,j);
     % the derivative V_x(t_n+1)
```

```
      Vx_new = mx*(Alpha1_new + Alpha2_new);
    % the derivative V_y(t_n+1)
      Vy_new = my*(Beta1_new + Beta2_new);
    % the mix derivative V_xy(t_n+1)
      Vxy_new = (Beta2_new - Beta1_new)/(dx*dy);
   % Theta_x at time t_n
     Theta(i,j) = (-0.5*dt*(Vx_old + Vx_new)
                  +0.25*dt^2*Vxy_old*(Vy_old + Vy_new))/dx;
     Amin_old(i,j) = Theta(i,j)*(1-Theta(i,j));
     Amax_old(i,j) = Theta(i,j)*(1+Theta(i,j));
   % Theta_y at time t_n
     Theta(i,j) = (-0.5*dt*(Vy_old + Vy_new)
                  +0.25*dt^2*Vxy_old*(Vx_old + Vx_new))/dy;
     Bmin_old(i,j) = Theta(i,j)*(1-Theta(i,j));
     Bmax_old(i,j) = Theta(i,j)*(1+Theta(i,j));
   % Theta_x at time t_n+1
     Theta(i,j) = (0.5*dt*(Vx_old + Vx_new)
                   + 0.25*dt^2*Vxy_new*(Vy_old + Vy_new))/dx;
     Amin_new(i,j) = Theta(i,j)*(1-Theta(i,j));
     Amax_new(i,j) = Theta(i,j)*(1+Theta(i,j));
   % Theta_y at time t_n+1
     Theta(i,j) = (0.5*dt*(Vy_old + Vy_new)
                   +0.25*dt^2*Vxy_new*(Vx_old + Vx_new))/dy;
     Bmin_new(i,j) = Theta(i,j)*(1-Theta(i,j));
     Bmax_new(i,j) = Theta(i,j)*(1+Theta(i,j));
    end
end
```

```
%------------------------ computing the right hand side ------------
  for i = 1: M
  for j = 1:M
 % constant term from the Mass matrix
   I_mass = 1/64 *(U_old(i,j+2)+ 6*U_old(i,j+1)+U_old(i,j)
          + 6*U_old(i+1,j)+ 36*U_old(i+1,j+1)+6*U_old(i+1,j+2)
          +U_old(i+2,j)+6*U_old(i+2,j+1)+U_old(i+2,j+2));
 % constant term from the diffusion matrix x-direction
   Ix_diff =  kappa*dt/(16*dx2)*( U_old(i,j+2)+ 6*U_old(i,j+1)
          +U_old(i,j)-2*U_old(i+1,j)-12*U_old(i+1,j+1)
          -2*U_old(i+1,j+2)+U_old(i+2,j)+6*U_old(i+2,j+1)
          +U_old(i+2,j+2));
 % constant term from the diffusion matrix y-direction
   Iy_diff = kappa*dt/(16*dy2)*(U_old(i,j+2)-2*U_old(i,j+1)+U_old(i,j)
           + 6*U_old(i+1,j)-12*U_old(i+1,j+1)+6*U_old(i+1,j+2)
           + U_old(i+2,j)-2*U_old(i+2,j+1)+U_old(i+2,j+2));
   RHS(i,j) = I_mass + Ix_diff + Iy_diff ;
 % adding the fluxes
 % adding U(i,j) = U(i+1,j+1)  ( after shifting the indices)
   RHS(i,j) = RHS(i,j) + (R*(Amin_old(i+1,j+1)*Bmin_old(i+1,j+1)
            - Amin_old(i+1,j)*Bmax_old(i+1,j)+ Amax_old(i,j)*Bmax_old(i,j)
            - Amax_old(i,j+1)*Bmin_old(i,j+1))+ S*(3*(Amin_old(i+1,j+1)
            + Amin_old(i+1,j)- Amax_old(i,j+1)- Amax_old(i,j)
            + Bmin_old(i+1,j+1)+ Bmin_old(i,j+1)- Bmax_old(i+1,j)
            - Bmax_old(i,j))))*U_old(i+1,j+1);
   RHS(i,j) = RHS(i,j) + (- R*Amin_old(i,j+1)*Bmax_old(i,j+1)
```

