

ABSTRACT

JIANG, QUNLEI. Analysis and Computation for a Fluid Mixture Model of Tissue Deformations. (Under the direction of Professors Zhilin Li and Sharon R. Lubkin).

A fluid mixture model of tissue deformations in one and two dimensions has been studied in this dissertation. The model is a mixed system of nonlinear hyperbolic and elliptic partial differential equations with interfaces. Both theoretical and numerical analysis are presented. We found the relationship between physical parameters and the pattern of tissue deformations via linear stability analysis. Several numerical experiments support our theoretical analysis.

The solution of the system exhibits non-smoothness and discontinuities at the interfaces. The conventional high order finite difference methods (FDM), such as WENO scheme and TVD Runge Kutta method, for the hyperbolic equation coupled with the central FDM for the elliptic equation give spurious oscillations near the interfaces in our problem. By enforcing the jump conditions across the interfaces, our approach, the immersed interface method (IIM), eliminates non-physical oscillations, improves the accuracy of the solution, and maintains the sharp interface as time evolves.

The IIM has been applied to solve a one dimensional linear advection equation with discontinuous initial conditions. By building the jump conditions into the conventional finite difference method, the Lax-Wendroff method, the solutions of second order accuracy are observed. The IIM showed its robustness to solve the linear advection equation with nonhomogeneous jump conditions across the moving interface.

The two dimensional fluid mixture model has been derived asymptotically from the three dimensional model so that the thickness of the gel is taken into account. A lot of numerical examples have been completed using Clawpack and expected numerical solutions have been obtained.

Analysis and Computation for a Fluid Mixture Model of Tissue Deformations

by
Qunlei Jiang

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Computational Mathematics

Raleigh, North Carolina

2008

APPROVED BY:

Dr. Zhilin Li
Chair of Advisory Committee

Dr. Sharon Lubkin
CoChair of Advisory Committee

Dr. Hien Tran

Dr. Kazufumi Ito

Dr. Xiao-Biao Lin

DEDICATION

To: My heavenly Father God, to my parents, and to my husband Yanping Xu.

BIOGRAPHY

Qunlei Jiang was born on September 25, 1970 in Cixi, Zhejiang Province, P. R. China. She was admitted to Shanghai Jiaotong University 1990 and earned her Bachelor of Engineering degree in the Department of Engineering Mechanics with major of Nuclear Engineering in 1994.

After graduation, she found a job in China Institute of Atomic Energy. She worked there as a research assistant, later research associate, on designing reactor core for a research reactor. During the years in China Institute of Atomic Energy, she earned her M.S. in Nuclear Engineering from the Graduate School of China Nuclear Industry Corporation as a part time student.

Qunlei Jiang was admitted to the Ph.D program in Department of Nuclear Engineering at North Carolina State University in 2000.

In the fall of 2003, Qunlei Jiang changed her Ph.D program in Nuclear Engineering to a master's program and transferred to the Department of Mathematics at North Carolina State University. She entered the Ph.D program in Computational Mathematics with a brand new field in her life.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor, Dr. Zhilin Li, for his constant guidance, frequent encouragement and generous support throughout my graduate years at North Carolina State University. It was Dr. Li who led me into the field of computational mathematics from the field of nuclear engineering. I have greatly benefited from countless discussions with Dr. Li.

I would also like to thank the cochair of my committee, Dr. Sharon Lubkin, for her invaluable guidance, suggestions and generous support. For the rest of my committee members, Dr. Hien Tran, Dr. Kazufumi Ito, and Dr. Xiao-Biao Lin, I am much indebted to them for their valuable comments and suggestions.

I am thankful to the faculty, staff and students in the Department of Mathematics for their kind help and warm smile.

During my graduate study, my family has been very supportive. Their unconditional support has made my study easier. Finally, I would like to thank all dear brothers and sisters in Church in Raleigh for their love and support.

This work was partially supported by NSF-NIH 0201094, NSF-DMS-0412654, ARO 43751-MA and 18526-MA, and AFSOR-FA9550-06-1-0241 in USA.

TABLE OF CONTENTS

LIST OF TABLES.....	vii
LIST OF FIGURES	viii
1 Introduction.....	1
1.1 A Fluid Mixture Model of Tissue Deformations	1
1.2 A Brief Review of Literature	3
1.3 Our Strategy - the Immersed Interface Method	9
1.3.1 The Process	9
1.3.2 The Finite Difference Schemes for the Hyperbolic Equation	10
1.4 Contributions of the Dissertation	11
1.5 Overview of the Dissertation	13
2 WENO and IIM for a 1D Fluid Mixture Model of Tissue Deformation .	14
2.1 The Mathematical Model	14
2.2 The Linear Stability Analysis	16
2.3 Numerical Methods	25
2.3.1 The Upwind Scheme and Central FDM	25
2.3.2 The WENO-Roe Scheme	26
2.3.3 The Time Discretization Using the TVD Runge-Kutta Method	29
2.3.4 Applying the Immersed Interface Method	33
2.4 Numerical Results	40
2.4.1 Cases with parameters of Model 1	40
2.4.2 Cases with parameters of Model 2 and $\theta_l = 0$	41
2.4.3 Cases with parameters of Model 2 and $\theta_l = 0.1$	46
2.4.4 Cases with parameters of Model 3	50
2.4.5 Cases with parameters of Model 4	52
2.5 Conclusions	52
3 IIM for a Linear Advection Equation	56
3.1 A Linear Advection Equation	56
3.2 The Immersed Interface Method - Modification for Lax-Wendroff Method	59
3.3 Numerical Examples	62
3.4 Conclusions	69
4 IIM for a 1D Fluid Mixture Model of Tissue Deformation with Constant Physical Parameters	70
4.1 Numerical Methods	71
4.1.1 The Conventional Finite Difference Method for Regular Grid Points	71
4.1.2 Applying the Immersed Interface Method	72

4.2	Numerical Example	80
4.2.1	The IIM and the Leapfrog Method	80
4.2.2	The IIM and MacCormack's Method	84
4.3	Conclusions	88
5	A 2D Fluid Mixture Model of Tissue Deformations	89
5.1	Introduction	89
5.1.1	Deriving Body Force	91
5.1.2	Simplification	94
5.1.3	Nondimensionalization	95
5.1.4	Stability Analysis	95
5.2	Numerical Methods	99
5.3	Numerical Examples	101
5.3.1	Cases with Regular Initial Data	101
5.3.2	Cases with Random Initial Data	101
5.4	Conclusions	105
6	Conclusions and Future Work	114
6.1	Current Work	114
6.2	Future Work	116
6.2.1	Problems with Variable Coefficients	116
6.2.2	Theoretical Analysis for the Immersed Interface Method	116
6.2.3	Two Dimensional Problem with Discontinuous Initial Conditions . .	117
6.2.4	Two Dimensional Problem with Non-Discontinuous Initial Conditions	117
	Bibliography	119
	Appendix	127
	Appendix A. Accuracy of MacCormack's Method for a Fluid Mixture Model of Tissue Deformations	128

LIST OF TABLES

Table 2.1	Expected ranges of parameter values [2, 3].	15
Table 2.2	Choices of the nondimensionalized functions M , φ , ψ , and σ	18
Table 2.3	The constants c_{rj}	27
Table 2.4	Accuracy on θ in the System with $\theta_0(x) = \frac{1}{2} + \frac{1}{8} \cos(2\pi x)$	30
Table 2.5	Accuracy on v in the System with $\theta_0(x) = \frac{1}{2} + \frac{1}{8} \cos(2\pi x)$	30
Table 2.6	Accuracy on θ in the System with Discontinuous Initial Data.	31
Table 2.7	Accuracy on v in the System with Discontinuous Initial Data.	32
Table 2.8	List of Numerical Experiments.	41
Table 3.1	Accuracy of IIM for solving a linear advection equation with initial condition in (3.4).	64
Table 3.2	Accuracy of IIM for solving a linear advection equation with initial condition in (3.5).	65
Table 4.1	Accuracy of θ solved by the IIM based on the leapfrog method for the system in (4.1) with constant physical parameters.	81
Table 4.2	Accuracy of v solved by the IIM based on the leapfrog method for the system in (4.1) with constant physical parameters.	82
Table 4.3	Accuracy of θ solved by the leapfrog method for the hyperbolic equation in (4.1) with constant physical parameters.	83
Table 4.4	Accuracy of θ solved by the IIM based on MacCormack's method for the hyperbolic equation in (4.1) with constant physical parameters.	86
Table 5.1	Expected ranges of parameter values [2, 3].	91

LIST OF FIGURES

Figure 1.1	A bio-artificial tendon (BAT) may be constructed using the linear anchor and stem configuration. Cells consolidate the matrix and contract the gel over time. Garvin et al, 2003 [1]	2
Figure 1.2	Diagram of the stencils. (a) Lax-Wendroff method; (b) Beam-Warming method; (c) Leapfrog method.	11
Figure 2.1	A typical initial data of θ	19
Figure 2.2	Plot of stability regions of system with $\sigma = 1$, $\psi = \frac{\sigma}{1-\theta}$, $\psi_{equiv.}$ determined by (2.23), and $\theta_l = 0.1$. The dashed line and dash-dot line indicate the ψ with which the system is neutrally stable for θ_u and θ_l , respectively; the solid line indicates the ψ value with which the system is at equilibrium.....	20
Figure 2.3	Solution plots with the perturbed initial data (2.24) with $M = 0.5$, $\varphi = 0$, $\sigma = 1$, $\psi = 1.47$, $\theta_u = 0.5$, $\theta_l = 0.1$, $\varepsilon = 0.05$, $k = 60\pi$, $x_1 = 0.4$, and $x_2 = 0.6$. In both plots, the dash-dot line shows the steady state; the dotted one shows the initial data and the solid one shows the solution at $t = 20$	21
Figure 2.4	Maximum frequency evaluated by (2.27) with $M = \frac{\theta}{0.5+\theta}$, $\varphi = 1$, $\psi = \theta$, $\sigma = 1 - \theta$, and $\varepsilon = 0.001$	24
Figure 2.5	Diagram indicates the sign of λ evaluated by $1.8(1 - \theta_0) - \ln(1 - \theta_0) - \frac{1}{1-\theta_0}$ for the system with $M = 0.5e^\theta$; $\varphi = e^\theta$, $\psi = 1.8e^{-\theta}$ and $\sigma = e^{-\theta}$	24
Figure 2.6	Linear regression analysis of the convergence order of θ in log-log scale for the system in (2.50) at $t = 0.05$. The average convergence order for the L_∞ and L_1 errors are 2.01 and 2.11, respectively.	31
Figure 2.7	Linear regression analysis of the convergence order of v in log-log scale for the system in (2.50) at $t = 0.05$. The average convergence order for the L_∞ , L_1 and L_2 errors are 2.04 and 2.04, respectively.....	32
Figure 2.8	Linear regression analysis of the convergence order of θ in log-log scale for the system in (2.50) with the discontinuous IC at $t = 0.05$. The average convergence order for the L_∞ and L_1 errors are 1.06 and 1.15, respectively.....	33
Figure 2.9	Linear regression analysis of the convergence order of v in log-log scale for the system in (2.50) with the discontinuous IC at $t = 0.05$. The average convergence order for the L_∞ , L_1 and L_2 errors are 1.02 and 1.06, respectively...	34

Figure 2.10	1-D interface.....	37
Figure 2.11	Computed solutions of (2.1), (2.2) and (2.3) along with IC in (2.5) and parameters in (2.25) by the upwind scheme along with CFDM. (a) – the solutions of θ (left), velocity (middle) and pressure (right) for the case with $\varepsilon = 0.001$, $t \leq 5.0$, and $k = 2$; (b), (c) and (d) – similar to (a), but with $k=6, 8$ and 18 , respectively. In all plots, the dotted lines represent the initial data, the solid lines represent the intermediate solutions between initial and final time, and the * lines in (a) represent the solutions at $t = 5.0$, those in (b), (c) and (d) are the solutions when $\theta \geq 1$. In (d), solutions blow up at $t = 0.1$. $N = 160$	42
Figure 2.12	$[v_x]$ for the case with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$	43
Figure 2.13	Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$ for $T = 5$. (1) – the solutions of θ (left) and velocity (right) of Case 1 solved via the upwind scheme along with CFDM; (2) – the solutions of θ (left) and velocity (right) of Case 2 solved via the upwind scheme along with IIM. Fine meshes, $N = 640$	44
Figure 2.14	Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$ for $T = 15$ by the upwind scheme along with CFDM. (3) – the solutions of θ and velocity of Case 3 with fine meshes, $N = 640$; (4) – the solutions of θ and velocity of Case 4 with coarse meshes, $N = 80$	45
Figure 2.15	Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\sigma = 1$ by the upwind scheme along with CFDM. (5) – the solutions of θ (left) and velocity (right) of Case 5 with $\psi = 2.5$ and fine meshes, $N = 640$, $T = 5$; (6) – the solutions of θ (left) and velocity (right) of Case 6 with $\psi = 1$ and coarse meshes, $N = 80$, $T = 20$. In plots of Case 5, the dashed lines indicate the computed solutions while passing the neutral stable state at $t = 0.9$ (referring the dashed line in Fig. 2.2).	47
Figure 2.16	Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$ by the WENO-Roe scheme along with TVD Runge-Kutta method and CFDM. (7) – the solutions of θ (left) and velocity (right) of Case 7 with $N = 640$, $T = 5$; (8) – the solutions of θ (left) and velocity (right) of Case 8 with $N = 320$, $T = 15$	48
Figure 2.17	Computed solutions using the WENO scheme and central finite difference scheme with different parameters. The top two plots are computed with $\psi = 2.5$, $\sigma = 1$, and $N = 640$. The solution θ grows in the middle. Some oscillations developed near the discontinuities. The bottom plots are computed with $\psi = 1$, $\sigma = 1$, and $N = 80$. The solution θ decays and no oscillations occurred so we take a coarse grid. The final time is $T = 5$ for the top plots while it is $T = 20$ for the bottom ones since it takes a longer time to reach the steady state solution.....	49

Figure 2.18	Computed solution using the IIM scheme. The results are computed with $\psi = 1.8$, $\sigma = 1$, and $N = 640$. The final time is $T = 5$. There are no oscillations in the computed solution.	50
Figure 2.19	Computed solutions of (2.31), (2.32) along with IC in (2.6) and parameters of Model 3 in Table 2.2. (12) – the solutions of θ (left) and velocity (right) of Case 12 with $T = 5$ and $N = 640$, solved by the upwind scheme along with IIM; (13) – the solutions of θ (left) and velocity (right) of Case 13 with $T = 5$ and $N = 640$, solved by the upwind scheme along with CFDM; (14) – the solutions of θ (left) and velocity (right) of Case 14 with $T = 15$, solved by the upwind scheme along with CFDM.	51
Figure 2.20	Computed solutions of (2.31), (2.32) along with IC in (2.6) and parameters in (2.28). (15) – the solutions of θ (left) and velocity (right) of Case 15 with $T = 5$ and $N = 640$, solved by the upwind scheme along with IIM; (16) – the solutions of θ (left) and velocity (right) of Case 16 under same conditions as Case 15 but solved by the upwind scheme along with CFDM; (17) – the solutions of θ (left) and velocity (right) of Case 17 similar to Case 16 but with $T = 15$	53
Figure 2.21	Computed solutions of (2.31), (2.32) along with IC in (2.6) and parameters in (2.28) solved via the WENO-Roe scheme along with TVD Runge-Kutta method and CFDM. (18) – the solutions of θ (left) and velocity (right) of Case 18 with $T = 5$ and $N = 640$; (19) – the solutions of θ (left) and velocity (right) of Case 19 with $T = 15$ and $N = 320$	54
Figure 3.1	Initial condition for the computations. Top left: piecewise constant initial condition described in (3.3). Top right: piecewise C^∞ initial condition described in (3.4) with $[q] \neq 0$ only. Bottom left: piecewise C^∞ initial condition described in (3.5) with $[q] \neq 0$, $[q_x] \neq 0$ and $[q_{xx}] \neq 0$. Bottom right: derivatives of the initial condition described in (3.5).	57
Figure 3.2	Diagram of the stencils for irregular grid points.	58
Figure 3.3	Diagram of the interface location. (a) no crossing, (b) crossing a grid line.	60
Figure 3.4	1-D linear advection with $v = 1$ and initial condition in (3.4). Comparison of computed solution and exact solution at $t = 0.1, 0.2, 0.3$, and 0.4 respectively. The solid line represents the exact solution and the symbol 'o' represents the numerical solutions. The mesh size for the computation is $h = 1/80$	63
Figure 3.5	1-D linear advection with $v = 1$ and initial condition in (3.4). Comparison of computed solution and exact solution at $t = 0.1, 0.2, 0.3$, and 0.4 respectively. The solid line represents the exact solution and the symbol 'o' represents the numerical solutions. The mesh size for the computation is $h = 1/80$	65

- Figure 3.6 Linear regression analysis of the convergence order in log-log scale for 1-D linear advection with $v = 1$ and initial condition in (3.4) at $t = 0.5$. The average convergence order for the L_∞ , L_1 and L_2 errors are 2, 2.02 and 2.03, respectively. 66
- Figure 3.7 1-D linear advection with $v = 1$ and initial condition in (3.5). Comparison of computed solution and exact solution at $t = 0.1, 0.2, 0.3$, and 0.4 respectively. The solid line represents the exact solution and the symbol 'o' represents the numerical solutions. The mesh size for the computation is $h = 1/80$ 67
- Figure 3.8 Linear regression analysis of the convergence order in log-log scale for 1-D linear advection with $v = 1$ and initial condition in (3.5) at $t = 0.5$. The average convergence order for the L_∞ , L_1 and L_2 errors are 1.99, 2.04 and 2.03, respectively. 68
- Figure 4.1 Diagram of the interface location for a two-level method. (a) no crossing, (b) crossing a grid line as $\alpha_i(t)$ increases with time, (c) crossing a grid line as $\alpha_i(t)$ decreases with time. 73
- Figure 4.2 Diagram of the interface location for a three-level method. (a) no crossing, (b) crossing a grid line as $\alpha_i(t)$ increases with time at $t^n < \tau_2 < t^{n+1}$ only, (c) crossing a grid line as $\alpha_i(t)$ increases with time at $t^{n-1} < \tau_1 < t^n$ only, (d) crossing a grid line as $\alpha_i(t)$ increases with time at both $t^{n-1} < \tau_1 < t^n$ and $t^n < \tau_2 < t^{n+1}$, (e) crossing a grid line as $\alpha_i(t)$ decreases with time at $t^n < \tau_2 < t^{n+1}$ only, (f) crossing a grid line as $\alpha_i(t)$ decreases with time at $t^{n-1} < \tau_1 < t^n$ only, (g) crossing a grid line as $\alpha_i(t)$ decreases with time at both $t^{n-1} < \tau_1 < t^n$ and $t^n < \tau_2 < t^{n+1}$. 76
- Figure 4.3 Comparison of solution of θ solved by the IIM based on the leapfrog method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$ 81
- Figure 4.4 Comparison of solution of v solved by the IIM based on the leapfrog method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$ 82
- Figure 4.5 Computed solutions of (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ for $T = 0.1$. (1) – the solutions of θ (left) and v (right) solved via the leapfrog method; (2) – the solutions of θ (left) and v (right) solved via the IIM based on leapfrog method. Fine meshes, $N = 640$ 83

- Figure 4.6 Computed solutions of (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ for $T = 5$ solved by the IIM based on the leapfrog method (the solutions of θ : left plot, and v : right plot). The dotted lines represent the initial data, dash-dotted lines represent the solution at the final time ($T = 5$), and solid lines represent the intermediate solution between initial and final time. Fine meshes, $N = 640$ 84
- Figure 4.7 Comparison of solution of θ solved by the IIM based on MacCormack's method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$ 85
- Figure 4.8 Comparison of solution of v solved by the IIM based on MacCormack's method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$ 86
- Figure 4.9 Enlargement of simulations from Fig. 4.7. Partial computed solution of θ near the interface solved by the IIM based on MacCormack's method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$. Oscillations are observed in the enlarged plots. 87
- Figure 4.10 Computed solutions θ of (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ for $T = 5$ solve by two algorithms. Left: using MacCormack's method; right: using the IIM based on MacCormack's method. The dotted lines represent the initial data, dash-dotted lines represent the solution at the final time ($T = 5$), and solid lines represent the intermediate solution between initial and final time. Fine meshes, $N = 640$. Oscillations are observed. 88
- Figure 5.1 Possible distribution of ψ 96
- Figure 5.2 Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with piecewise constant initial condition, in which $\theta_0 = 0.5$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 3, 6, 9$, and 12 (pcolor plot: 82×82). 102
- Figure 5.3 Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.8), in which $\theta_0 = 0.2$ and $\varepsilon = 0.01$, and with

- nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82)..... 103
- Figure 5.4 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.8), in which $\theta_0 = 0.2$ and $\varepsilon = 0.01$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82)..... 104
- Figure 5.5 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82)..... 106
- Figure 5.6 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82)..... 107
- Figure 5.7 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82)..... 108
- Figure 5.8 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82)..... 109
- Figure 5.9 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, on a 5×5 domain, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82)..... 110
- Figure 5.10 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, on a 5×5 domain, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82)..... 111
- Figure 5.11 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, and dimensional parameters $h(x) = 0.1 + \frac{0.4}{15}x$, on a 15×5 domain, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 194×66)..... 112
- Figure 5.12 Solution of θ, u, v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with

nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, and dimensional parameters $h(x) = 0.1 + \frac{0.4}{15}x$, on a 15×5 domain, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 194×66). 113

Figure 6.1 Maximum of minimum values of $\Delta\theta$ along with time 82×82 . The simulation crashes at $t = 17.7$. The plot on the left is a plot of $|\Delta\theta|_{max}$ vs time. The plot on the right is a plot of $|\Delta\theta|_{min}$ vs time. Here $\Delta\theta$ is the difference between adjacent grid points. 118

Chapter 1

Introduction

There are two themes in this dissertation: (1) theoretical analysis for a fluid mixture model of tissue deformations, and (2) numerical computations using an immersed interface method for this model with interfaces. Problems with interfaces widely arise in the real world. For many applications the immersed interfaces are material interfaces, such as bubbles in water, an elastic membrane, or cells and fibrous components of tissue. Due to the differences in the properties of the different materials, the solutions to those kind of problems are non-smooth, or even discontinuous, i.e., the solutions have jumps across the interfaces. Our method, the immersed interface method, incorporating the jump conditions for the solutions into some generally used numerical methods, gives sharp resolution of the solution across the interface.

1.1 A Fluid Mixture Model of Tissue Deformations

A bio-artificial tendon development experiment, as shown in Fig. 1.1 [1], may be constructed using the linear anchor and stem configuration. It takes 14 days to show how cells consolidate the matrix and contract the gel over time. However, it is hard to understand the origins and mechanisms of cells' consolidation and gel's contraction from the experiments. Understanding the origins and mechanisms of the deformation may be important in understanding the tendon development and treatment.

Partial differential equations (PDEs) are widely used to model many phenomena in biological science to gain understanding. Those PDEs are generally very complicated due to the different scales, non-regular shapes, heterogeneous media, etc. Hence there are

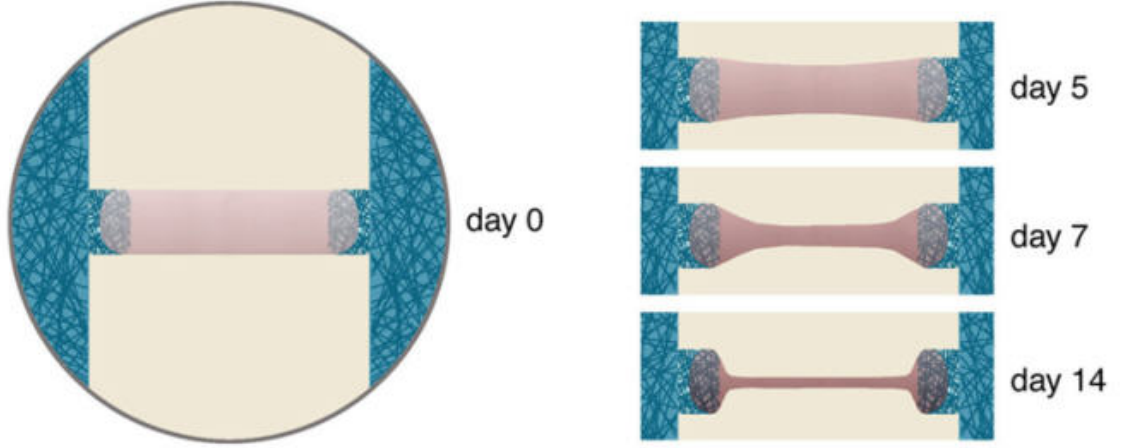


Figure 1.1: A bio-artificial tendon (BAT) may be constructed using the linear anchor and stem configuration. Cells consolidate the matrix and contract the gel over time. Garvin et al, 2003 [1]

numerical difficulties in solving them.

In our study, we consider a mathematical model (a nonlinear system with a hyperbolic equation and elliptic equations coupled together) developed in [2, 3, 4] for modeling deformations of contractile and/or growing and/or swelling mesenchymal tissues. The tissues are considered to be composed of two inter-penetrating material phases: an aqueous phase and a cell-fiber phase. The aqueous phase is composed of all the water and dissolved extracellular components of the tissues. The cell-fiber phase consists of the cells and the remaining, generally fibrous, extracellular components. It is assumed that: (1) the two phases occupy complementary portions of the space, (2) the aqueous phase behaves as a Stokes fluid, (3) the stresses in the cell-fiber phase are dissipated by permanent deformation on the relevant time scale and can also be treated as a Stokes flow. These assumptions lead to the following system of partial differential equations:

$$\begin{aligned}
 \frac{\partial \theta}{\partial t} + \nabla \cdot (\theta \underline{V}) &= 0, \\
 \nabla \cdot (\theta \underline{V} + (1 - \theta) \underline{W}) &= 0, \\
 \nabla \cdot (\theta (\lambda (\nabla \cdot \underline{V}) \underline{I} + 2\mu \underline{E})) - \theta \nabla p - \varphi \frac{\theta}{1 - \theta} (\underline{V} - \underline{W}) &= \nabla (\theta (\Upsilon - \Psi)), \\
 - (1 - \theta) \nabla p - \varphi \frac{\theta}{1 - \theta} (\underline{W} - \underline{V}) &= \nabla ((1 - \theta) \Upsilon),
 \end{aligned} \tag{1.1}$$

where θ is volume fraction of cells and fibers, \underline{V} is velocity of cell-fiber phase, \underline{W} is velocity of aqueous phase, p is interphase pressure, φ is specific drag coefficient, Ψ is the contractile stress, and Υ is the solvation stress. μ is specific shear modulus of viscosity of the cell-fiber fraction, λ is specific bulk modulus of viscosity of cell-fiber phase, $\underline{\underline{I}}$ is the identity tensor, and $\underline{\underline{E}} = \frac{1}{2}(\nabla \underline{V} + (\nabla \underline{V})^T)$ is the strain rate tensor for the cell-fiber phase. A reasonable range of dimensional and non-dimensional parameters are presented in the later chapters.

Note that $0 \leq \theta \leq 1$ and parameters φ , μ , λ , ψ , and Υ are nonnegative. Also note that these parameters may vary in space and change with the time; they may explicitly depend on θ or other variables.

The boundary conditions (BC) are given as follows

$$\begin{aligned}\underline{V}|_{\partial\Omega} &= \underline{W}|_{\partial\Omega} = 0 \\ \nabla\theta \cdot \hat{n}|_{\partial\Omega} &= 0 \\ \nabla p \cdot \hat{n}|_{\partial\Omega} &= 0\end{aligned}\tag{1.2}$$

One way to model two phases in tissues is to simply include them in the same equations and account for their different densities with θ . Thus a simple interaction between two phases can be modeled with piecewise constant initial condition,

$$\theta(\underline{x}, 0) = \begin{cases} \theta_l, & \text{if } \underline{x} \in \mathbf{B}, \mathbf{B} \subset \Omega, \\ \theta_u, & \text{if } \underline{x} \in \Omega - \mathbf{B}, \end{cases}\tag{1.3}$$

where we use θ_l for the smaller constant (lower), and θ_u for the larger constant (upper).

Note that the mathematical model for the tissue deformations is a non-linear, mixed (hyperbolic and elliptic) system of differential equations. Shock waves may develop in the time evolution. For cases with interfaces, special treatment is necessary to obtain high resolution of solutions.

1.2 A Brief Review of Literature

The theory of mixtures, which combines continuum theories for the motion and deformation of solids and fluids with general principles of chemistry, has been well applied to the study of biological tissues and cells for over three decades. These studies mainly focused on mechanics and transport of non-reacting mixtures, tissue growth and remodeling and tissue engineering [5, 6, 7, 8, 9, 10, 11].

In this field modeling itself is more important than numerical computation. Very few of these studies have carried out numerical calculations. Stastna used an explicit finite difference scheme (second order in space, first order in time) for the time-dependent case in [11]. In [3], Lubkin and Jackson used a finite difference scheme with a one-step predictor-corrector method: Lax-Wendroff method was used in the predictor and corrector steps to advect the volume fraction of cells and fibers and the fraction of cell-fiber phase which is neoplastic. Other quantities were solved via implicit differencing. They reported that upwinding gave similar results. In [12], He and Dembo used a Galerkin finite element method in cylindrical geometry with adaptive mesh. However, none of them have reported the results of the accuracy analysis for the numerical method that they used. So on the one hand, there is no way to know how accurate their approaches are; on the other hand, we do not have a lot of knowledge about efficient methods for the model problems we are working on.

However, when observing the PDEs in (1.1), we know that the system consists of an advection-like hyperbolic equation for the volume fraction of cell and fiber phase and elliptic equations for the velocities and pressure. There are many existing methods for the hyperbolic equations and elliptic equation, respectively. Finite element methods are popular among engineers, and there are many commercial packages implementing finite element methods, for instance, COMSOL, ANSYS, FLUENT, etc. However, for a moving interface or free boundary problems, the cost and limitations of possible remeshing at every time step are major concerns. Finite volume methods are widely used in solving hyperbolic equations. Finite difference methods are widely adopted in solving elliptic equations and parabolic equations. We would like to use finite difference methods on our problem.

For a linear advection equation, the upwind scheme, Lax-Friedrichs method, Lax-Wendroff method, Beam-Warming method, etc, are well used as explicit methods. Note that the Lax-Wendroff method and Beam-Warming method are both second order accurate and the dominant terms in the truncation errors depend on the third derivative of the variable, whereas the upwind scheme and Lax-Friedrichs method are both first order accurate and the dominant terms in the truncation errors depend on the second derivative of the variable. Also note that the above statement is valid only for smooth solutions. LeVeque did an informal analysis for a linear advection equation with a discontinuous solution solved by an upwind scheme [13]. He showed that the 1-norm of the error decays only like $(\Delta x)^{1/2}$ though the method is formally “first order accurate”.

Both the Lax-Wendroff and Beam-Warming methods for the linear hyperbolic system,

$$q_t + Aq_x = 0, \quad (1.4)$$

are based on the Taylor series expansion:

$$q(x, t_{n+1}) = q(x, t_n) + \Delta t q_t(x, t_n) + \frac{1}{2}(\Delta t)^2 q_{tt}(x, t_n) + \cdots. \quad (1.5)$$

From the differential equation (1.4) we have that $q_t = -Aq_x$, so $q_{tt} = -Aq_{xt} = -Aq_{tx} = -A(-Aq_x)_x A^2 q_{xx}$. Using these expressions for q_t and q_{tt} in (1.5), we have

$$q(x, t_{n+1}) = q(x, t_n) - \Delta t A q_x(x, t_n) + \frac{1}{2}(\Delta t)^2 A^2 q_{xx}(x, t_n) + \cdots. \quad (1.6)$$

Both the Lax-Wendroff and Beam-Warming methods keep only the first three terms on the right hand side and replace the spatial derivatives by the different finite difference approximations: the central finite difference for the spatial derivatives gives the Lax-Wendroff method,

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{2\Delta x} A(Q_{i+1}^n - Q_{i-1}^n) + \frac{1}{2}\left(\frac{\Delta t}{\Delta x}\right)^2 A^2(Q_{i-1}^n - 2Q_i^n + Q_{i+1}^n), \quad (1.7)$$

and the one-sided formula for the spatial derivatives gives the Beam-Warming method,

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{2\Delta x} A(3Q_i^n - 4Q_{i-1}^n + Q_{i-2}^n) + \frac{1}{2}\left(\frac{\Delta t}{\Delta x}\right)^2 A^2(Q_i^n - 2Q_{i-1}^n + Q_{i-2}^n), \quad (1.8)$$

provided that all the eigenvalues of A are positive.

LeVeque's work [13, 14] shows that excessive dissipation of the solution is evident when the upwind scheme is used for the scalar advection equation $q_t + q_x = 0$ with periodic boundary condition. His work also shows that the smooth pulse in the solution is captured much better, but the square wave pulse in the solution gives rise to an oscillatory solution when the Lax-Wendroff and Beam-Warming methods are used to solve the same advection test. The oscillations come from the approximation of the methods: the dominant error term, $q_{ttt} = -A^3 q_{xxx}$, is a dispersive term. A flux-limiter method can help keep the solution from smearing out too fast, and will significantly increase the resolution and keep discontinuities fairly sharp [15, 16, 17, 18, 19, 20].

However, we can not apply the Lax-Wendroff or Beam-Warming method to our problem. For the most simple case in one dimension, our problem can be written as

$$\begin{aligned} \theta_t + (v\theta)_x &= 0, \\ v_{xx} + (\psi\theta)_x + (\sigma \ln(1 - \theta))_x &= 0, \end{aligned} \quad (1.9)$$

where ψ is contractility coefficient and σ is the swelling number.

It is easy to obtain $\theta_t = -(v\theta)_x$. However, we do not have enough information about θ_{tt} from the system. So we can not use the Lax-Wendroff or Beam-Warming method.

The Crank-Nicolson scheme is known as an implicit second order unconditionally stable method. It is a fine method for a linear or nonlinear system of hyperbolic equations. Applying the Crank-Nicolson scheme to (1.4) gives

$$Q_i^{n+1} + \frac{\nu}{4}(Q_{i+1}^{n+1} - Q_{i-1}^{n+1}) = Q_i^n - \frac{\nu}{4}(Q_{i+1}^n - Q_{i-1}^n), \quad (1.10)$$

where $\nu = \frac{\Delta t}{\Delta x} A$, which is the Courant-Friedrichs-Lewy (or CFL) number. Consider the time evolution of a single Fourier mode of wave-number k :

$$Q_j^n = Q_0 \lambda^n e^{ikj\Delta x},$$

where n is n -th time step, j is j -th grid point and i is $\sqrt{-1}$.