```
          + S*(-Amin_old(i,j+1)+ Bmax_old(i,j+1)))*U_old(i,j+2);
  RHS(i,j) = RHS(i,j) + (R*(- Amin_old(i,j+1)*Bmin_old(i,j+1)
          + Amin_old(i,j)*Bmax_old(i,j)) + S*(3*(-Amin_old(i,j+1)
           - Amin_old(i,j))+ Bmin_old(i,j+1)
           - Bmax_old(i,j)))*U_old(i,j+1);
  RHS(i,j) = RHS(i,j) + (R*Amin_old(i,j)*Bmin_old(i,j)- S*(Amin_old(i,j)
          + Bmin_old(i,j)))*U_old(i,j);
  RHS(i,j) = RHS(i,j) + (R*(-Amin_old(i+1,j)*Bmin_old(i+1,j)
          + Amax_old(i,j)*Bmin_old(i,j))+ S*(Amin_old(i+1,j)
          - Amax_old(i,j) - 3*(Bmin_old(i+1,j)
          + Bmin_old(i,j))))*U_old(i+1,j);
  RHS(i,j) = RHS(i,j)+ (R*(Amin_old(i+1,j+1)*Bmax_old(i+1,j+1)
          - Amax_old(i,j+1)*Bmax_old(i,j+1))+ S*(Amin_old(i+1,j+1)
          - Amax_old(i,j+1)+ 3*(Bmax_old(i+1,j+1)
          + Bmax_old(i,j+1))))*U_old(i+1,j+2);
  RHS(i,j) = RHS(i,j) + (R*Amax_old(i+1,j+1)*Bmax_old(i+1,j+1)
          + S*(Amax_old(i+1,j+1)+ Bmax_old(i+1,j+1)))*U_old(i+2,j+2);
  RHS(i,j) = RHS(i,j) + (R*(Amax_old(i+1,j+1)*Bmin_old(i+1,j+1)
          - Amax_old(i+1,j)*Bmax_old(i+1,j))+ S*(3*(Amax_old(i+1,j+1)
          + Amax_old(i+1,j)) + Bmin_old(i+1,j+1)
          - Bmax_old(i+1,j)))*U_old(i+2,j+1);
  RHS(i,j) = RHS(i,j) + (- R*Amax_old(i+1,j)*Bmin_old(i+1,j)
          + S*(Amax_old(i+1,j)- Bmin_old(i+1,j)))*U_old(i+2,j);
% adding fluxes from the diffusion term
% the x-direction
  RHS(i,j) = RHS(i,j) + R*rky*( -Amin_old(i+1,j+1)- Amin_old(i+1,j)
          + Amax_old(i,j+1)+ Amax_old(i,j))*U_old(i+1,j+1);
```

```
  RHS(i,j) = RHS(i,j) + R*rky*((-Amax_old(i+1,j+1)
           - Amax_old(i+1,j))*U_old(i+2,j+1)+ (Amin_old(i,j+1)
           + Amin_old(i,j))*U_old(i,j+1) - Amin_old(i,j)*U_old(i,j));
  RHS(i,j) = RHS(i,j)+ R*rky*((Amin_old(i+1,j)
           - Amax_old(i,j))*U_old(i+1,j)
           + Amax_old(i+1,j)*U_old(i+2,j)+ (Amin_old(i+1,j+1)
           - Amax_old(i,j+1))*U_old(i+1,j+2)
           - Amin_old(i,j+1)*U_old(i,j+2)
           + Amax_old(i+1,j+1)*U_old(i+2,j+2));
% ydirection
  RHS(i,j) = RHS(i,j) + R*rkx*(- Bmin_old(i+1,j+1)- Bmin_old(i,j+1)
            + Bmax_old(i+1,j)+ Bmax_old(i,j))*U_old(i+1,j+1);
  RHS(i,j) = RHS(i,j) + R*rkx*((-Bmax_old(i+1,j+1)
           - Bmax_old(i,j+1))*U_old(i+1,j+2)+ (Bmin_old(i+1,j)
           + Bmin_old(i,j))*U_old(i+1,j)- Bmin_old(i,j)*U_old(i,j));
  RHS(i,j) = RHS(i,j) + R*rkx*((Bmin_old(i,j+1)
           - Bmax_old(i,j))*U_old(i,j+1)
           + Bmax_old(i,j+1)*U_old(i,j+2)+(Bmin_old(i+1,j+1)
           - Bmax_old(i+1,j))*U_old(i+2,j+1)- Bmin_old(i+1,j)*U_old(i+2,j)
           + Bmax_old(i+1,j+1)*U_old(i+2,j+2));
% adding the source term
  RHS(i,j) = RHS(i,j) + 0.5*dt*(test2DGfn(xbar(i),ybar(j),t1,parm4v,kappa)
           +test2DGfn(xbar(i),ybar(j),t,parm4v,kappa));
     end
 end
```

```
%------------- constructing the global matrix of the flux ---------------
% the indices i and j are shifted one unit due the ghost points i.e. i =
% i+1, j = j+1.
% the corner points when j = 1 and i = 1 or i = M
A_flux(1,1) = R*(Amin_new(2,2)*Bmin_new(2,2))...
             + S*(3*(Amin_new(2,2) +Bmin_new(2,2))
             + 4*(Amin_new(2,1)+Bmin_new(1,2)));
A_flux(1,2) = R*Amax_new(2,2)*Bmin_new(2,2)
            + S*(3*Amax_new(2,2)+ 4 *Amax_new(2,1)+ Bmin_new(2,2));
A_flux(M,M-1)= -R*Amin_new(M,2)*Bmin_new(M,2) + S*(-3*Amin_new(M,2)
             - 4*Amin_new(M,1) + Bmin_new(M,2));
A_flux(M,M) = R*(-Amax_new(M,2)*Bmin_new(M,2))+ S*(4*(- Amax_new(M,1)
            +Bmin_new(M+1,2))+3*(-Amax_new(M,2)+Bmin_new(M,2)));
B_flux(1,1) = R*Amin_new(2,2)*Bmax_new(2,2)+ S*(Amin_new(2,2)
            + 3*Bmax_new(2,2) + 4*Bmax_new(1,2));
B_flux(1,2) = R*Amax_new(2,2)*Bmax_new(2,2) + S*(Amax_new(2,2)
            + Bmax_new(2,2));
B_flux(M,M-1) = -R*Amin_new(M,2)*Bmax_new(M,2)
              + S*(-Amin_new(M,2)+ Bmax_new(M,2));
B_flux(M,M)  = -R* Amax_new(M,2)*Bmax_new(M,2)
             + S*(-Amax_new(M,2)+ 4*Bmax_new(M+1,2)+3*Bmax_new(M,2));
% interior points for j = 1 ( 1< i < M)
  for i = 2:M-1
   A_flux(i,i) = R*(Amin_new(i+1,2)*Bmin_new(i+1,2)
                - Amax_new(i,2)*Bmin_new(i,2))
                + S*(4*(Amin_new(i+1,1)- Amax_new(i,1))+3*(Amin_new(i+1,2)
```