A von Neumann stability analysis [21] of (1.10) yields the following expression for the amplification factor

$$\lambda = \frac{1 - i(\nu/2) \sin(k \Delta x)}{1 + i(\nu/2) \sin(k \Delta x)}.$$

Note that $|\lambda| = 1$ for all values of k regardless of the value of ν . This implies that the Crank-Nicolson scheme is not subject to the CFL constraint and there is no spurious decay in the Fourier harmonics of the solution. In other words, we would not expect the Crank-Nicolson scheme to introduce strong numerical dispersion into the advection problem. However, the price we need to pay for the high accuracy and unconditional stability of the Crank-Nicolson scheme is having to invert a tri-diagonal matrix equation at each time-step. Usually, this price is well worth paying. However, the Crank-Nicolson scheme can not be adapted to our 1D problem in (1.9) since v is nonlinearly coupled with θ .

Weighted essentially non-oscillatory (WENO) finite difference schemes, introduced by Liu et al. [22, 23] have become one of the most popular methods for solving hyperbolic conservation equations. WENO was developed from the essentially non-oscillatory (ENO) schemes using a convex combination of all candidate stencils instead of just one as in the original ENO [24, 25, 26, 22, 23, 27]. Both ENO and WENO schemes are high order accurate finite difference schemes designed for problems with piecewise smooth solutions containing discontinuities. Generally, those schemes couple with total variation diminishing (TVD) Runge-Kutta method for the time discretizations [28, 29, 30, 31, 32, 33]. However, grid refinements for error analysis published in the papers are for smooth solutions.

Recently, Henrick proposed a mapped WENO scheme [34] to achieve optimal order near critical points. Zhang and Shu proposed a new smoothness indicator for the WENO schemes to improve the post-shock oscillation [35]. Xu and Shu proposed anti-diffusive flux corrections for high order WENO schemes to improve the resolution for contact discontinuities [36]. However, those otherwise fine methods failed to achieve high order accuracies for problems with shocks. Nominally, fifth order shock capturing algorithms converge at roughly 5/6 order in the L_1 norm for discontinuities in linearly degenerate fields [37]. We applied the fifth order WENO with Roe flux for space discretization and the strong stability preserving optimal third order TVD Runge-Kutta method for time discretization to the hyperbolic equation in our system and used the second order central finite difference scheme for the elliptic equation. Similarly, we got second order accuracy for the system when we computed the cases with continuous solutions and first order accuracy (error was measured some distance (0.1) away from the interfaces) when we computed the cases with piecewise smooth solutions. Non-physical oscillations were observed in the solutions.

For an elliptic equation without interfaces, the central finite difference methods are usually adopted. Generally, the centered three-point stencil is used for a second order scheme solving for a 1D problem. Like other finite difference methods, central finite difference methods are based on Taylor series expansion. However, whenever the problem involves interfaces in the domain, special treatments are necessary to get high order solution and high resolution near the interfaces [38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48].

The interface problems always introduce numerical difficulties in computation. Currently, the smoothing method, harmonic averaging method, the immersed boundary method, the immersed interface method, the ghost fluid method, etc., are the commonly used finite difference methods for interface problems. The smoothing method is used for discontinuous coefficients with the smoothed Heaviside function or the smoothing function corresponding to the discrete hat delta function. The harmonic averaging method [48, 49, 50, 51] is another one for elliptic interface problems with discontinuous coefficients. The immersed boundary method (IBM) was originally developed by Peskin to model blood flow in a human heart. When the IBM was applied, the wall of a heart (modeled as the curve) was immersed into a Cartesian grid with discrete delta function to distribute the singular forces at the elastic boundary (the wall of the heart) to nearby grid points [52, 53, 54, 55]. The IBM has been applied to many other problems [56, 57, 58, 59, 60]. One of the most important concerns of IBM is to find the best way to discretize a Dirac delta function,

i.e., find a good discrete delta function, to distribute a singular source to nearby grid points to get an accurate solution. The IBM is simple and robust. However, the IBM can not achieve second order accuracy except in a few special situations [38]. The ghost fluid method (GFM) is another notable method for interface problems. It is simple and relatively easier to implement for compressible multi-medium flows which are generally challenging to simulate. The key idea of the GFM-based algorithm is to properly define the ghost fluids at the vicinities of the material interfaces. The original GFM was developed by Fedkiw [61, 62]. Liu, Khoo and Yeo proposed a modified GFM (MGFM) for a strong shock impacting on a material interface that the original GFM does not work consistently and efficiently [63]. However, in [64] Liu et al found that there are ranges of conditions for each type of solution where either the original GFM or the MGFM or both can not provide correct or consistent Riemann waves in one of the real fluids.

For moving interface problems, another big issue is how to track the interfaces accurately with the time. As mentioned, shock capturing methods, e.g., WENO schemes, high resolution methods, etc., are used to capture discontinuities in the solutions automatically without explicitly tracking them. However, the accuracy of shock capturing methods is poor when the discontinuities are presented across the interfaces. One of the advantages of shock capturing methods is that they are much simpler to implement. An alternative to shock capturing methods where the schemes themselves are able to handle discontinuities is shock tracking (or called front tracking) where the positions of shocks are explicitly followed to allow a different numerical treatment in smooth regions and near discontinuities. The front tracking methods track the interfaces (location of discontinuities) explicitly to obtain a sharp interface profile as time evolves. Glimm et al proposed a fully conservative front tracking algorithm for systems of nonlinear conservation laws [65]. They reported that their algorithm has $O(\Delta x)$ errors near the tracked discontinuities in the solution where the $O(1)$ errors were commonly found [66, 67]. Aslam used a level set algorithm for tracking discontinuities in hyperbolic conservation laws [37, 68]. The zero-level of a level-set function presents the geometry of the front (interface). The author's algorithm converged at fifth order in both the L_1 and L_∞ norms for the a linear advection equation and Burger's equation with convex flux functions [37]. However, the author's algorithm converged at first order, like most shock capturing schemes, for systems of conservation laws, e.g., Euler equations [68]. A level set formulation is a very useful and easy tool to present the interfaces of problems, specially for the high dimensional problems.

1.3 Our Strategy - the Immersed Interface Method

We use uniform Cartesian grids in our computations since the price paid for grid generation is almost nothing, and the conventional finite difference schemes can be used in the regions where the solution is smooth (away from the interface). Only the irregular grid points, i.e., those where there is an interface somewhere within the supporting stencil, need special treatment. Since the interface is codimension of the problem, the number of irregular grid points is usually much fewer than that of the regular ones.

The immersed interface method (IIM) was developed by LeVeque and Li [46, 38, 47] for Poisson problems with discontinuous coefficients or singular source terms giving rise to non-smooth solutions across the interfaces. Jump conditions were built into the finite difference scheme near the interfaces, hence a solution of second order accuracy was obtained. The same idea has been used for the heat equation with discontinuous heat conduction coefficients, and for Stokes flows with moving interfaces [38, 39, 69, 70, 40, 71, 72, 73]. The idea of the IIM was adapted by Zhang to hyperbolic systems with discontinuous coefficients [74], and by Lee to incompressible Navier Stokes flow with moving interface [75, 76]. Chen applied the IIM to biharmonic equations defined on irregular domains [77]. Yang [78] and Gong [79] used the IIM for elasticity problems with interfaces. Gong built the jump conditions into finite element method to achieve a second order solution for the elliptic and elasticity interface problems.

All the studies about the IIM show that the IIM is a robust approach for many types of interface problems. So we adapt this method to our particular problem.

The idea of the IIM mainly counts on Taylor expansion and builds jump conditions into the stencil (irregular grid points) near the interface. So it is very important for the IIM that we need to know the locations of interfaces and the jump conditions across the interfaces. Deriving the jump conditions is not a trivial work for many problems, especially for high dimension problems. Note that the correction for the irregular grid points depends on the main theme of finite difference scheme used for the regular grid points.

1.3.1 The Process

Now consider our simple toy problem as shown in the equation (1.9). Our strategy is simple:

- Initially, we know the distribution of θ and locations of the discontinuities of θ .

Apply the standard central finite difference method, which is second order accurate, to the elliptic equation for the velocity v at regular grid points which are away from the interfaces. Derive the jump relations across the interface and find correction terms for the irregular grid points which are near the interfaces to ensure the local truncation errors at irregular grid points to the order of $O((\Delta x)^2)$, where Δx is the mesh size. Modify the finite difference scheme at those irregular grid points with the correction terms – build the jump conditions into the standard finite difference scheme. Solve the resulting system of difference equations to obtain the numerical solution of the elliptic equation. In this way, we use the IIM to solve velocity to second order.

- Then from the computed velocity, we determine the new locations of the discontinuities by solving

$$\frac{dX}{dt} = v(X, t).$$

- We update θ piece by piece to the next time step. Apply the standard explicit second order finite difference method to the hyperbolic equation for the volume fraction θ at regular grid points. Derive the jump relations across the interfaces for the space and/or for the time whenever crossing happens as time evolves from t^n to t^{n+1} . Find the corrections for those irregular grid points. Modify the finite difference scheme at those irregular grid points with the correction terms. In this way, we use the IIM to update the volume fraction to the next time step.

- The process then is repeated till the final time for the problem.

Note that the IIM is a sharp interface method which relies on the fact that it is possible to determine jump conditions for the solution and its derivatives at the interfaces. Our goal is to achieve second order accuracy. So it is very important to find the jump conditions for the solution and its first and second derivatives.

1.3.2 The Finite Difference Schemes for the Hyperbolic Equation

The conventional finite difference methods for the hyperbolic equation, for instance, Lax-Wendroff method, Beam-Warming method, etc., are typically derived using Taylor expansions and rely on the assumption that the solution to the problem is smooth on the domain. Obviously, these approximations are not valid in the vicinity of an interface where the solution is not smooth.

Furthermore, as we discussed above, for our particular problem, some of the con-

ventional finite difference methods, e.g., Lax-Wendroff method, Beam-Warming methods, and Crank-Nicolson method, can not be applied to the hyperbolic equation in our system. Unfortunately, we can not use those two-level methods. Hence we have to use multi-level methods (see Fig. 1.2) to get a classical solution of second order. We choose the leapfrog method [80, 81, 14], which is a three-level second order method with neutral stability, for the regular grid points which are away from the interfaces:

$$\frac{\Theta_i^{n+1} - \Theta_i^{n-1}}{2\Delta t} = -\frac{(\Theta V)_{i+1}^n - (\Theta V)_{i-1}^n}{2\Delta x}. \quad (1.11)$$

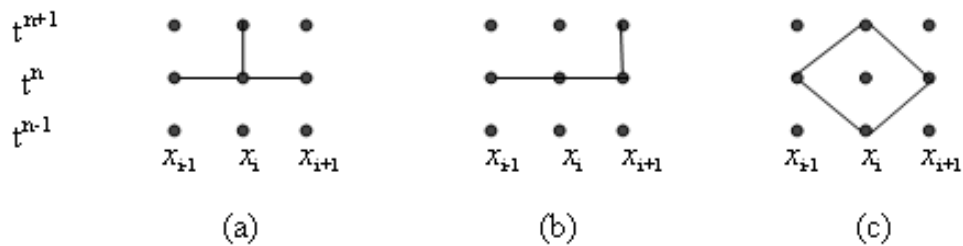


Figure 1.2: Diagram of the stencils. (a) Lax-Wendroff method; (b) Beam-Warming method; (c) Leapfrog method.

Suppose we know the location of the interface and the jumps in the solution and its derivatives at the interface. Then we can use that information to derive an algorithm for the irregular grid points to improve the accuracy of the numerical solution.

1.4 Contributions of the Dissertation

The immersed interface method has been used to solve other type of problems, such as Poisson problems, the heat equation, wave equations with fixed interface and homogeneous jump conditions, etc. This is the first known extension of this idea to a system combining hyperbolic and elliptic equations with moving interface and nonhomogeneous jump conditions, resulting in a new high resolution sharp interface method for solving a fluid mixture model of tissue deformations. The main contributions of this work are listed as follows.

- The problems analyzed in this dissertation are systems of hyperbolic and elliptic equations with four physical parameters which affect the behavior of the system. The one dimensional systems with four types of parameters have been analyzed theoretically

and numerically. We found the relationship between those parameters and the pattern of tissue deformations via linear stability analysis. Numerical experiments support our theoretical analysis. Discontinuities and non-smoothness in the solution introduce numerical difficulties.

- To maintain accuracy for the velocity v near the discontinuities, we apply the immersed interface method to the elliptic equation. In order to use the immersed interface method, we need to derive the jump condition for the elliptic equation. According to our analysis, the solution of v is non-smooth, piece-wise linear. So the jump conditions of v across the interfaces are homogeneous but the jump conditions of first derivative of v across the interfaces are nonhomogeneous.

- Comparison of two algorithms has been made for the one dimensional case. One algorithm uses Runge-Kutta with WENO for the hyperbolic equation and the central finite difference method for the elliptic equation without tracking the interfaces as time evolves. Another one uses the immersed interface method for the elliptic equation with interfaces tracked via Runge-Kutta method as time evolves. Oscillations are observed near the interfaces when the interfaces are not tracked. The immersed interface method, with the interfaces tracked, avoids non-physical oscillations, and gives high resolution solution.

- We have applied the immersed interface method to a linear advection equation with nonhomogeneous jump conditions for a moving interface. Solutions of second order resolution have been obtained.

- The immersed interface method has been applied to a one dimensional fluid mixture model of tissue deformations with all constant physical parameters. Deriving the jump conditions for both hyperbolic and elliptic equations is necessary and important. For the system with other types of physical parameters, deriving the jump conditions becomes very difficult, even impossible for some cases.

- The model system for the two dimension case is derived from the three dimension model so that the thickness of the gel is taken into account.

- The two dimensional systems with four types of parameters have been analyzed theoretically and numerically. We found the relationship between those parameters and the pattern of tissue deformations via linear stability analysis. Discontinuities and non-smoothness in the solution introduce numerical difficulties. A lot of numerical examples give high resolution numerical solutions.

1.5 Overview of the Dissertation

Chapter 2 contains the discussion about the one dimension mixture fluid model for tissue deformation. Theoretical and numerical analysis has been performed for the one dimension model with different physical parameters. The immersed interface method is used to solve the elliptic equation only in this chapter. In chapter 3, an linear scalar advection has been studied in order to clarify the application of the immersed interface method to a hyperbolic equation with moving interface. In chapter 4, the immersed interface method has been applied to the system: both the hyperbolic and elliptic equations. Chapter 5 contains the discussion about the two dimension mixture fluid model for tissue deformations. Again the theoretical and numerical analysis has been completed. The dissertation concludes in Chapter 6.

Chapter 2

WENO and IIM for a 1D Fluid Mixture Model of Tissue Deformation

2.1 The Mathematical Model

In this chapter, we consider a mathematical model developed in [2, 3, 4] for modeling deformations of contractile mesenchymal tissues. The tissues are considered to be composed of two inter-penetrating material phases: an aqueous phase and a cell-fiber phase. The aqueous phase is composed of all the water and dissolved extracellular components of the tissues. The cell-fiber phase consists of the cells and the remaining, generally fibrous, extracellular components. It is assumed that: (1) the two phases occupy complementary portions of the space, (2) the aqueous phase behaves as a Stokes fluid, (3) the stresses in the cell-fiber phase are dissipated by permanent deformation on the relevant time scale and can also be treated as a Stokes flow. These assumptions lead to the following system of partial differential equations (in 1D):

$$\frac{\partial \theta}{\partial t} + \frac{\partial(\theta v)}{\partial x} = 0, \quad 0 < x < L, \quad (2.1)$$

$$\frac{\partial}{\partial x} \left(\frac{1 - \theta}{\varphi \theta} \cdot \frac{\partial p}{\partial x} - v \right) = 0, \quad (2.2)$$

$$\frac{\partial}{\partial x} \left(2M \frac{\partial v}{\partial x} - p + \theta \psi + \sigma \ln(1 - \theta) \right) = 0, \quad (2.3)$$

where θ is the volume fraction of cells and fibers, v is the velocity of the cell-fiber phase, p is the pressure, φ is the drag coefficient, ψ is the contractility coefficient, σ is the swelling coefficient, and M is the viscosity coefficient of the cell-fiber fraction. Note that $0 < \theta < 1$ and the parameters φ , M , ψ and σ are nonnegative and can depend on time, space, and θ .

A reasonable range of dimensional and non-dimensional parameters are presented in Table 2.1; see [2, 3] for the references.

Table 2.1: Expected ranges of parameter values [2, 3].

parameter	symbol	units	range
specific drag coefficient	φ	$kg/m^3\text{-sec}$	$10^{12} - 10^{14}$
tissue viscosity	M	$kg/m\text{-sec}$	10^5
specific contractility coefficient	ψ	$kg/m\text{-sec}^2$	$10^3 - 10^4$
swelling number	σ	$kg/m\text{-sec}^2$	$10 - 10^3$
volume fraction of cell-fiber phase	θ_0	-	$\epsilon^\dagger - (1 - \epsilon^\dagger)$

\dagger ϵ is a small positive number.

The boundary conditions (BC) are given as follows

$$v(0, t) = v(L, t) = 0, \quad \frac{\partial \theta}{\partial x}(0, t) = \frac{\partial \theta}{\partial x}(L, t) = 0, \quad \frac{\partial p}{\partial x}(0, t) = \frac{\partial p}{\partial x}(L, t) = 0. \quad (2.4)$$

Two types of initial condition (IC) are considered in this study. One is with continuous data

$$\theta(x, 0) = \theta_0 + \varepsilon \cos(2\pi kx). \quad (2.5)$$

This IC is used for studying pattern-forming instabilities.

One way to model two adjacent tissues is to simply include them in the same equations and account for their different densities with θ . Thus a simple interaction between two tissues can be modeled with piecewise constant initial condition (see Fig. 2.1),

$$\theta(x, 0) = \begin{cases} \theta_l, & \text{if } 0 \leq x < x_1 \text{ or } x_2 < x \leq L, \\ \theta_u, & \text{if } x_1 \leq x \leq x_2, \end{cases} \quad (2.6)$$

where we use θ_l for the smaller constant (lower), and θ_u for the larger constant (upper).

In this chapter, we will focus on simulating tissue deformations numerically for the one dimensional model. Note that the mathematical model is a non-linear, mixed (hyperbolic and elliptic) system of differential equations. Shock waves will be developed in the time evolution process. The purpose of this research is to investigate the stability of the equilibria, and to verify stability and accuracy of our numerical methods.

We will present two numerical methods to solve the one-dimensional system. The first one is to use high order numerical methods such as WENO (weighted essentially non-oscillatory) schemes, for example [32, 31, 33] for the hyperbolic equation and the standard central finite difference scheme for the elliptic equation. In the second approach, we use the immersed interface method to track the shock waves, which avoids non-physical oscillations.

2.2 The Linear Stability Analysis

There are four physical coefficients, φ , M , ψ and σ , in the system (2.1)-(2.3). In order to predict the overall behavior of the solution, it is necessary to carry out linear stability analysis.

An obvious steady state of the model, which can be easily verified, is the following

$$\begin{bmatrix} \theta \\ v \\ p \end{bmatrix} = \begin{bmatrix} \theta_0 \\ v_0 \\ p_0 \end{bmatrix}, \quad (2.7)$$

where θ_0 is a constant. For simplicity, we will take $v_0 = p_0 = 0$. We have the following theorem about the stability of the equilibrium.

Theorem 1: *Assume that the parameters φ , M , ψ and σ in (2.1)-(2.3) only depend on θ . Then the solution (2.7) of the system (2.1)-(2.3) is unstable if*

$$\psi + \frac{\partial \psi}{\partial \theta} \theta + \frac{\partial \sigma}{\partial \theta} \ln(1 - \theta) > \frac{\sigma}{1 - \theta}, \quad (2.8)$$

where the functions ψ , σ , and their derivatives are evaluated at θ_0 .

Sketch of the proof. We consider small perturbations of the form

$$\begin{bmatrix} \theta \\ v \\ p \end{bmatrix} = \begin{bmatrix} \theta_0 \\ 0 \\ 0 \end{bmatrix} + \varepsilon \begin{bmatrix} \theta_1 \\ v_1 \\ p_1 \end{bmatrix}, \quad 0 < \varepsilon \ll 1. \quad (2.9)$$

Plugging it into (2.1), ignoring the higher order term $O(\varepsilon^2)$, we get the simplified form for the $O(\varepsilon)$ term

$$\frac{\partial \theta_1}{\partial t} + \theta_0 \frac{\partial(v_1)}{\partial x} = 0 \quad (2.10)$$

Expanding the term of $\frac{1-\theta}{\varphi\theta}$ via Taylor expansion, we get

$$\frac{1 - \theta}{\varphi\theta} = \frac{1 - \theta_0}{\varphi_0\theta_0} + O(\varepsilon),$$

where $\varphi_0 = \varphi(\theta_0)$. Substituting the above equation along with (2.9) into (2.2) and ignoring the higher order terms, we get the $O(\varepsilon)$ terms

$$\frac{1 - \theta_0}{\varphi_0 \theta_0} \frac{\partial^2 p_1}{\partial x^2} - \frac{\partial v_1}{\partial x} = 0. \quad (2.11)$$

Now expanding the term of $\ln(1 - \theta)$ in (2.3) via Taylor Expansion, we get

$$\ln(1 - \theta) = \ln(1 - \theta_0) - \frac{1}{1 - \theta_0} \varepsilon \theta_1 + O(\varepsilon^2).$$

Similarly, expanding the parameters at the steady state via Taylor expansion, we get

$$M(\theta) = M_0 + \varepsilon \theta_1 M'(\theta_0) + O(\varepsilon^2),$$

$$\psi(\theta) = \psi_0 + \varepsilon \theta_1 \psi'(\theta_0) + O(\varepsilon^2),$$

$$\sigma(\theta) = \sigma_0 + \varepsilon \theta_1 \sigma'(\theta_0) + O(\varepsilon^2),$$

where $F_0 = F(\theta_0)$, and $F'(\theta_0) = \frac{dF}{d\theta}(\theta_0)$, in which F indicates M , ψ , and σ respectively.

Substituting the above equations along with (2.9) into (2.3) and ignoring the higher order terms, we get the $O(\varepsilon)$ terms

$$\begin{aligned} 2M_0 \frac{\partial^2 v_1}{\partial x^2} - \frac{\partial p_1}{\partial x} + \psi_0 \frac{\partial \theta_1}{\partial x} + \theta_0 \psi'(\theta_0) \frac{\partial \theta_1}{\partial x} \\ + \ln(1 - \theta_0) \sigma'(\theta_0) \frac{\partial \theta_1}{\partial x} - \frac{\sigma_0}{1 - \theta_0} \frac{\partial \theta_1}{\partial x} = 0. \end{aligned} \quad (2.12)$$

Now we assume

$$\begin{bmatrix} \theta_1 \\ v_1 \\ p_1 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} e^{\lambda t + i k x}, \quad (2.13)$$

where c_1 , c_2 and c_3 are constants and k is a typical Fourier mode.

Substituting (2.13) into (2.10), (2.11) and (2.12), we obtain the following linearized system

$$\begin{bmatrix} \lambda & i k \theta_0 & 0 \\ 0 & -i k & A \\ B & C & -i k \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad (2.14)$$

where

$$\begin{aligned} A &= -k^2 \frac{1 - \theta_0}{\varphi_0 \theta_0}, \\ B &= i k (\psi_0 + \theta_0 \psi'(\theta_0) + \sigma'(\theta_0) \ln(1 - \theta_0) - \frac{\sigma_0}{1 - \theta_0}), \\ C &= -2M_0 k^2. \end{aligned} \quad (2.15)$$

To obtain non-trivial solutions to above system, λ needs to satisfy

$$\lambda = \frac{ikAB\theta_0}{k^2 + AC}. \quad (2.16)$$

It follows that there exist infinite solutions to (2.14)

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} -\frac{\theta_0 A}{\lambda} \\ -\frac{iA}{k} \\ 1 \end{bmatrix} c_3 \quad (2.17)$$

Hence,

$$\begin{bmatrix} \theta_1 \\ v_1 \\ p_1 \end{bmatrix} = c_3 e^{\lambda t} \left(\begin{bmatrix} -\frac{\theta_0 A}{\lambda} \cos(kx) \\ \frac{A}{k} \sin(kx) \\ \cos(kx) \end{bmatrix} + i \begin{bmatrix} -\frac{\theta_0 A}{\lambda} \sin(kx) \\ -\frac{A}{k} \cos(kx) \\ + \sin(kx) \end{bmatrix} \right) \quad (2.18)$$

Substituting (2.15) into (2.16), we obtain

$$\lambda = \frac{k^2 \frac{1-\theta_0}{\varphi_0} \left(\psi_0 + \psi'(\theta_0)\theta_0 + \sigma'(\theta_0) \ln(1-\theta_0) - \frac{\sigma_0}{1-\theta_0} \right)}{1 + 2k^2 M_0 \frac{1-\theta_0}{\varphi_0 \theta_0}} \quad (2.19)$$

Note that λ is a function of k^2 , and it is a real number and bounded. The system is unstable if $\lambda > 0$, which is true when

$$\psi_0 + \psi'(\theta_0)\theta_0 + \sigma'(\theta_0) \ln(1-\theta_0) > \frac{\sigma_0}{1-\theta_0}. \quad (2.20)$$

To analyze the system further, we need to choose specific ψ and σ . To facilitate testing our method on various functional forms, we nondimensionalized the system using length scale L of 1 *cm* and time scale 1 hour. We analyzed and simulated several choices of the parameters listed in Table 2.2.

Table 2.2: Choices of the nondimensionalized functions M , φ , ψ , and σ .

Models	M	φ	ψ	σ
1	$\frac{\theta}{0.5+\theta}$	1.0	θ	$1-\theta$
2	0.5	ϵ^\dagger	C_1	C_2
3	$0.5e^\theta$	e^θ	C_1	C_2
4	$0.5e^\theta$	e^θ	$1.8e^{-\theta}$	$e^{-\theta}$

$\dagger \epsilon$ is a small positive number.

Note that the stability condition (2.20) depends on θ_0 , ψ , σ , and derivatives of ψ and σ with respect to θ . Hence we can treat Model 2 and 3 in the same way since in both

models ψ and σ are constants. In these two models, λ can be expressed as

$$\lambda = \frac{k^2 \frac{1-\theta_0}{\varphi_0} (\psi_0 - \frac{\sigma_0}{1-\theta_0})}{1 + 2k^2 M_0 \frac{1-\theta_0}{\varphi_0 \theta_0}}. \quad (2.21)$$

The system is unstable for all k if (2.20), which is reduced to

$$\psi_0 > \frac{\sigma_0}{1 - \theta_0},$$

that is, when contractility outweighs swelling, instability results.

We are also interested in the cases where the initial θ has a jump discontinuity as plotted in Fig. 2.1, in which we take $L = 1$ for simplicity of the discussion and thereafter. In general, the tissue will swell or contract to an equilibrium state. For such an initial θ , we can determine ψ that makes the system be an equilibrium state. Note that $\frac{\partial \theta}{\partial t} = 0$ but θ_x does not exist at the discontinuities. We use $[\theta] = \lim_{x \rightarrow \alpha^+} \theta(x) - \lim_{x \rightarrow \alpha^-} \theta(x) = \theta^+ - \theta^-$ to denote the jump of θ at $x = \alpha$. For Model 2, it is easy to get the relation of the parameters in the equilibrium,

$$-\psi[\theta] - \sigma[\ln(1 - \theta)] = 0. \quad (2.22)$$

Hence,

$$\psi = -\frac{\sigma[\ln(1 - \theta)]}{[\theta]} = -\frac{\sigma(\ln(1 - \theta_u) - \ln(1 - \theta_l))}{\theta_u - \theta_l}. \quad (2.23)$$

For the other models, it is much more complicated to find such relation of the parameters corresponding to equilibrium state.

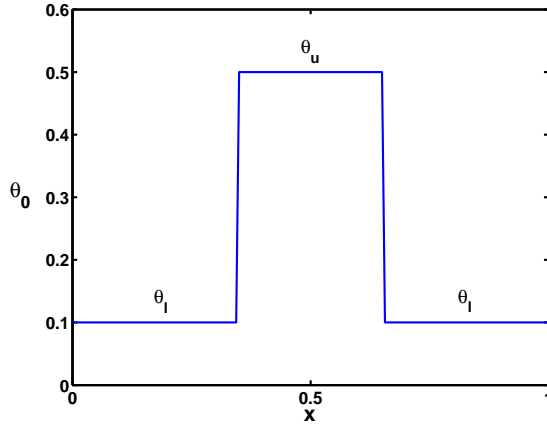


Figure 2.1: A typical initial data of θ .

When $\psi = \frac{\sigma}{1-\theta_0}$, we conclude that $\lambda = 0$ indicating a neutral stability. Fig. 2.2 shows the plot of the values ψ determined by (2.23), $\psi(\theta_l) = \frac{\sigma}{1-\theta_l}$, and $\psi(\theta_u) = \frac{\sigma}{1-\theta_u}$ with $\sigma = 1$, and $\theta_l = 0.1$. Fig. 2.2 indicates that for the given σ and θ_l , the system will be at

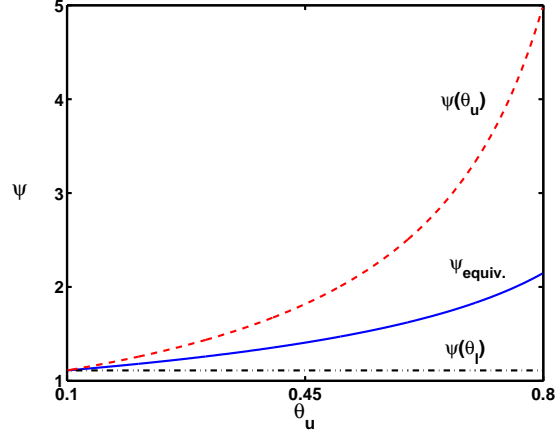


Figure 2.2: Plot of stability regions of system with $\sigma = 1$, $\psi = \frac{\sigma}{1-\theta}$, $\psi_{equiv.}$ determined by (2.23), and $\theta_l = 0.1$. The dashed line and dash-dot line indicate the ψ with which the system is neutrally stable for θ_u and θ_l , respectively; the solid line indicates the ψ value with which the system is at equilibrium.

the steady state if one chooses ψ and θ_u along the solid curve; the system will be stable if one chooses ψ under the dash-dot horizontal line; the system will be unstable if one chooses ψ above the dashed curve; and the system will be un-settled with the solution near θ_l appearing to be unstable and the solution near θ_u appearing to be stable if one chooses ψ and θ_u from the region under the dashed curve and above the dash-dot horizontal line. We will call this region an oscillation region. Note that the steady state curve always lies between $\psi(\theta_l)$ and $\psi(\theta_u)$. This implies that any perturbation around the steady state with the piecewise constant initial θ will be depressed at the middle part (the solution near θ_u) but the solution will grow near the two ends (the solution near θ_l).

We demonstrate the stability analysis in Fig. 2.3 with the following initial condition

$$\theta(x, 0) = \begin{cases} \theta_l + \varepsilon \cos(kx), & \text{if } 0 \leq x < x_1 \text{ or } x_2 < x \leq 1, \\ \theta_u + \varepsilon \cos(kx), & \text{if } x_1 \leq x \leq x_2, \end{cases} \quad (2.24)$$

The parameters are $\sigma = 1.0$, $\theta_u = 0.5$, $\theta_l = 0.1$, $\psi = 1.47$, $\varepsilon = 0.05$, $k = 60\pi$, $x_1 = 0.4$, and $x_2 = 0.6$.

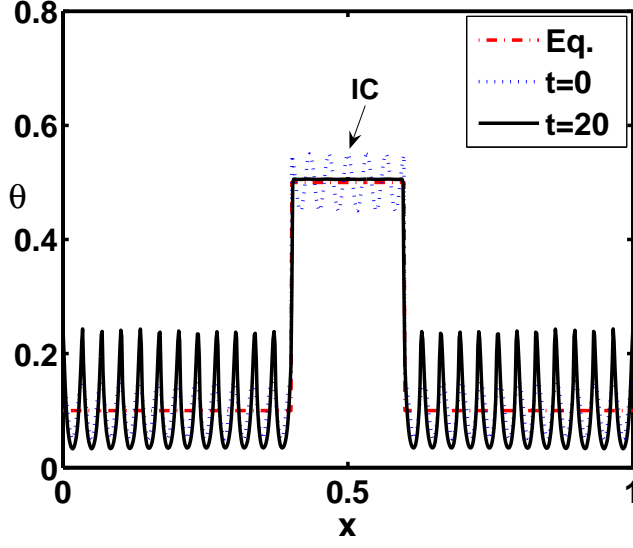
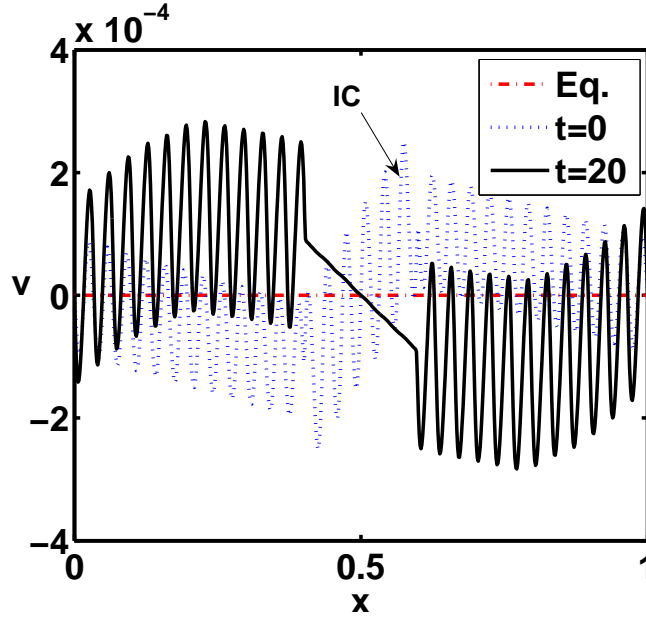
(a) Solution for θ (b) Solution for v

Figure 2.3: Solution plots with the perturbed initial data (2.24) with $M = 0.5$, $\varphi = 0$, $\sigma = 1$, $\psi = 1.47$, $\theta_u = 0.5$, $\theta_l = 0.1$, $\varepsilon = 0.05$, $k = 60\pi$, $x_1 = 0.4$, and $x_2 = 0.6$. In both plots, the dash-dot line shows the steady state; the dotted one shows the initial data and the solid one shows the solution at $t = 20$.

In Fig. 2.3, the dash-dot line in the plot shows the steady state solution; the dotted line shows the initial data and the solid line shows the solution at $t = 20$. As discussed above, the perturbation middle part decays in time and approaches steady state, and the perturbation near the two ends grows with time.

For Models 1 and 4 of parameters, we still can find the eigenvalue relations for the stability analysis, but it is difficult to find the exact steady state solutions in terms of the parameters.