```
            - Amax_new(i,2)+ Bmin_new(i+1,2) + Bmin_new(i,2)));
  A_flux(i,i-1) = -R*Amin_new(i,2)*Bmin_new(i,2)+ S*(-3*Amin_new(i,2)
                  -4*Amin_new(i,1) + Bmin_new(i,2));
  A_flux(i,i+1) = R*Amax_new(i+1,2)*Bmin_new(i+1,2) + S*(3*Amax_new(i+1,2)
                  + 4*Amax_new(i+1,1) + Bmin_new(i+1,2));
  B_flux(i,i) = R*(Amin_new(i+1,2)*Bmax_new(i+1,2)
            - Amax_new(i,2)*Bmax_new(i,2))+ S*(Amin_new(i+1,2)
            - Amax_new(i,2) + 3*(Bmax_new(i+1,2)+ Bmax_new(i,2)));
  B_flux(i,i-1) = -R*Amin_new(i,2)*Bmax_new(i,2)+ S*(-Amin_new(i,2)
                  + Bmax_new(i,2));
  B_flux(i,i+1) = R*Amax_new(i+1,2)*Bmax_new(i+1,2) + S*(Amax_new(i+1,2)
                  + Bmax_new(i+1,2));
end
 globalflux(1:M,1:M) = A_flux;
 globalflux(1:M,M+1:2*M) = B_flux;
 % constructing the interior blocks. We fix the column and construct the
 % matrix. So for every column j we have different matrix.
 for j = 2:M-1
% constructing the diagonal blocks
% these entries are for i = 1
  A_flux(1,1) = R*(Amin_new(2,j+1)*Bmin_new(2,j+1)
            -Amin_new(2,j)*Bmax_new(2,j))+S*(3*(Amin_new(2,j+1)
            + Amin_new(2,j)- Bmax_new(2,j)+ Bmin_new(2,j+1))
            + 4*(Bmin_new(1,j+1)- Bmax_new(1,j)));
  A_flux(1,2) = R*(Amax_new(2,j+1)*Bmin_new(2,j+1)
            - Amax_new(2,j)*Bmax_new(2,j))+ S*(3*(Amax_new(2,j+1)
            + Amax_new(2,j))+ Bmin_new(2,j+1)- Bmax_new(2,j));
```

```
% the off diagonal blocks
% upper diagonal blocks
  B_flux(1,1) = R*(Amin_new(2,j+1)*Bmax_new(2,j+1))
              + S*(Amin_new(2,j+1)+3*Bmax_new(2,j+1)
              + 4*Bmax_new(1,j+1));
  B_flux(1,2) = R*Amax_new(2,j+1)*Bmax_new(2,j+1)
               + S*(Amax_new(2,j+1) + Bmax_new(2,j+1));
% lower diagonal blocks
  C_flux(1,1) = - R*Amin_new(2,j)*Bmin_new(2,j) + S*(Amin_new(2,j)
               - 3*Bmin_new(2,j) - 4*Bmin_new(1,j));
  C_flux(1,2) = - R*Amax_new(2,j)*Bmin_new(2,j) + S*(Amax_new(2,j)
              - Bmin_new(2,j));
for i = 2:M-1   % interior points 1< j < M ; 1< i < M
  A_flux(i,i-1) = R*(-Amin_new(i,j+1)*Bmin_new(i,j+1)
                + Amin_new(i,j)*Bmax_new(i,j))+ S*(3*(-Amin_new(i,j+1)
                - Amin_new(i,j))+ Bmin_new(i,j+1)- Bmax_new(i,j));
  A_flux(i,i) = R*(Amin_new(i+1,j+1)*Bmin_new(i+1,j+1)
              - Amin_new(i+1,j)*Bmax_new(i+1,j)
              + Amax_new(i,j)*Bmax_new(i,j)
              - Amax_new(i,j+1)*Bmin_new(i,j+1))+ S*(3*(Amin_new(i+1,j+1)
              + Amin_new(i+1,j)- Amax_new(i,j+1)- Amax_new(i,j)
              + Bmin_new(i+1,j+1)+ Bmin_new(i,j+1)- Bmax_new(i+1,j)
              - Bmax_new(i,j)));
  A_flux(i,i+1) = R*(Amax_new(i+1,j+1)*Bmin_new(i+1,j+1)
                - Amax_new(i+1,j)*Bmax_new(i+1,j))
                + S*(3*(Amax_new(i+1,j+1)+ Amax_new(i+1,j))
                + Bmin_new(i+1,j+1)- Bmax_new(i+1,j));
```