The parameters for Model 1 are

$$\psi = \theta, \quad \sigma = 1 - \theta. \quad (2.25)$$

We have derived that

$$\lambda = \frac{k^2 \frac{1-\theta_0}{\varphi_0} (2\theta_0 - \ln(1 - \theta_0) - 1)}{1 + 2k^2 M_0 \frac{1-\theta_0}{\varphi_0 \theta_0}}. \quad (2.26)$$

The system is unstable if (2.20), which is reduced to

$$2\theta_0 - \ln(1 - \theta_0) - 1 > 0$$

in this case. Using the MATLAB built-in function *fzero*, it is easy to find that $\lambda > 0$ when $\theta_0 > 0.31$.

Suppose the frequency, k , is nonzero. Then rewrite λ in the following form

$$\lambda = \frac{\frac{1-\theta_0}{\varphi_0} (2\theta_0 - \ln(1 - \theta_0) - 1)}{\frac{1}{k^2} + 2M_0 \frac{1-\theta_0}{\varphi_0 \theta_0}}$$

Obviously, $\lambda \rightarrow C(\theta_0)$ as $k \rightarrow \infty$, where $C(\theta_0)$ is a constant for a given θ_0 . Suppose M adopts the form of $M = \frac{a_1 \theta}{a_2 + \theta}$, where a_1 and a_2 are constants. Choose $a_1 = 1$, $a_2 = 0.5$ and $\varphi_0 = 1$ during the numerical calculation.

Now consider the range of solution for θ , which by definition has $0 \leq \theta \leq 1$.

$$\theta = \theta_0 + \varepsilon c_3 \frac{1 + 2M_0 k^2 \frac{1-\theta_0}{\varphi_0 \theta_0}}{2\theta_0 - \ln(1 - \theta_0) - 1} e^{\lambda t} (\cos(kx) + i \sin(kx)).$$

Hence

$$Re(\theta) = \theta_0 + \varepsilon c_3 \frac{1 + 2M_0 k^2 \frac{1-\theta_0}{\varphi_0 \theta_0}}{2\theta_0 - \ln(1 - \theta_0) - 1} e^{\lambda t} \cos(kx).$$

The condition on θ implies that

$$0 \leq \theta_0 + \varepsilon c_3 \frac{1 + 2M_0 k^2 \frac{1-\theta_0}{\varphi_0 \theta_0}}{2\theta_0 - \ln(1 - \theta_0) - 1} e^{\lambda t} \cos(kx) \leq 1.$$

Consider the initial possible k , $t = 0$,

$$0 \leq \theta_0 + \varepsilon c_3 \frac{1 + 2M_0 k^2 \frac{1-\theta_0}{\varphi_0 \theta_0}}{2\theta_0 - \ln(1-\theta_0) - 1} \cos(kx) \leq 1.$$

Since $\frac{1+2M_0 k^2 \frac{1-\theta_0}{\varphi_0 \theta_0}}{2\theta_0 - \ln(1-\theta_0) - 1} > 0$, $\cos(kx) \in [-1, 1]$, and choosing $c_3 = 1$, we have

$$\max(\theta) = \theta_0 + \varepsilon \frac{1 + 2M_0 k^2 \frac{1-\theta_0}{\varphi_0 \theta_0}}{2\theta_0 - \ln(1-\theta_0) - 1} \leq 1,$$

and

$$\min(\theta) = \theta_0 - \varepsilon \frac{1 + 2M_0 k^2 \frac{1-\theta_0}{\varphi_0 \theta_0}}{2\theta_0 - \ln(1-\theta_0) - 1} \geq 0,$$

i.e.,

$$k^2 \leq \frac{\frac{1}{\varepsilon}(1-\theta_0)(2\theta_0 - \ln(1-\theta_0) - 1) - 1}{2M_0 \frac{1-\theta_0}{\varphi_0 \theta_0}} = \kappa_{10},$$

and

$$k^2 \leq \frac{\frac{1}{\varepsilon}\theta_0(2\theta_0 - \ln(1-\theta_0) - 1) - 1}{2M_0 \frac{1-\theta_0}{\varphi_0 \theta_0}} = \kappa_{20}.$$

Hence, we obtain the initial possible maximum k as a function of the initial steady state value θ_0

$$k \leq \sqrt{\min(\kappa_{10}, \kappa_{20})}. \quad (2.27)$$

Fig. 2.4 shows the initial possible maximum k for the different initial steady state θ_0 with $\varepsilon = 0.001$. Note that large k will induce very large condition numbers for the coefficient matrices for v and p which cause the numerical methods difficulty in solving the system.

The parameters for Model 4 are

$$M = 0.5e^\theta; \quad \varphi = e^\theta; \quad \psi = 1.8e^{-\theta}; \quad \sigma = e^{-\theta}. \quad (2.28)$$

We have shown that λ can be expressed as

$$\lambda = \frac{k^2 \theta_0 (1 - \theta_0) e^{-2\theta_0} f(\theta_0)}{\theta_0 + k^2 (1 - \theta_0)}, \quad (2.29)$$

where

$$f(\theta_0) = 1.8(1 - \theta_0) - \ln(1 - \theta_0) - \frac{1}{1 - \theta_0}. \quad (2.30)$$

Note that the sign of λ depends on the sign of $f(\theta_0)$. Hence, the system is unstable if (2.20), which is reduced to

$$1.8(1 - \theta_0) - \ln(1 - \theta_0) - \frac{1}{1 - \theta_0} > 0.$$

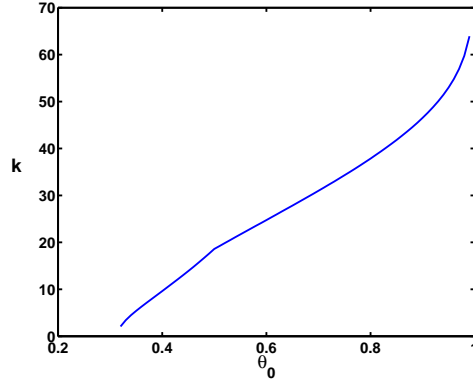


Figure 2.4: Maximum frequency evaluated by (2.27) with $M = \frac{\theta}{0.5+\theta}$, $\varphi = 1$, $\psi = \theta$, $\sigma = 1 - \theta$, and $\varepsilon = 0.001$.

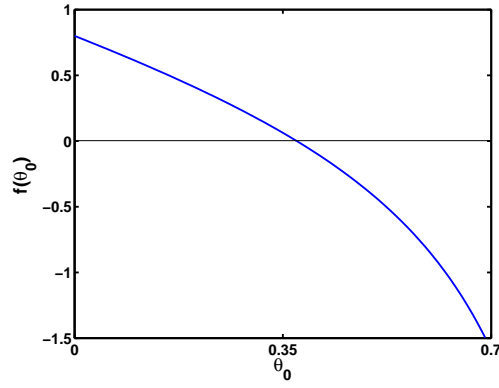


Figure 2.5: Diagram indicates the sign of λ evaluated by $1.8(1 - \theta_0) - \ln(1 - \theta_0) - \frac{1}{1-\theta_0}$ for the system with $M = 0.5e^\theta$; $\varphi = e^\theta$, $\psi = 1.8e^{-\theta}$ and $\sigma = e^{-\theta}$.

The left hand side of the inequality is plotted in Fig. 2.5 against θ . For smaller θ , say, $\theta < 0.37$, we have $\lambda > 0$, and the linearized system is unstable. For larger θ , we have $\lambda < 0$, and the linearized system is stable and will approach the equilibrium. For the piecewise constant initial condition with $\theta_l < 0.37$ and $\theta_u > 0.37$, the situation is the same as in Model 2 and 3. The solution is unstable near the ends but stable in the middle. This presents a challenge for numerical computations.

2.3 Numerical Methods

From (2.2), one can obtain

$$\frac{1 - \theta}{\varphi\theta} p_x - v = c,$$

where c is a constant. Applying the boundary conditions for pressure and velocity, we have $c = 0$, i.e.,

$$\frac{1 - \theta}{\varphi\theta} p_x = v.$$

We can eliminate p from the system (2.1)-(2.3) to get

$$\frac{\partial\theta}{\partial t} + \frac{\partial(\theta v)}{\partial x} = 0, \quad (2.31)$$

$$\frac{\partial}{\partial x} \left(2M \frac{\partial v}{\partial x} \right) - \frac{\varphi\theta}{1 - \theta} v + \frac{\partial(\theta\psi)}{\partial x} + \frac{\partial(\sigma \ln(1 - \theta))}{\partial x} = 0. \quad (2.32)$$

Our numerical computation is based on the above simplified system. Finite difference methods (FDM) are employed to solve the system. A Runge-Kutta method is used for time discretization. The upwind scheme or WENO scheme is used for spatial discretization for the hyperbolic equation. We describe two approaches. The first approach assumes no knowledge of the shock locations. The second approach is the immersed interface method that uses the jump conditions to solve the differential equations to second order accuracy assuming knowledge of the locations of the discontinuities.

2.3.1 The Upwind Scheme and Central FDM

The finite difference method(FDM), upwind scheme, has been employed to solve for θ . This is the 1st order scheme. We apply this scheme to the PDE for θ

$$\Theta_j^{n+1} = \begin{cases} \Theta_j^n - \frac{\Delta t}{h} (\Theta_j^n V_j^n - \Theta_{j-1}^n V_{j-1}^n), & \text{if } V_j^n > 0, \\ \Theta_j^n - \frac{\Delta t}{h} (\Theta_{j+1}^n V_{j+1}^n - \Theta_j^n V_j^n), & \text{if } V_j^n < 0, \end{cases} \quad (2.33)$$

where Θ_j^n is the numerical solution of volume fraction at time step n and mesh grid j , V_j^n is the numerical solution of velocity at time step n and mesh grid j , h is the mesh size and Δt is the time step size.

The upwind scheme is a kind of method which smooths out the solution. After a long-time run, the numerical solution via the upwind scheme will be totally different from the exact solution for a linear shock propagation problem [13]. For our problem, the upwind scheme works fine for the case with continuous data, but oscillation occurs at later time near discontinuous points for the cases with parameters in the oscillation region.

The central finite difference method, known as 2^{nd} order scheme, has been used to solve for velocity. We apply this scheme to (2.32)

$$\begin{aligned} & \frac{2M_{j-1/2}V_{j-1} - 2(M_{j-1/2} + M_{j+1/2})V_j + 2M_{j+1/2}V_{j+1}}{(\Delta x)^2} - \frac{\varphi_j\Theta_j}{1-\Theta_j}V_j \\ &= -\frac{\Theta_{j+1}\psi_{j+1} - \Theta_{j-1}\psi_{j-1}}{2\Delta x} - \frac{\sigma_{j+1}\ln(1-\Theta_{j+1}) - \sigma_{j-1}\ln(1-\Theta_{j-1})}{2\Delta x} \end{aligned} \quad (2.34)$$

2.3.2 The WENO-Roe Scheme

Weighted essentially non-oscillatory (WENO) finite difference schemes, introduced by Liu et al. [22, 23] have become one of the most popular methods for solving the hyperbolic conservation equations. For the hyperbolic equation (2.31), we use the fifth order WENO scheme with Roe flux (WENO-Roe) for the spatial discretization, combined with third order TVD (total variation diminishing) Runge-Kutta method [30, 31, 32, 33] for the time discretization.

We use a uniform grid

$$a = x_0 < x_1 < \cdots < x_{N-1} < x_N = L \quad (2.35)$$

and define

$$x_{i+\frac{1}{2}} = \frac{x_i + x_{i+1}}{2}, \quad i = 0, 1, \dots, N-1. \quad (2.36)$$

The conservative approximation to the spatial derivative is applied directly to (2.31)

$$\frac{d\theta_i(t)}{dt} = -\frac{1}{\Delta x}(\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}), \quad (2.37)$$

where $\theta_i(t)$ is the numerical approximation to $\theta(x_i, t)$, and $\hat{f}_{i+\frac{1}{2}}$ is the numerical flux. Let $u(x) = f(\theta(x, t)) = \theta v$. Let the finite difference stencils be

$$S_r(i) = \{x_{i-r}, \dots, x_{i-r+R-1}\}, \quad r = 0, \dots, R-1. \quad (2.38)$$

We can get $2R$ different ENO reconstructions to form the upwind biased R th order approximations to the values $u_{i+\frac{1}{2}}^{(r)}$ and $u_{i-\frac{1}{2}}^{(r)}$, that is,

$$u_{i+\frac{1}{2}}^{(r)} = \sum_{j=0}^{R-1} c_{rj} u_{i-r+j}, \quad u_{i-\frac{1}{2}}^{(r)} = \sum_{j=0}^{R-1} \tilde{c}_{rj} u_{i-r+j}, \quad r = 0, \dots, R-1. \quad (2.39)$$

Note that

$$\tilde{c}_{rj} = c_{r-1,j},$$

where

$$c_{rj} = \sum_{m=j+1}^R \frac{\sum_{l=0, l \neq m}^R \prod_{q=0, q \neq m, l}^R (r - q + l)}{\prod_{l=0, l \neq m}^R (m - l)} \quad (2.40)$$

for a uniform grid. We list the constants c_{rj} for $R = 2$ and 3 in Table 2.3.

Table 2.3: The constants c_{rj} .

R	r	$j = 0$	$j = 1$	$j = 2$
	-1	3/2	-1/2	
2	0	1/2	1/2	
	1	-1/2	3/2	
	-1	11/6	-7/6	1/3
3	0	1/3	5/6	-1/6
	1	-1/6	5/6	1/3
	2	1/3	-7/6	11/6

The WENO reconstruction will take a convex combination of all $u_{i+\frac{1}{2}}^{(r)}$ and $u_{i-\frac{1}{2}}^{(r)}$ defined in (2.39), respectively, as the new approximations to $u(x_{i+\frac{1}{2}})$ and $u(x_{i-\frac{1}{2}})$ with $(2R - 1)$ th order accuracy:

$$u_{i+\frac{1}{2}}^- = \sum_{r=0}^{R-1} \omega_r u_{i+\frac{1}{2}}^{(r)}, \quad u_{i-\frac{1}{2}}^+ = \sum_{r=0}^{R-1} \tilde{\omega}_r u_{i-\frac{1}{2}}^{(r)}, \quad (2.41)$$

where the weights ω_r and $\tilde{\omega}_r$ are defined as

$$\omega_r = \frac{\alpha_r}{\sum_{s=0}^{R-1} \alpha_s}, \quad \tilde{\omega}_r = \frac{\tilde{\alpha}_r}{\sum_{s=0}^{R-1} \tilde{\alpha}_s}, \quad r = 0, \dots, R-1, \quad (2.42)$$

with

$$\alpha_r = \frac{d_r}{(\epsilon + \beta_r)^2}, \quad \tilde{\alpha}_r = \frac{\tilde{d}_r}{(\epsilon + \beta_r)^2}. \quad (2.43)$$

The values of d_r are given by

$$\begin{aligned} d_0 &= \frac{2}{3}, \quad d_1 = \frac{1}{3}, \quad R = 2; \\ d_0 &= \frac{3}{10}, \quad d_1 = \frac{3}{5}, \quad d_2 = \frac{1}{10}, \quad R = 3, \end{aligned}$$

and

$$\tilde{d}_r = d_{R-1-r}$$

from the symmetry. We choose the parameter $\epsilon = 10^{-6}$ in all the numerical tests. The so-called “smooth indicators”, β_r , of the stencil $S_r(i)$ are given as follows

$$\begin{aligned} \beta_0 &= (u_{i+1} - u_i)^2, \\ \beta_1 &= (u_i - u_{i-1})^2 \end{aligned} \tag{2.44}$$

for $R = 2$, and

$$\begin{aligned} \beta_0 &= \frac{13}{12}(u_i - 2u_{i+1} + u_{i+2})^2 + \frac{1}{4}(3u_i - 4u_{i+1} + u_{i+2})^2, \\ \beta_1 &= \frac{13}{12}(u_{i-1} - 2u_i + u_{i+1})^2 + \frac{1}{4}(u_{i-1} - u_{i+1})^2, \\ \beta_2 &= \frac{13}{12}(u_i - 2u_{i-1} + u_{i-2})^2 + \frac{1}{4}(3u_i - 4u_{i-1} + u_{i-2})^2 \end{aligned} \tag{2.45}$$

for $R = 3$.

Once the numerical fluxes $f_{i+\frac{1}{2}}$ are obtained by the WENO reconstruction procedures, the upwinding scheme is used in constructing the flux for stability. The Roe flux is applied:

$$\hat{f}_{i+\frac{1}{2}} = \begin{cases} u_{i+\frac{1}{2}}^-, & \text{if } a_{i+\frac{1}{2}} \geq 0, \\ u_{i+\frac{1}{2}}^+, & \text{if } a_{i+\frac{1}{2}} < 0, \end{cases} \tag{2.46}$$

where $a_{i+\frac{1}{2}}$ is the Roe speed at $x_{i+\frac{1}{2}}$ defined as

$$a_{i+\frac{1}{2}} \equiv \frac{f(\theta_{i+1}) - f(\theta_i)}{\theta_{i+1} - \theta_i}. \tag{2.47}$$

In recent years, Henrick proposed a mapped WENO scheme [34] to achieve optimal order near critical points. Zhang and Shu proposed a new smoothness indicator for the WENO schemes to improve the post-shock oscillation [35]. Xu and Shu proposed anti-diffusive flux corrections for high order WENO schemes to improve the resolution for contact discontinuities [36]. However, those methods failed to achieve high order accuracies for problems with shocks. Nominally, fifth order shock capturing algorithms converge at roughly 5/6 order in the L_1 norm for discontinuities in linearly degenerate fields [37].

2.3.3 The Time Discretization Using the TVD Runge-Kutta Method

Now considering time discretization, we rewrite (2.37) as

$$\theta_t = L(\theta), \quad (2.48)$$

where $L(\theta)$ is the WENO approximation to the derivative $-(\theta v)_x$ in the PDE (2.31). The optimal strong stability preserving third order total variation diminishing (TVD) Runge-Kutta method [82, 83, 84] is employed:

$$\begin{aligned} \theta^{(1)} &= \theta^n + \Delta t L(\theta^n), \\ \theta^{(2)} &= \frac{3}{4}\theta^n + \frac{1}{4}\theta^{(1)} + \frac{1}{4}\Delta t L(\theta^{(1)}), \\ \theta^{n+1} &= \frac{1}{3}\theta^n + \frac{2}{3}\theta^{(2)} + \frac{2}{3}\Delta t L(\theta^{(2)}), \end{aligned} \quad (2.49)$$

with CFL (Courant-Friedrichs-Levy) coefficient $c = 1$.

Verification of the numerical method

We first test the numerical method using smooth solutions to check the order of the accuracy. The set-up of the test example is the following:

$$\begin{aligned} \theta_t + (v\theta)_x &= 0, \\ v_{xx} + (\psi\theta)_x + (\sigma \ln(1 - \theta))_x &= 0, \\ \theta(0, t) &= \theta(1, t), v(0, t) = v(1, t), \\ \theta_0(x) &= \frac{1}{2} + \frac{1}{8} \cos(2\pi x), \end{aligned} \quad (2.50)$$

where $\psi = 1.8$ and $\sigma = 1.0$. Since we do not know the exact solution of the system, we compare the computed results against the one computed from the finest grid which is $N = 1280$. The justification of such analysis can be found in [38].

Tables 2.4 and 2.5 list the grid refinement analysis against the solution computed from the finest grid. In the tables, the error ratio is defined as

$$\text{Ratio} = \frac{\|U(2h) - U_*\|}{\|U(h) - U_*\|},$$

where U stands for θ or v , and U_* is the solution computed from the finest grid. Tables 2.4 and 2.5 show the ratios for θ and v at $t = 0.05$, respectively. The ratio approaches the number 5 indicating second order accuracy, while the number 3 would indicate first order accuracy (see the justification in [38]). We can see that the proposed method is second

order accurate in both the L_∞ and L_1 norms at $t = 0.05$ for smooth solutions before any emerging shocks. The linear regression analysis of the convergence order in log-log scale for the test problem with continuous initial condition at $t = 0.05$ is shown in Figs. 2.6 and 2.7 for θ and v , respectively. The average convergence order for the L_∞ and L_1 errors are 2.01 and 2.11 for θ , and 2.04 for v , respectively.

Table 2.4: Accuracy on θ in the System with $\theta_0(x) = \frac{1}{2} + \frac{1}{8} \cos(2\pi x)$.

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio
10	7.98e-5	—	5.37e-5	—
20	2.73e-5	2.93	1.02e-5	5.26
40	6.48e-6	4.21	2.50e-6	4.08
80	1.65e-6	3.93	6.08e-7	4.12
160	4.11e-7	4.01	1.48e-7	4.12
320	9.82e-8	4.19	3.49e-8	4.24
640	1.96e-8	5.00	6.94e-9	5.02

Table 2.5: Accuracy on v in the System with $\theta_0(x) = \frac{1}{2} + \frac{1}{8} \cos(2\pi x)$.

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio
10	2.65e-4	—	1.16e-4	—
20	6.97e-5	3.81	3.02e-5	3.84
40	1.74e-5	4.01	7.54e-6	4.00
80	4.37e-6	3.98	1.89e-6	4.00
160	1.08e-6	4.05	4.66e-7	4.05
320	2.57e-7	4.20	1.11e-7	4.20
640	5.13e-8	5.00	2.22e-8	5.00

When the initial data is piecewise constant, that is, with shock waves present, as in the standard approach, we measure the errors conventionally at some distance (0.1 in our test case) from the discontinuities. Tables 2.6 and 2.7 show the grid refinement results ($t = 0.05$) for θ and v with the following initial data:

$$\theta_0(x) = \begin{cases} 0.5, & \text{if } 0.3 \leq x \leq 0.7, \\ 0.1, & \text{if } x < 0.3 \text{ or } x > 0.7. \end{cases}$$

We obtained first order convergence for θ and v . This is due to the delta function singularity from $\frac{\partial(\theta\psi)}{\partial x}$ and $\frac{\partial(\sigma \ln(1-\theta))}{\partial x}$.

The linear regression analysis of the convergence order at $t = 0.05$ in log-log scale for the test problem with the discontinuous initial condition is shown in Figs. 2.8 and 2.9

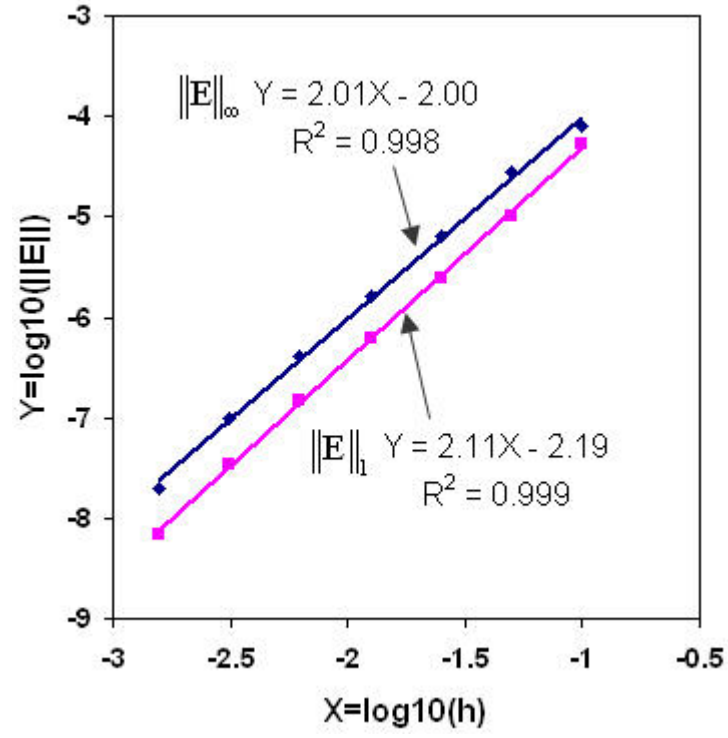


Figure 2.6: Linear regression analysis of the convergence order of θ in log-log scale for the system in (2.50) at $t = 0.05$. The average convergence order for the L_{∞} and L_1 errors are 2.01 and 2.11, respectively.

Table 2.6: Accuracy on θ in the System with Discontinuous Initial Data.

N	L_{∞} error	L_{∞} ratio	L_1 error	L_1 ratio
10	5.28e-4	—	1.57e-4	—
20	2.65e-4	1.99	6.48e-5	2.42
40	1.32e-4	2.01	2.97e-5	2.18
80	6.65e-5	1.98	1.41e-5	2.10
160	3.32e-5	2.01	6.84e-6	2.06
320	1.56e-5	2.13	3.16e-6	2.16
640	5.66e-6	2.75	1.14e-6	2.77

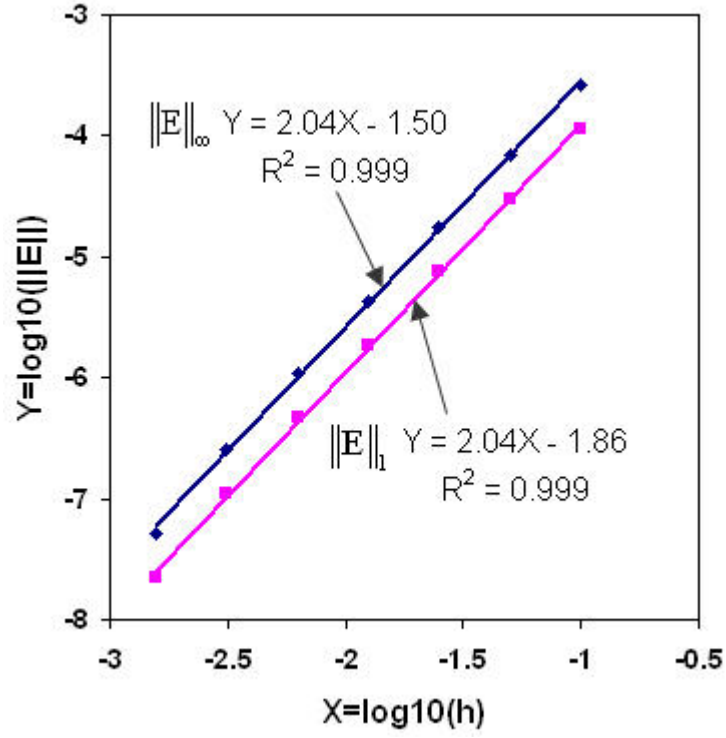


Figure 2.7: Linear regression analysis of the convergence order of v in log-log scale for the system in (2.50) at $t = 0.05$. The average convergence order for the L_{∞} , L_1 and L_2 errors are 2.04 and 2.04, respectively.

Table 2.7: Accuracy on v in the System with Discontinuous Initial Data.

N	L_{∞} error	L_{∞} ratio	L_1 error	L_1 ratio
10	4.22e-3	—	1.26e-3	—
20	2.15e-3	1.96	5.91e-4	2.14
40	1.11e-3	1.94	2.91e-4	2.03
80	5.83e-4	1.91	1.49e-4	1.95
160	3.06e-4	1.91	7.74e-5	1.93
320	1.48e-4	2.07	3.71e-5	2.09
640	5.30e-5	2.79	1.33e-5	2.79

for θ and v , respectively. The average convergence order for the L_∞ and L_1 errors are 1.06 and 1.15 for θ , and 1.02 and 1.06 for v , respectively.

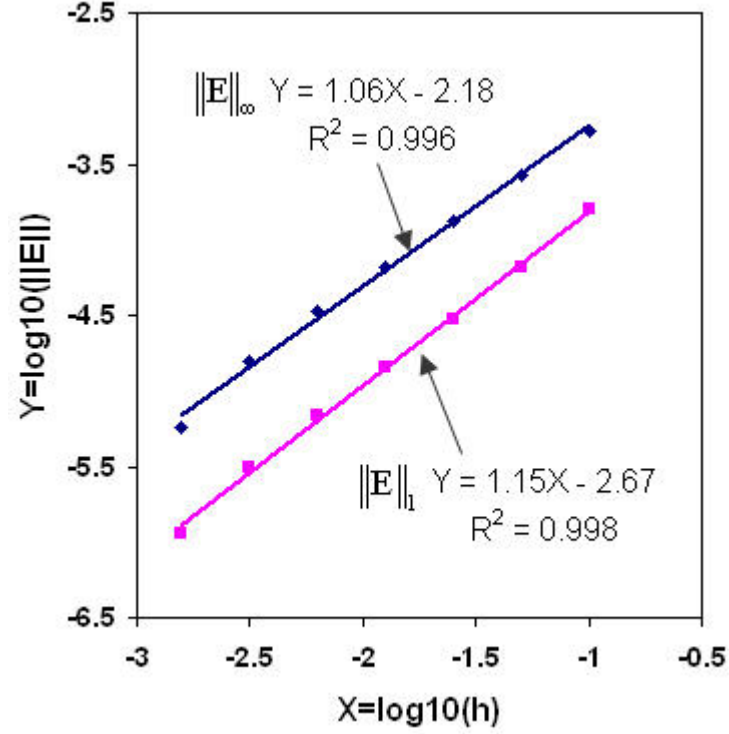


Figure 2.8: Linear regression analysis of the convergence order of θ in log-log scale for the system in (2.50) with the discontinuous IC at $t = 0.05$. The average convergence order for the L_∞ and L_1 errors are 1.06 and 1.15, respectively.

2.3.4 Applying the Immersed Interface Method

We got second order accuracy for the system (2.50) with smooth initial condition due to the smoothness of the solution. However, when the solution is not smooth or discontinuous, conventional finite difference methods give poor approximation on differentiation. We got first order accuracy for the system (2.50) with discontinuous initial condition. Hence we will try the immersed interface method (IIM) in hopes of getting back to second order.

To maintain accuracy for the velocity v near the discontinuities, we apply the IIM [41, 48]. The idea of the IIM mainly counts on Taylor expansion and builds jump conditions into the stencil (irregular grid points) near the interface.

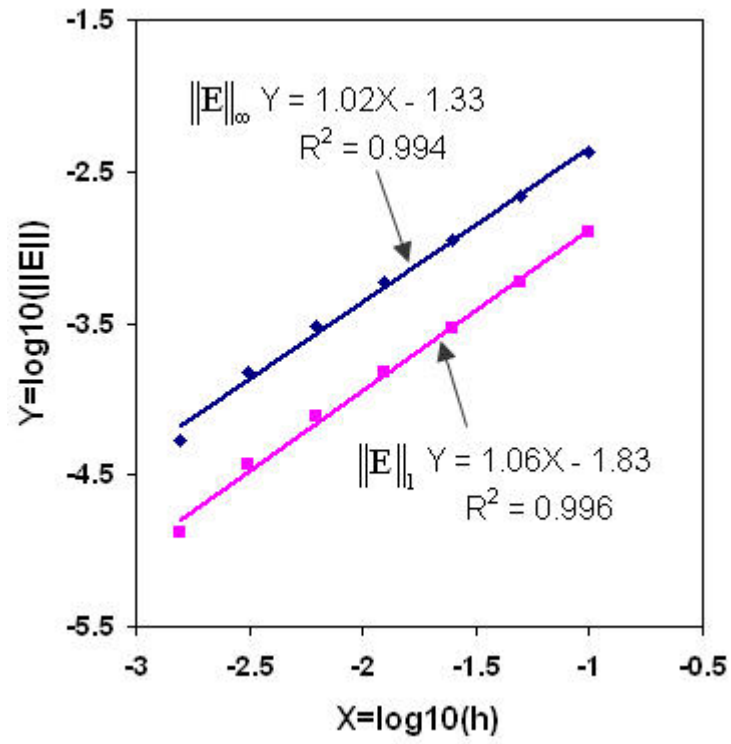


Figure 2.9: Linear regression analysis of the convergence order of v in log-log scale for the system in (2.50) with the discontinuous IC at $t = 0.05$. The average convergence order for the L_{∞} , L_1 and L_2 errors are 1.02 and 1.06, respectively.

The idea is simple. Initially, we know the locations of the discontinuities of θ , so we use the IIM to solve v to second order. Then from the computed velocity, we determine the new locations of the discontinuities and solve θ piece by piece accurately. The process then is repeated. We use Model 2 to illustrate the idea.

Jump conditions across the interface

Standard finite difference methods, for instance, Lax-Wendroff method, Beam-Warming method [13], etc., are typically derived using Taylor expansions and rely on the assumption that the solution to the problem is smooth on the domain of concern. Obviously, these approximations are not valid in the vicinity of an interface where the solution is not smooth. But if we know the location of the interface and the jumps in the solution and its derivatives at the interface, then we can use this information to derive an algorithm to achieve more accurate numerical solution.

The IIM is one of the sharp interface methods which relies on the fact that it is required to specify jump conditions for the solution and its derivatives at the interfaces. Our goal is to achieve second order accuracy. So it is very important to find the jump conditions for the solution and its first and second derivatives.

Jump conditions in tissue interaction problems can take on many forms, and may sometimes involve surface stresses or changes of state at the interface. In this study, we will look at simpler jump conditions. For the problems we are solving we will require equal velocities at the interface, and zero net tangential stress and zero net normal stress across the interface. However, the PDE system does not supply enough information on jump conditions needed for the numerical methods we are using to achieve high order accurate solutions.

Starting from the given piecewise constant initial condition for θ , we know there are jumps in θ , say

$$[\theta]_{\alpha,n} \equiv \lim_{x \rightarrow \alpha^+} \theta(x, t^n) - \lim_{x \rightarrow \alpha^-} \theta(x, t^n) = \theta(\alpha^+, t^n) - \theta(\alpha^-, t^n),$$

where the $+$ or $-$ superscript denotes the right or left side of the interfaces in 1D problems. This expression denotes the jump at location $x = \alpha$ and time step n for θ .

The elliptic equation for velocity in (2.50) shows that

$$v_{xx} = -(\psi\theta)_x - (\sigma \ln(1 - \theta))_x.$$

First considering the case that solutions are continuous, by integrating, we have

$$v_x = -\psi\theta - \sigma \ln(1 - \theta) + c(t), v = -\int_0^x (\psi\theta + \sigma \ln(1 - \theta))dx + c(t)x.$$

Applying the boundary condition, $v(0, t) = v(1, t) = 0$, we have

$$c(t) = \int_0^1 (\psi\theta + \sigma \ln(1 - \theta))dx$$

So we obtain the analytic solution of velocity in term of θ

$$v = -\int_0^x (\psi\theta + \sigma \ln(1 - \theta))dx + x \int_0^1 (\psi\theta + \sigma \ln(1 - \theta))dx$$

When θ is constant on the domain, v is linear in space.