```
  B_flux(i,i-1) = - R*Amin_new(i,j+1)*Bmax_new(i,j+1)
                + S*(-Amin_new(i,j+1) + Bmax_new(i,j+1));
  B_flux(i,i)   = R*(Amin_new(i+1,j+1)*Bmax_new(i+1,j+1)
                - Amax_new(i,j+1)*Bmax_new(i,j+1))
                + S*(Amin_new(i+1,j+1)- Amax_new(i,j+1)
                + 3*(Bmax_new(i+1,j+1)+ Bmax_new(i,j+1)));
  B_flux(i,i+1) = R*Amax_new(i+1,j+1)*Bmax_new(i+1,j+1)
                 + S*(Amax_new(i+1,j+1)+ Bmax_new(i+1,j+1));
  C_flux(i,i-1) = R*Amin_new(i,j)*Bmin_new(i,j) - S*( Amin_new(i,j)
                + Bmin_new(i,j));
  C_flux(i,i)  = R*(-Amin_new(i+1,j)*Bmin_new(i+1,j)
               + Amax_new(i,j)*Bmin_new(i,j))
               + S*(Amin_new(i+1,j)- Amax_new(i,j)- 3*(Bmin_new(i+1,j)
               + Bmin_new(i,j)));
  C_flux(i,i+1)= - R*Amax_new(i+1,j)*Bmin_new(i+1,j) + S*(Amax_new(i+1,j)
               - Bmin_new(i+1,j));
end
% the entries for i = M
  A_flux(M,M-1) = R*(-Amin_new(M,j+1)*Bmin_new(M,j+1)
                + Amin_new(M,j)*Bmax_new(M,j))
                + S*(-3*(Amin_new(M,j+1)+ Amin_new(M,j))+ Bmin_new(M,j+1)
                - Bmax_new(M,j));
  A_flux(M,M) = R*( Amax_new(M,j)*Bmax_new(M,j)
              - Amax_new(M,j+1)*Bmin_new(M,j+1))
              + S*(3*(- Amax_new(M,j+1)- Amax_new(M,j)+ Bmin_new(M+1,j+1)
              + Bmin_new(M,j+1)- Bmax_new(M+1,j)- Bmax_new(M,j))
              + Bmin_new(M+1,j+1)- Bmax_new(M+1,j));
```

```
  B_flux(M,M-1) = - R*Amin_new(M,j+1)*Bmax_new(M,j+1)
                 + S*(-Amin_new(M,j+1) + Bmax_new(M,j+1));
  B_flux(M,M) = R*(- Amax_new(M,j+1)*Bmax_new(M,j+1))
               + S*(- Amax_new(M,j+1)
               + Bmax_new(M+1,j+1)+ 3*(Bmax_new(M+1,j+1)
                + Bmax_new(M,j+1)));
  C_flux(M,M-1) = R*Amin_new(M,j)*Bmin_new(M,j) - S*(Amin_new(M,j)
                 + Bmin_new(M,j));
  C_flux(M,M) = R*( Amax_new(M,j)*Bmin_new(M,j))
               + S*(- Amax_new(M,j) - Bmin_new(M+1,j) -3*(Bmin_new(M+1,j)
               + Bmin_new(M,j)));
 % storing in the global matrix
   globalflux((j-1)*M+1:j*M,(j-1)*M+1:j*M) = A_flux;
   globalflux((j-1)*M+1:j*M,(j-1)*M+1+M:j*M+M) = B_flux;
   globalflux((j-1)*M+1:j*M,(j-1)*M+1-M:j*M-M) = C_flux;
 end
% the last column j = M
% for i = 1
  C_flux(1,1) = - R*Amin_new(2,M)*Bmin_new(2,M)+ S*(Amin_new(2,M)
               - 3*Bmin_new(2,M)-4*Bmin_new(1,M));
  C_flux(1,2) = - R*Amax_new(2,M)*Bmin_new(2,M)+ S*(Amax_new(2,M)
               - Bmin_new(2,M));
  A_flux(1,1)= R*( - Amin_new(2,M)*Bmax_new(2,M))
              + S*(3*(Amin_new(2,M)- Bmax_new(2,M)) + 4*(Amin_new(2,M+1)
              - Bmax_new(1,M)));
  A_flux(1,2) = R*( - Amax_new(2,M)*Bmax_new(2,M))
               + S*(3*Amax_new(2,M)+ 4*Amax_new(2,M+1)- Bmax_new(2,M));
```

```
% interior points for j = M , 1 < i < M
  for i = 2: M-1
   A_flux(i,i-1) = R*( Amin_new(i,M)*Bmax_new(i,M))
                 + S*(-4*Amin_new(i,M+1) - 3*Amin_new(i,M)- Bmax_new(i,M));
   A_flux(i,i)  = R*(- Amin_new(i+1,M)*Bmax_new(i+1,M)
                + Amax_new(i,M)*Bmax_new(i,M))+S*(3*(Amin_new(i+1,M+1)
                + Amin_new(i+1,M) - Amax_new(i,M+1) - Amax_new(i,M)
                - Bmax_new(i+1,M)- Bmax_new(i,M) )+ Amin_new(i+1,M+1)
                - Amax_new(i,M+1));
   A_flux(i,i+1) = R*(- Amax_new(i+1,M)*Bmax_new(i+1,M))
                 + S*(4*Amax_new(i+1,M+1)+ 3*Amax_new(i+1,M)
                 - Bmax_new(i+1,M));
   C_flux(i,i-1) = R*Amin_new(i,M)*Bmin_new(i,M)- S*(Amin_new(i,M)
                 + Bmin_new(i,M));
   C_flux(i,i) = R*(-Amin_new(i+1,M)*Bmin_new(i+1,M)
               + Amax_new(i,M)*Bmin_new(i,M))
               + S*(Amin_new(i+1,M)- Amax_new(i,M) - 3*(Bmin_new(i+1,M)
               + Bmin_new(i,M)));
   C_flux(i,i+1) = -R*Amax_new(i+1,M)*Bmin_new(i+1,M) + S*(Amax_new(i+1,M)
                 - Bmin_new(i+1,M));
   end
 % for i = M
  A_flux(M,M-1) = R*(+ Amin_new(M,M)*Bmax_new(M,M))...
               + S*(-3*Amin_new(M,M)- 4*Amin_new(M,M+1)- Bmax_new(M,M));
  A_flux(M,M) = R*( Amax_new(M,M)*Bmax_new(M,M)) + S*(3*(- Amax_new(M,M+1)
               - Amax_new(M,M)- Bmax_new(M+1,M)- Bmax_new(M,M))
               - Bmax_new(M+1,M)- Amax_new(M,M+1));
```