Now consider the case of θ piecewise constant. From the elliptic equation for v , it is obvious that v is piecewise linear on the domain. We assume that the interfaces move with the fluid and there is no fluid slip at the interface, so the velocity is continuous, i.e., $[v] = 0$ across the interfaces. So the analytic solution to v corresponding to piecewise constant θ can be written as

$$v(x, t) = \begin{cases} a_1(t)x + a_0(t), & \text{if } 0 \leq x \leq \alpha_1, \\ b_1(t)x + b_0(t), & \text{if } \alpha_1 \leq x \leq \alpha_2, \\ c_1(t)x + c_0(t), & \text{if } \alpha_2 \leq x \leq 1. \end{cases}$$

By the boundary condition for v , we get $a_0(t) = 0$ and $c_0(t) = -c_1(t)$. Applying the continuity of v , we have $b_0(t) = a_1(t) - b_1(t)$. Then

$$(b_1(t) - a_1(t))\alpha_1 + (c_1(t) - b_1(t))\alpha_2 = c_1(t) \quad (2.51)$$

in which we have three unknowns and one equation. We need another two equations to find the analytic solution to the elliptic equation.

The elliptic equation for v tells us

$$\begin{aligned} [v_{xx}] &= -[(\psi\theta)_x] - [(\sigma \ln(1 - \theta))_x], \\ [v_x] &= -[\psi\theta] - [\sigma \ln(1 - \theta)], \\ [v] &= 0. \end{aligned}$$

The jumps in first derivative of v across the interface give us two equations

$$\begin{aligned} b_1(t) - a_1(t) &= -[\psi\theta]_{\alpha_1, t} - [\sigma \ln(1 - \theta)]_{\alpha_1, t}, \\ c_1(t) - b_1(t) &= -[\psi\theta]_{\alpha_2, t} - [\sigma \ln(1 - \theta)]_{\alpha_2, t}. \end{aligned}$$

Along with (2.51), we obtain

$$\begin{aligned} a_1(t) &= [\Lambda]_1(\alpha_1 - 1) + [\Lambda]_2(\alpha_2 - 1), \\ b_1(t) &= [\Lambda]_1\alpha_1 + [\Lambda]_2(\alpha_2 - 1), \\ c_1(t) &= [\Lambda]_1\alpha_1 + [\Lambda]_2\alpha_2, \end{aligned}$$

where $[\Lambda]_\ell$ denote $-[\psi\theta]_{\alpha_\ell, t} - [\sigma \ln(1 - \theta)]_{\alpha_\ell, t}$ at the interface ℓ , and $\ell = 1, 2$ in our problem. Hence the analytic solution to v in term of θ is

$$v(x, t) = \begin{cases} ([\Lambda]_1(\alpha_1 - 1) + [\Lambda]_2(\alpha_2 - 1))x, & \text{if } 0 \leq x \leq \alpha_1, \\ ([\Lambda]_1\alpha_1 + [\Lambda]_2(\alpha_2 - 1))x - [\Lambda]_1\alpha_1, & \text{if } \alpha_1 \leq x \leq \alpha_2, \\ ([\Lambda]_1\alpha_1 + [\Lambda]_2\alpha_2)(x - 1), & \text{if } \alpha_2 \leq x \leq 1. \end{cases} \quad (2.52)$$

Based on our analysis, we can conclude that the jumps for v are

$$\begin{aligned} [v] &= 0, & \text{continuity of } v, \\ [v_{xx}] &= 0, & \text{no jump in stress,} \\ [v_x] &= -[\psi\theta] - [\sigma \ln(1 - \theta)]. \end{aligned} \quad (2.53)$$

So the interface relations are

$$\begin{aligned} v_{xx}(\alpha^+, t) &= v_{xx}(\alpha^-, t), \\ v_x(\alpha^+, t) &= v_x(\alpha^-, t) - [\psi\theta]_{\alpha, t} - [\sigma \ln(1 - \theta)]_{\alpha, t}, \\ v(\alpha^+, t) &= v(\alpha^-, t). \end{aligned} \quad (2.54)$$

Correction terms for irregular grids

Suppose θ_i^n , denoting θ at the grid point i and time t^n , are known for all i in the domain. And suppose the interfaces are at $x = \alpha_\ell$, $\ell = 1, 2$, which lie in the intervals between x_{j_ℓ} and $x_{j_\ell+1}$ (see Fig. 2.10).

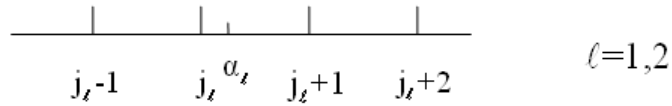


Figure 2.10: 1-D interface.

We apply the central finite difference method (CFDM) for the elliptic equation for v in (2.50) at regular grid points, i.e., those for which a interface does not lie in the interval

of the 3-point stencil,

$$\frac{V_{i-1}^n - 2V_i^n + V_{i+1}^n}{\Delta x^2} = -\frac{(\psi\theta)_{i+1}^n - (\psi\theta)_{i-1}^n}{2\Delta x} - \frac{(\sigma \ln(1-\theta))_{i+1}^n - (\sigma \ln(1-\theta))_{i-1}^n}{2\Delta x}.$$

The local truncation error at regular grid points is $O(\Delta x^2)$

The irregular grid points for the CFDM are j_ℓ and $j_\ell + 1$ near the interface ℓ . For simplicity, we drop the subscript ℓ in the rest of this chapter. The CFDM is not valid at those irregular points. We need to find the correction terms to make the method valid. Since the parameters in our test problem in (2.50) are constant, the coefficients of the finite difference equation are the same as those for the regular grid points except for the correction term,

$$\frac{V_{J-1}^n - 2V_J^n + V_{J+1}^n}{\Delta x^2} = -\frac{(\psi\theta)_{J+1}^n - (\psi\theta)_{J-1}^n}{2\Delta x} - \frac{(\sigma \ln(1-\theta))_{J+1}^n - (\sigma \ln(1-\theta))_{J-1}^n}{2\Delta x} + C_J^n, \quad (2.55)$$

where $J = j, j + 1$.

The method to find the correction terms is based on the Taylor series expansion at the interfaces. Consider the irregular point at $J = j$ first. The local truncation error is

$$\begin{aligned} T_J = & \frac{v(x_{J-1}, t^n) - 2v(x_J, t^n) + v(x_{J+1}, t^n)}{\Delta x^2} + \frac{(\psi\theta)(x_{J+1}, t^n) - (\psi\theta)(x_{J-1}, t^n)}{2\Delta x} \\ & + \frac{(\sigma \ln(1-\theta))(x_{J+1}, t^n) + (\sigma \ln(1-\theta))(x_{J-1}, t^n)}{2\Delta x} - C_J^n. \end{aligned} \quad (2.56)$$

We want to make the magnitude of the truncation error as small as possible by choosing the correction term. In order to obtain second order global accuracy, we need to ensure an $O(\Delta x)$ local truncation error at those irregular points [38]. In other words, we need to expand out v through $O(\Delta x^3)$ and expand out $\psi\theta$ and $\sigma \ln(1-\theta)$ through $O(\Delta x^2)$. We expand $v(x_{J-1}, t^n)$, $v(x_J, t^n)$, $v(x_{J+1}, t^n)$, $(\psi\theta)(x_{J+1}, t^n)$, $(\psi\theta)(x_{J-1}, t^n)$, $(\sigma \ln(1-\theta))(x_{J+1}, t^n)$ and $(\sigma \ln(1-\theta))(x_{J-1}, t^n)$ in Taylor series about the interface α relating to J . Dropping the t^n for simplicity due to no time evolution in the equation for v , and using the notation

$$v^+ = \lim_{x \rightarrow \alpha^+} v(x), \quad v^- = \lim_{x \rightarrow \alpha^-} v(x),$$

we obtain

$$\begin{aligned}
v(x_{J-1}) &= v^- + v_x^-(x_{J-1} - \alpha) + \frac{1}{2}v_{xx}^-(x_{J-1} - \alpha)^2 + O(\Delta x^3), \\
v(x_J) &= v^- + v_x^-(x_J - \alpha) + \frac{1}{2}v_{xx}^-(x_J - \alpha)^2 + O(\Delta x^3), \\
v(x_{J+1}) &= v^+ + v_x^+(x_{J+1} - \alpha) + \frac{1}{2}v_{xx}^+(x_{J+1} - \alpha)^2 + O(\Delta x^3), \\
(\psi\theta)(x_{J-1}) &= (\psi\theta)^- + (\psi\theta)_x^-(x_{J-1} - \alpha) + O(\Delta x^2), \\
(\psi\theta)(x_{J+1}) &= (\psi\theta)^+ + (\psi\theta)_x^+(x_{J+1} - \alpha) + O(\Delta x^2), \\
(\sigma \ln(1 - \theta))(x_{J-1}) &= (\sigma \ln(1 - \theta))^- + (\sigma \ln(1 - \theta))_x^-(x_{J-1} - \alpha) + O(\Delta x^2), \\
(\sigma \ln(1 - \theta))(x_{J+1}) &= (\sigma \ln(1 - \theta))^+ + (\sigma \ln(1 - \theta))_x^+(x_{J+1} - \alpha) + O(\Delta x^2).
\end{aligned}$$

Plugging the above expansions into the local truncation error (2.56) and using the interface relations for v (2.54), along with the following interface relations for $\psi\theta$ and $\sigma \ln(1 - \theta)$:

$$\begin{aligned}
\psi\theta_x(\alpha^+, t) &= \psi\theta_x(\alpha^-, t), \\
\psi\theta(\alpha^+, t) &= \psi\theta(\alpha^-, t) + [\psi\theta]_{\alpha, t}, \\
\sigma \ln(1 - \theta)_x(\alpha^+, t) &= \sigma \ln(1 - \theta)_x(\alpha^-, t), \\
\sigma \ln(1 - \theta)(\alpha^+, t) &= \sigma \ln(1 - \theta)(\alpha^-, t) + [\sigma \ln(1 - \theta)]_{\alpha, t},
\end{aligned} \tag{2.57}$$

we can eliminate the quantities from one side in terms of those from the other side to get the following expression for the local truncation error at grid point x_J

$$\begin{aligned}
T_J &= \frac{v_{xx}^-((x_{J+1} - \alpha)^2 - 2(x_J - \alpha)^2 + (x_{J-1} - \alpha)^2) + O(\Delta x^3)}{\Delta x^2} \\
&\quad + \frac{2\Delta x((\psi\theta)_x^- + (\sigma \ln(1 - \theta))_x^-) + O(\Delta x^2)}{2\Delta x} + \frac{[\Lambda]_\alpha(x_{J+1} - \alpha)}{(\Delta x)^2} - \frac{[\Lambda]_\alpha}{2\Delta x} - C_J \tag{2.58} \\
&= v_{xx}^- + (\psi\theta)_x^- + (\sigma \ln(1 - \theta))_x^- + [\Lambda]_\alpha\left(\frac{x_{J+1} - \alpha}{\Delta x^2} - \frac{1}{2\Delta x}\right) - C_J + O(\Delta x)
\end{aligned}$$

where $[\Lambda]_\alpha = -[\psi\theta]_\alpha - [\sigma \ln(1 - \theta)]_\alpha$. From the elliptic equation itself, we know that $v_{xx}^- + (\psi\theta)_x^- + (\sigma \ln(1 - \theta))_x^- = 0$. Setting

$$C_J = [\Lambda]_\alpha\left(\frac{x_{J+1} - \alpha}{\Delta x^2} - \frac{1}{2\Delta x}\right) = \frac{1}{2}[\Lambda]_\alpha(\delta_{\Delta x}(x_J - \alpha) - \delta_{\Delta x}(x_{J+1} - \alpha))$$

where

$$\delta_{\Delta x}(x) = \begin{cases} \frac{\Delta x - |x|}{\Delta x^2}, & \text{if } |x| < h, \\ 0, & \text{otherwise,} \end{cases} \tag{2.59}$$

we get $O(\Delta x)$ for the local truncation error at the irregular grid point x_J .

Likewise, we can find the correction term for the irregular grid point x_{J+1} by analyzing the local truncation error at that point:

$$C_{J+1} = \frac{1}{2}[\Lambda]_\alpha(\delta_{\Delta x}(x_{J+1} - \alpha) - \delta_{\Delta x}(x_J - \alpha))$$

Note that all the analysis above for the jump conditions across the interfaces and the correction terms for the irregular grid points are valid for the cases with parameters of Model 2 listed in Table 2.2. For the cases with parameters of other models listed in Table 2.2, it is nearly impossible to derive sufficient jump conditions from the given PDE system for the numerical method. This implies that we can not apply the IIM to those case unless all the necessary jump conditions are given.

The IIM gives more accurate results for both θ and v and eliminates non-physical oscillations (the Gibbs phenomenon). But in order to use this method, we need a prior knowledge of the shock locations.

2.4 Numerical Results

We take $L = 1$ for simplicity in this section. We have done a number of numerical experiments for different parameters and initial conditions. All the cases are listed in Table 2.8. We use continuous initial condition in Cases (a-d) to understand the pattern formation. In the other cases, we use discontinuous initial condition. We want to know the long time behavior of the solution and whether θ will grow and then stabilize. Our results indicate that θ in Model 2 grows faster compared with other models. In all cases presented here, we use dotted lines to represent the initial data, dash-dotted lines to represent the solution at the final time (often $T = 5$), and solid lines to represent the intermediate solution between initial and final time. The initial θ is piecewise constant with the discontinuities at $x_1 = 0.35$, $x_2 = 0.65$ with $\theta_l = 0$ or 0.1 and $\theta_u = 0.5$.

2.4.1 Cases with parameters of Model 1

First we consider the continuous initial condition described in (2.5) to find out the pattern of tissue deformation.

Using the parameters of Model 1 in Table 2.2 for M , ψ and σ , we apply a central finite difference scheme to set up the system of difference equations for v at each time step.

Table 2.8: List of Numerical Experiments.

case	M	φ	ψ	σ	N	T	IC	θ_l	θ_u	alg.	fig. ref.
(a-d)	$\frac{\theta}{0.5+\theta}$	1.0	θ	$1 - \theta$	160	5	(2.5)	-	-	1	Fig. 2.11
(1)	0.5	0	1.8	1.0	640	5	(2.6)	0	0.5	1	Fig. 2.13
(2)	0.5	0	1.8	1.0	640	5	(2.6)	0	0.5	3	Fig. 2.13
(3)	0.5	0	1.8	1.0	640	15	(2.6)	0	0.5	1	Fig. 2.14
(4)	0.5	0	1.8	1.0	80	15	(2.6)	0	0.5	1	Fig. 2.14
(5)	0.5	0	2.5	1.0	640	5	(2.6)	0	0.5	1	Fig. 2.15
(6)	0.5	0	1.0	1.0	80	20	(2.6)	0	0.5	1	Fig. 2.15
(7)	0.5	0	1.8	1.0	640	5	(2.6)	0	0.5	2	Fig. 2.16
(8)	0.5	0	1.8	1.0	320	15	(2.6)	0	0.5	2	Fig. 2.16
(9)	0.5	0	2.5	1.0	640	5	(2.6)	0.1	0.5	2	Fig. 2.17
(10)	0.5	0	1.0	1.0	80	20	(2.6)	0.1	0.5	2	Fig. 2.17
(11)	0.5	0	1.8	1.0	640	5	(2.6)	0.1	0.5	3	Fig. 2.18
(12)	$0.5e^\theta$	e^θ	1.8	1.0	640	5	(2.6)	0	0.5	3	Fig. 2.19
(13)	$0.5e^\theta$	e^θ	1.8	1.0	640	5	(2.6)	0	0.5	1	Fig. 2.19
(14)	$0.5e^\theta$	e^θ	1.8	1.0	640	15	(2.6)	0	0.5	1	Fig. 2.19
(15)	$0.5e^\theta$	e^θ	$1.8e^{-\theta}$	$e^{-\theta}$	640	5	(2.6)	0	0.5	3	Fig. 2.20
(16)	$0.5e^\theta$	e^θ	$1.8e^{-\theta}$	$e^{-\theta}$	640	5	(2.6)	0	0.5	1	Fig. 2.20
(17)	$0.5e^\theta$	e^θ	$1.8e^{-\theta}$	$e^{-\theta}$	640	15	(2.6)	0	0.5	1	Fig. 2.20
(18)	$0.5e^\theta$	e^θ	$1.8e^{-\theta}$	$e^{-\theta}$	640	5	(2.6)	0	0.5	2	Fig. 2.21
(19)	$0.5e^\theta$	e^θ	$1.8e^{-\theta}$	$e^{-\theta}$	320	15	(2.6)	0	0.5	2	Fig. 2.21

Algorithm 1: upwind + CFDM;

Algorithm 2: WENO + CFDM;

Algorithm 3: upwind + IIM;

Fig. 2.11 shows the numerical solutions for θ , v and p by the upwind scheme along with central FDM for $t \leq 5$. Four cases with different initial spatial frequencies, $k=2, 6, 8$ and 18 respectively, are used in the computations.

It is observed that it takes longer for the case with lower frequency to blow up. The value of θ will be greater than 1 when $t \geq 4.3$ for $k = 6$ and $t \geq 3.0$ for $k = 8$. $k = 18$ is very close to the possible maximum k for $\theta_0 = 0.5$ (see Fig. 2.4). Hence after several time steps, $t \geq 0.1$, the solution blows up very quickly, i.e., $\theta > 1$.