```
  C_flux(M,M-1) = R*Amin_new(M,M)*Bmin_new(M,M) - S*(Amin_new(M,M)
                + Bmin_new(M,M));
  C_flux(M,M) = R*( Amax_new(M,M)*Bmin_new(M,M) )+ S*( - Amax_new(M,M)
              - Bmin_new(M+1,M)- 3*(Bmin_new(M+1,M)+ Bmin_new(M,M)));
% storing the last two blocks for j = M
  globalflux((M-1)*M+1:M*M,(M-1)*M+1:M*M) = A_flux;
  globalflux((M-1)*M+1:M*M,(M-1)*M+1-M:M*M-M) = C_flux;


 %-- Constructing the flux matrices from the diffusion term x-direction--
 % for j = 1
   A_flux(1,1)   = -  Amin_new(2,2) ;       % i = 1
   A_flux(1,2)   = -  Amax_new(2,2);
   B_flux(1,1)   = Amin_new(2,2);
   B_flux(1,2)   = Amax_new(2,2);
  for i = 2:M-1
   A_flux(i,i-1) = Amin_new(i,2);
   A_flux(i,i)   = - Amin_new(i+1,2) + Amax_new(i,2);
   A_flux(i,i+1) = - Amax_new(i+1,2);
   B_flux(i,i-1) = - Amin_new(i,2);
   B_flux(i,i)   = Amin_new(i+1,2)- Amax_new(i,2);
   B_flux(i,i+1) = Amax_new(i+1,2);
  end
  A_flux(M,M-1) =   Amin_new(M,2);        % i = M
  A_flux(M,M)   =   Amax_new(M,2);
  B_flux(M,M-1) = - Amin_new(M,2);
  B_flux(M,M)   = - Amax_new(M,2);
% store the first two blocks
```

```
  X_diffflux(1:M,1:M) = A_flux;
  X_diffflux(1:M,M+1:2*M) = B_flux;
 for j = 2:M-1
% the first entries when i = 1, 1< j < M
  A_flux(1,1) = - Amin_new(2,j+1)- Amin_new(2,j);
  A_flux(1,2) = - Amax_new(2,j+1) - Amax_new(2,j);
  B_flux(1,1) = Amin_new(2,j+1);                % upper diagonal block
  B_flux(1,2) = Amax_new(2,j+1);
  C_flux(1,1) = Amin_new(2,j);                 % lower diagonal block
  C_flux(1,2) = Amax_new(2,j);
   for i = 2:M-1
    A_flux(i,i-1)= Amin_new(i,j+1)+ Amin_new(i,j);
    A_flux(i,i) = - Amin_new(i+1,j+1)- Amin_new(i+1,j)+ Amax_new(i,j+1)
                + Amax_new(i,j);
    A_flux(i,i+1) = - Amax_new(i+1,j+1) - Amax_new(i+1,j);
    B_flux(i,i-1) = - Amin_new(i,j+1);
    B_flux(i,i) = Amin_new(i+1,j+1) - Amax_new(i,j+1);
    B_flux(i,i+1) = Amax_new(i+1,j+1);
    C_flux(i,i-1) = - Amin_new(i,j) ;
    C_flux(i,i) = Amin_new(i+1,j) - Amax_new(i,j);
    C_flux(i,i+1) = Amax_new(i+1,j);
 end
% i = M  and 1< j <M
  A_flux(M,M-1) = Amin_new(M,j+1) + Amin_new(M,j);
  A_flux (M,M) =  Amax_new(M,j+1)+ Amax_new(M,j);
  B_flux(M,M-1) = - Amin_new(M,j+1);
  B_flux(M,M) =   - Amax_new(M,j+1);
```

```
  C_flux(M,M-1) = - Amin_new(M,j);
  C_flux(M,M) =  - Amax_new(M,j);
% storing the internal blocks
  X_diffflux((j-1)*M+1:j*M,(j-1)*M+1:j*M)     = A_flux;
  X_diffflux((j-1)*M+1:j*M,(j-1)*M+1+M:j*M+M) = B_flux;
  X_diffflux((j-1)*M+1:j*M,(j-1)*M+1-M:j*M-M) = C_flux;
end
% the last column : j = M
  A_flux(1,1) = - Amin_new(2,M) ;            % i = 1
  A_flux(1,2) = - Amax_new(2,M);
  C_flux(1,1) = Amin_new(2,M);
  C_flux(1,2) = Amax_new(2,M);
 for i = 2:M-1
  A_flux(i,i-1) =   Amin_new(i,M);
  A_flux(i,i)   = - Amin_new(i+1,M)+ Amax_new(i,M);
  A_flux(i,i+1) = - Amax_new(i+1,M);
  C_flux(i,i-1) = - Amin_new(i,M);
  C_flux(i,i)   =   Amin_new(i+1,M)- Amax_new(i,M);
  C_flux(i,i+1) =   Amax_new(i+1,M);
 end
  A_flux(M,M-1) = Amin_new(M,M);           % i = M
  A_flux(M,M)   = Amax_new(M,M);
  C_flux(M,M-1) = - Amin_new(M,M);
  C_flux(M,M) =  - Amax_new(M,M);
  X_diffflux((M-1)*M +1: M*M, (M-1)*M+1:M*M) = A_flux;
  X_diffflux((M-1)*M +1: M*M, (M-1)*M+1-M:M*M-M) = C_flux;
```

```
%-------- constructing the diffusion flux matrix y direction -------------
% the first column j = 1
  A_flux(1,1) = - Bmin_new(2,2);         % i = 1
  A_flux(1,2) =  Bmin_new(2,2);
  B_flux(1,1) = - Bmax_new(2,2);
  B_flux(1,2) =   Bmax_new(2,2);
 for i = 2:M-1
  A_flux(i,i-1) = Bmin_new(i,2);
  A_flux(i,i) = -Bmin_new(i+1,2) - Bmin_new(i,2);
  A_flux(i,i+1) = Bmin_new(i+1,2);
  B_flux(i,i-1) = Bmax_new(i,2);
  B_flux(i,i)   = - Bmax_new(i+1,2) - Bmax_new(i,2);
  B_flux(i,i+1) = Bmax_new(i+1,2);
 end
  A_flux(M,M-1) = Bmin_new(M,2);        % i = M
  A_flux(M,M) = - A_flux(M,M-1);
  B_flux(M,M-1) = Bmax_new(M,2);
  B_flux(M,M) = - Bmax_new(M,2);
% storing the blocks
  Y_diffflux(1:M,1:M) = A_flux;
  Y_diffflux(1:M,M+1:2*M) = B_flux;
% internal blocks 1 < j< M
 for j = 2:M-1
  A_flux(1,1)   = - Bmin_new(2,j+1)+ Bmax_new(2,j);         % i = 1
  A_flux(1,2)   =   Bmin_new(2,j+1) - Bmax_new(2,j);
  B_flux(1,1)   = - Bmax_new(2,j+1);
  B_flux(1,2)   =   Bmax_new(2,j+1);
```

```
 C_flux(1,1)   =   Bmin_new(2,j);
 C_flux(1,2)   = - Bmin_new(2,j);
for i = 2:M-1
 A_flux(i,i-1) = Bmin_new(i,j+1)- Bmax_new(i,j);
 A_flux(i,i)   = - Bmin_new(i+1,j+1)- Bmin_new(i,j+1)
               + Bmax_new(i+1,j)+ Bmax_new(i,j);
 A_flux(i,i+1) = Bmin_new(i+1,j+1)- Bmax_new(i+1,j);
 B_flux(i,i-1) = Bmax_new(i,j+1);
 B_flux(i,i)   = - Bmax_new(i+1,j+1)- Bmax_new(i,j+1);
 B_flux(i,i+1) = Bmax_new(i+1,j+1);
 C_flux(i,i-1) = - Bmin_new(i,j);
 C_flux(i,i)   =   Bmin_new(i+1,j) + Bmin_new(i,j);
 C_flux(i,i+1) = - Bmin_new(i+1,j);
end
 A_flux(M,M-1) =   Bmin_new(M,j+1) - Bmax_new(M,j);     % i = M
 A_flux(M,M)   = - Bmin_new(M,j+1)+ Bmax_new(M,j);
 B_flux(M,M-1) =   Bmax_new(M,j+1);
 B_flux(M,M)   = - Bmax_new(M,j+1);
 C_flux(M,M-1) = - Bmin_new(M,j);
 C_flux(M,M)   =   Bmin_new(M,j);
% storing the blocks in the global matrix
 Y_diffflux((j-1)*M+1:j*M,(j-1)*M+1:j*M)    = A_flux;
 Y_diffflux((j-1)*M+1:j*M,(j-1)*M+1+M:j*M+M)= B_flux;
 Y_diffflux((j-1)*M+1:j*M,(j-1)*M+1-M:j*M-M)= C_flux;
end
% the last column j = M ( the last two blocks )
 A_flux(1,1) =  Bmax_new(2,M);   % i = 1
```

```
 A_flux(1,2) = - Bmax_new(2,M) ;
 C_flux(1,1) = Bmin_new(2,M);
 C_flux(1,2) = - Bmin_new(2,M);
for i = 2:M-1
 A_flux(i,i-1) = - Bmax_new(i,M);
 A_flux(i,i)   =  + Bmax_new(i+1,M) + Bmax_new(i,M);
 A_flux(i,i+1) =  - Bmax_new(i+1,M);
 C_flux(i,i-1) = - Bmin_new(i,M);
 C_flux(i,i)   =   Bmin_new(i+1,M) + Bmin_new(i,M);
 C_flux(i,i+1) = - Bmin_new(i+1,M);
end
 A_flux(M,M-1) = - Bmax_new(M,M);   % i = M
 A_flux(M,M)   = Bmax_new(M,M);
 C_flux(M,M-1) = - Bmin_new(M,M);
 C_flux(M,M)   = Bmin_new(M,M);
% storing the blocks
 Y_diffflux((M-1)*M+1:M*M,(M-1)*M+1:M*M)= A_flux;
 Y_diffflux((M-1)*M+1:M*M,(M-2)*M+1:(M-1)*M)= C_flux;
% The global matrix of the system
 A_global = 1/64*Mass + globalflux - 0.5*R * (rkx*XDiff + rky* YDiff)...
          -R *(rkx*Y_diffflux + rky*X_diffflux);
% convert the right hand side  to long vector
for L = 1:M
    Rb(1+(L-1)*M:L*M) = RHS(:,L);
end
 A_global = sparse ( A_global);   % store sparse matrix
% solving the system
```

```
 Usol_1 = A_global\Rb;
 Usol = zeros(M,M);
% Usol = (vec2mat(Usol_1,M))';              % convert vector to matrix
 for L = 1:M
     Usol(:,L) = Usol_1(1+ (L-1)*M: L*M);
 end
% U_new = U_old;
 U_old(2:M+1,2:M+1) = Usol;
 t = t1;
 end
 v_exact = ExactV(parm4v,xbar,ybar,t,M);    % exact solution of v
 U_exact = exp(v_exact);          % exact solution of U
 v_err(k+1) = norm(abs(v - v_exact),'fro')/M;        % error
 V_rate = v_err(k)/v_err(k+1);               % rate of convergence
 U_error(k+1) = norm(abs(U_exact - Usol),'fro')/M;
 U_rate = U_error(k)/U_error(k+1);
disp(sprintf(' %1d %5d %4d  %4.2e  %5.2e  %5.2e  %5.2e',k,N,M,v_err(k+1),
  V_rate,U_error(k+1), U_rate))
end
figure
mesh(xbar,ybar,v_exact)
xlabel('x-axis'); ylabel('y-axis')
title('exact solution of V')
figure
 mesh(xbar,ybar,v);% zlim([0,2.1])
 xlabel('x-axis'); ylabel('y-axis')
 title('approximate solution of V')
```

```
figure
mesh(xbar, ybar,U_exact)
xlabel('x-axis') ; ylabel('y-axis')
title ( ' Exact soluion for U')
figure
mesh(xbar, ybar, Usol)
xlabel('x-axis'); ylabel('y-axix')
title(' approximate solution of U ')
```

(test2DFfcn)

```
% This function defines the source term f(x,y,t) in the
% model system equation for v(x,y,t) with known solutions
function f = test2DFfcn(x,y,t,parm4F)
lambda = parm4F(1);   sigma = parm4F(2); n = parm4F(3); m = parm4F(4);
f0 = parm4F(5);     fn = parm4F(6);
f = f0 + fn*cos(n*pi*x)*cos(m*pi*y);
```

(test2DGfn)

```
% This function returns the source term g(x,y,t,u,v) in the u equation
function G = test2DGfn(x,y,t,parm4V,kappa)
lambda = parm4V(1);   sigma = parm4V(2);   n = parm4V(3);   m = parm4V(4);
f0 = parm4V(5) ;      fn = parm4V(6);
a0 = lambda;
an= sigma*(n*n + m*m)*pi*pi+lambda;
v = (f0/a0)*(1 - exp(-a0*t));
v = v + (fn/an)*(1 - exp(-an*t))*cos(n*pi*x)*cos(m*pi*y);
v_t = f0*exp(-lambda*t)+ fn*exp(-an*t)*cos(n*pi*x)*cos(m*pi*y);
```

```
v_x = -fn/an * (n*pi)* (1-exp(-an*t))*sin(n*pi*x)*cos(m*pi*y);
v_xx = -fn/an * (n*pi)^2 *(1-exp(-an*t))*cos(n*pi*x)*cos(m*pi*y);
v_y = -fn/an * (m*pi) *(1-exp(-an*t))*cos(n*pi*x)*sin(m*pi*y);
v_yy = -fn/an * (m*pi)^2 *(1-exp(-an*t))*cos(n*pi*x)*cos(m*pi*y);
G = exp(v)*(v_t + (1-kappa)*(v_x^2 + v_y^2 + v_xx + v_yy));
```

(ExactV)

```
% This function returns the exact solution of v
function [V] = ExactV(parm4V,x,y,t,M)
lambda = parm4V(1);   sigma = parm4V(2); n = parm4V(3); m = parm4V(4);
f0 = parm4V(5);    fn = parm4V(6);
a0 = lambda;
an= sigma*(n*n + m*m)*pi*pi+lambda;
v1 = (f0/a0)*(1 - exp(-a0*t));
V = zeros(M,M);
% Exact solution
for i =1:M
    for j =1:M
     V(i,j) = v1 + (fn/an)*(1 - exp(-an*t))*cos(n*pi*x(i))*cos(m*pi*y(j));
    end
end
```

BIBLIOGRAPHY

1. E.F. Keller and L.A. Segel, Initiation of slime mold aggregation viewed as an instability, *J Theor Biol* 26(1970), 399-415.

2. D. Horstmann, From 1970 until present: the Keller-Segel model in chemotaxis and its consequences, I Jahresber. Deutsch. Math.-Verein.,Vol. 105 (2003), 103-65.

3. R.E Ewing and H. Wang, A summary of numerical methods for time-dependent advection-dominated partial differential equation, Journal of Computational Applied Math 128 (2001), 423-445.

4. M.W. Smiley, A monotone conservative Eulerian-Lagrangian scheme for reaction-diffusion-convection equations modeling chemotaxis, Numerical Methods for Partial Differential Equations, 23(2007), 553-586.

5. M.W. Smiley, An efficient implementation of numerical method for a chemotaxis system, International Journal of Computer Mathematics, 86(2009), 219-235.

6. A. Chertock and A. Kurganov, A second-order positivity preserving central-upwind scheme for chemotaxis and haptotaxis models, Numer. Math., 111(2008), 169-205.

7. K. W. Morton, Numerical solution of convection-diffusion problems, Chapman and Hall, London, 1996.

8. Peter H. Lauritzen, Ramachandran D. Nair, Paul A. Ullrich, A conservative semi-Lagrangian multi-trace transport scheme (CSLAM) on the cubed-sphere grid, Journal of Computational Physics 229(2010), 1401-1424.

9. T. Nagai, Blowup of nonradial solutions to parabolic-elliptic systems modeling chemotaxis in twodimensional domains, J. Inequal. Appl. 6 (2001), 37-55.

10. K. Osaki and A. Yagi, Finite dimensional attractors for one-dimensional Keller-Segel equations. Funkcialaj Ekvacioj 44 (2001), 441-469.

11. N. Saito, Conservative upwind finite-element method for a simplified Keller-Segel system modelling chemotaxis, IMA J. Numer. Anal., 27 (2007), 332-365.

12. T. Senba and T. Suzuki, Parabolic System of Chemotaxis: Blowup in a Finite and the Infinite Time,Methods Appl. Anal. 8 (2001), 349-368.

13. N. Saito, Conservative numerical schemes for the Keller-Segel system and numerical results, RIMS Kôkyûroku Bessatsu, B15 (2009), 125-146.

14. Y. Epshteyn and A. Kurganov, New interior penalty discontinuous Galerkin methods for the Keller-Segel chemotaxis model, SIAM Journal on Numerical Analysis , Volume 47(2008), p.386-408.

15. Y. Epshteyn, Discontinuous Galerkin methods for the chemotaxis and haptotaxis models, J. Comput. Appl. Math. 224, no. 1 (2009), 168-181.

16. C.J Budd, R. Carretero-Gonzàlez and R.D. Russel, Precise computations of chemotactic collapse using moving mesh methods, Journal of Computational Physics 202(2005), 463-487.

17. H.-G.Roos, M.Stynes, L.Tobiska, Numerical methods for singularly perturbed differential equations convection-diffusion and flow problems,Springer Verlag, Berlin, 1996

18. J.Doglus, Jr. and T.F. Russel, Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures, SIAM J Numer Anal 19 (1982), 871-885.

19. M. A. Celia, T. F. Russell, I. Herrera, and R. E. Ewing, An Eulerian-Lagrangian localized adjoint method for the advection-diffusion equation, Adv. Water Resour 13 (1990), 187-206.

20. T. Arbogast and M. F. Wheeler, A characteristics-mixed finite element method for advection-dominated transport problems, SIAM J. Numer. Anal. 32 (1995), 404-424.

21. T. N. Phillips and A. J. Williams, Conservative semi-Lagrangian finite volume schemes, Numer. Methods Partial Differential Eq. 17 (2001), no. 4, 403-425.

22. T. N. Phillips and A. J. Williams, A semi-Lagrangian finite volume method for solving viscoelastic flow problems, In: I. Emri and R. Cvelbar, editors, Proc Fifth Euro Rheology Conf, Portorož, Slovenia, 1998, Steinkopff, 299-300.

23. T. N. Phillips and A. J. Williams,Viscoelastic flowthrough a planar contraction using a semi-Lagrangian finite volume method, J Non-Newtonian Fluid Mech 246 (1999), 215-246.

24. C. Chen; W. Liu and C. Bi, A two-grid characteristic finite volume element method for semilinear advection-dominated diffusion equations. Numer. Methods Partial Differential Equations 29 (2013), no. 5, 1543-1562.

25. Yang, Min; Yuan, Yi-rang, A symmetric characteristic finite volume element scheme for nonlinear convection-diffusion problems. Acta Math. Appl. Sin. Engl. Ser. 24 (2008), no. 1, 41-54.

26. J. W. Thomas, Numerical partial differential equations, Springer Verlag, New York, 1995

27. H. Notsu, H. Rui and M. Tabata,Characteristics finite difference schemes of second order in time for convection-diffusion problems, WIAS DP-2011-004, 2012.

28. Todd F. Dupont and Yingjie Liu, Back and Forth Error Compensation and Correction Methods for Semi-Lagrangian Schemes with Application to Level Set Interface Computations. Math. Comp., 76 (2007), pp 647–668.

29. Lloyd N. Trefethen, David Bau, Numerical linear algebra, Siam, Philadelphia, 1997.

30. J. Elliot, The characteristic roots of certain real symmetric matrices, Masters thesis, University of Tennessee, 1953.

31. J.W. Demmel, Applied numerical linear algebra, SIAM.Philadelphia, PA, 1997.

32. M. Marcus and H. Minc, A Survey of matrix theory and matrix inequalities, Allyn and Bacon, Inc., Boston, 1964.