2.4.2 Cases with parameters of Model 2 and $\theta_l = 0$

In this subsection, we consider the cases with the piecewise constant initial condition (2.6). Using the parameters of Model 2 in Table 2.2 for $M = 0.5$, $\psi = 1.8$ and $\sigma = 1$

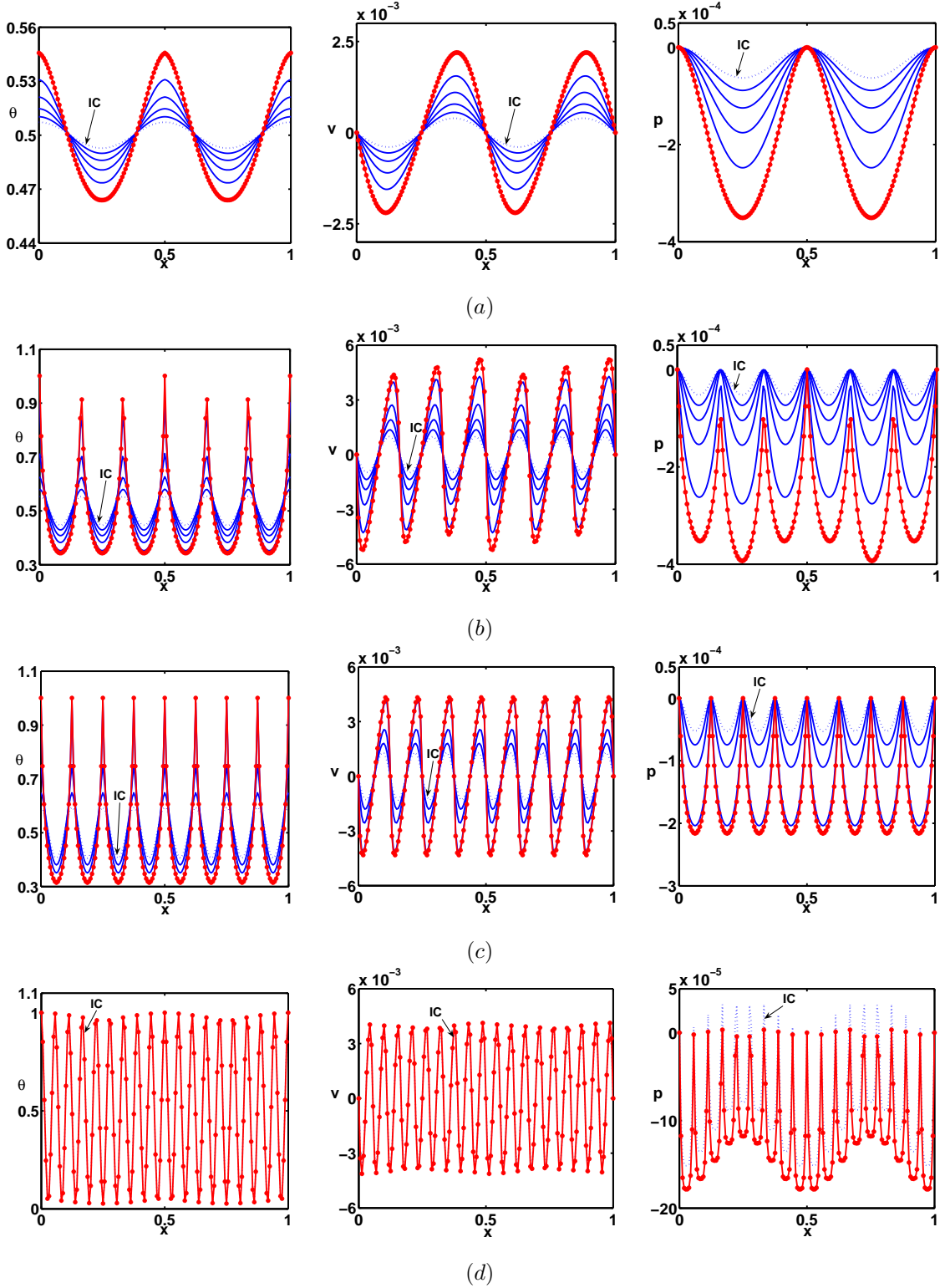


Figure 2.11: Computed solutions of (2.1), (2.2) and (2.3) along with IC in (2.5) and parameters in (2.25) by the upwind scheme along with CFDM. (a) – the solutions of θ (left), velocity (middle) and pressure (right) for the case with $\varepsilon = 0.001$, $t \leq 5.0$, and $k = 2$; (b), (c) and (d) – similar to (a), but with $k=6, 8$ and 18 , respectively. In all plots, the dotted lines represent the initial data, the solid lines represent the intermediate solutions between initial and final time, and the * lines in (a) represent the solutions at $t = 5.0$, those in (b), (c) and (d) are the solutions when $\theta \geq 1$. In (d), solutions blow up at $t = 0.1$. $N = 160$.

and choosing $x_1 = 0.35$, $x_2 = 0.65$, $\theta_l = 0$, initial $\theta_u = 0.5$, we employ several numerical algorithms to compute the system. For the given parameters, we can use (2.22) to determine the value of θ_u which makes the system reach steady state. Fig. 2.12 shows the curve for (2.22) as a function of θ_u .

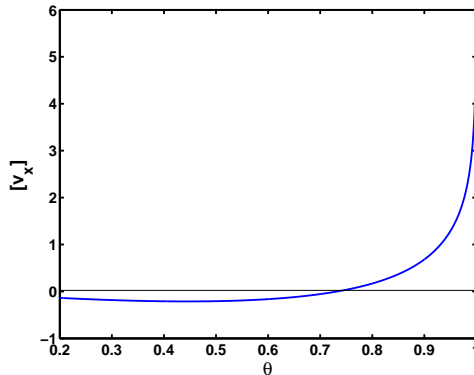


Figure 2.12: $[v_x]$ for the case with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$

Using the command *fzero* in MATLAB, one can easily find the value of θ_u that makes $[v_x] = 0$, which is $\theta_u = 0.73$ for our case.

Computations are completed for $T = 5$. Fig. 2.13 shows the solutions for θ and v to the system with the parameters indicated above via the upwind scheme along with CFDM (case 1) and the upwind scheme along with IIM (case 2).

The maximum difference of θ between two time steps near the last time step is about $3.6e^{-5}$ for the upwind scheme along with IIM, *i.e.*, the solution converges to the steady state. The maximum difference of θ between two time steps near last time step for the upwind scheme along the CFDM is about $5e^{-5}$. Oscillations at the discontinuities at later time were observed in θ (2.14). This could be induced by the initial data and the numerical method although the upwind scheme does not produce oscillations near the interface for the advection equations. As discussed in the linear stability analysis section, θ at two side regions, θ_l , always stays in the region where the system is unstable, but the θ in the middle region, θ_u , stays in the region where the system is stable to approach to the steady state for the parameters and initial data we used. The upwind scheme is known to be first order accurate. The numerical errors accumulate with time. The discontinuities in the numerical solution will be smoothed out at later times by this scheme. However, when

the upwind scheme and CFDM are applied to our problem, oscillations are observed. Our study shows that oscillation spreads out widely in the domain if coarser grids are used (see case 4 with coarser grids, $N = 80$, in Fig. 2.14). When the set of parameters chosen stays in the region that under the horizontal line or above the dashed curve in Fig. 2.2, there are no oscillations observed for longer time computations, i.e., $t \gg 20$.

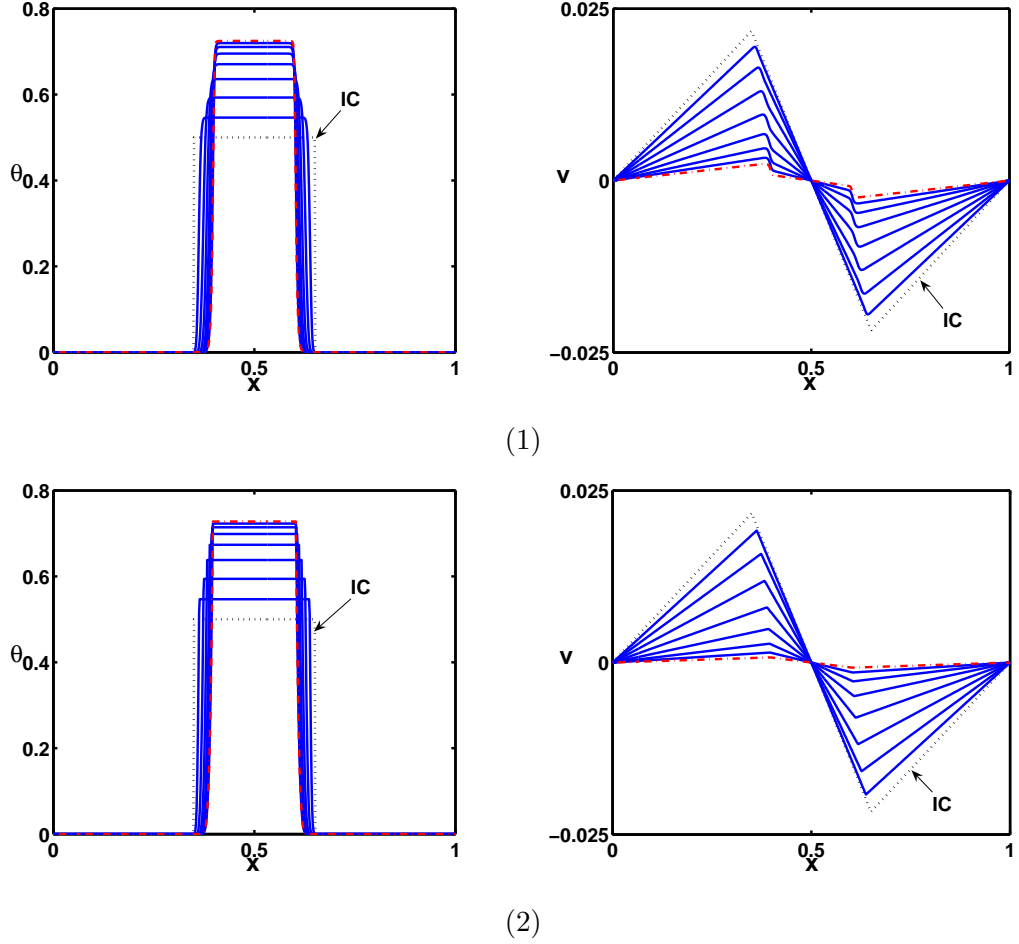
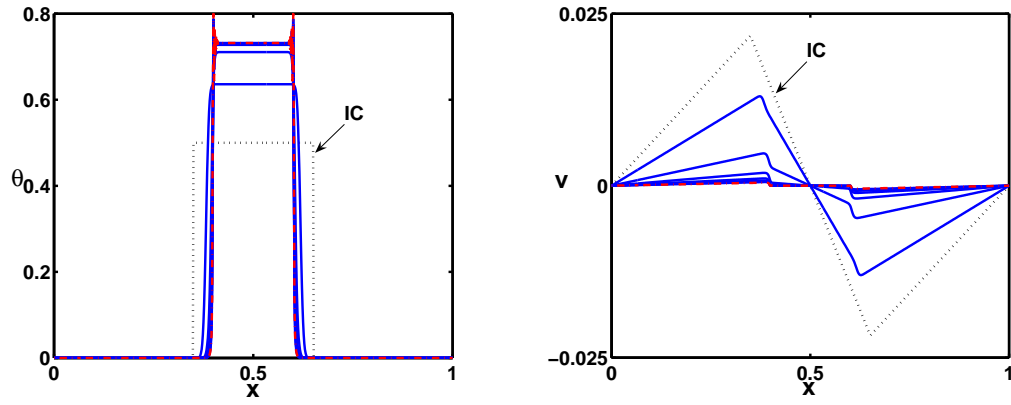
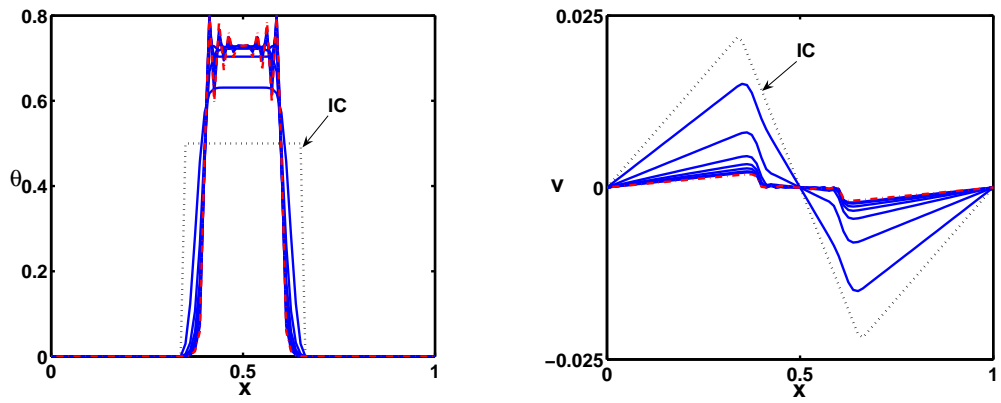


Figure 2.13: Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$ for $T = 5$. (1) – the solutions of θ (left) and velocity (right) of Case 1 solved via the upwind scheme along with CFDM; (2) – the solutions of θ (left) and velocity (right) of Case 2 solved via the upwind scheme along with IIM. Fine meshes, $N = 640$.

To test our conclusion about the system's stable region, two other computations for the system with constant parameters are completed. Same parameters were chosen except that $\psi = 2.5$ in case 5, in which the system is totally unstable, and $\psi = 1.0$ in case 6,



(3)



(4)

Figure 2.14: Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$ for $T = 15$ by the upwind scheme along with CFDM. (3) – the solutions of θ and velocity of Case 3 with fine meshes, $N = 640$; (4) – the solutions of θ and velocity of Case 4 with coarse meshes, $N = 80$.

in which the system is stable. Fig. 2.15 shows the computed solutions for both cases. Note that the dashed lines in Fig. 2.15 (5) indicate the solution while passing the neutral stable (the dashed line in Fig. 2.2) at $t = 0.9$ to approach to the equilibrium. Oscillation was observed in the case 5 at later time. This is because at very beginning with given initial data and chosen parameter, the system is in the unstable region (see Fig. 2.2), hence θ_u grows with time. However, when $\theta_u > 0.6$, the system transfers from the unstable region into the oscillation region and approaches to the equilibrium. We find that the oscillation occurs when the system is reaching equilibrium. For case 6, one can observe that θ decreases with time. The system will always stay in the stable region. No oscillation will occur for a long time run. These numerical experiments support our conclusions in linear stability analysis.

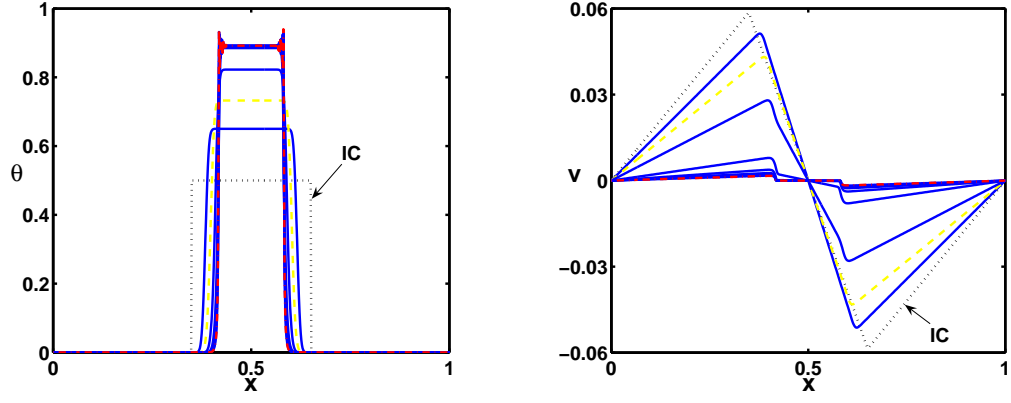
Fig. 2.16 shows the computed solutions solved by WENO-Roe scheme along with optimal third order TVD Runge-Kutta method for the hyperbolic equation and CFDM for the elliptic equation with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$. In Case 7, computation is completed with $T = 5$ and $N = 640$. Oscillation is observed for a longer time computation with this algorithm. See Case 8 in Fig. 2.16, in which computation is completed with $T = 15$ and $N = 320$.

2.4.3 Cases with parameters of Model 2 and $\theta_l = 0.1$

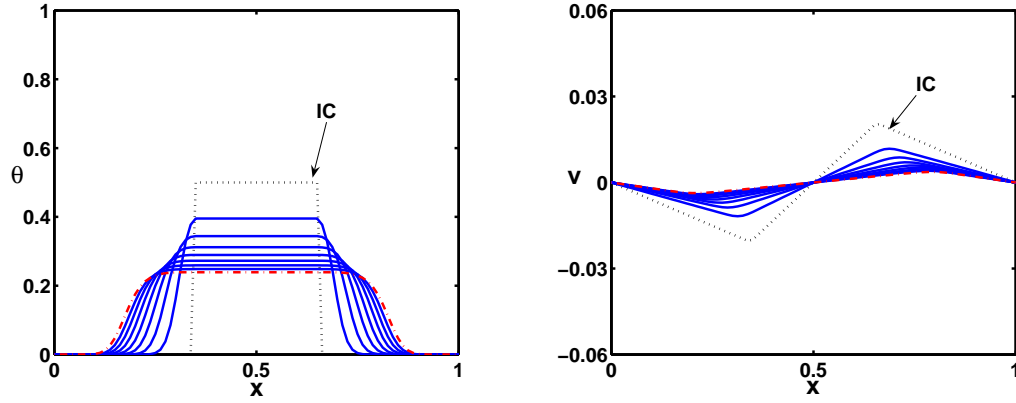
In this subsection, we consider the cases with all the same conditions as those in the last subsection except $\theta_l = 0.1$. We employ two numerical algorithms to compute the system.

In Fig. 2.17 and Fig. 2.18, we show the computed results using the two different numerical methods with different parameters. By choosing the parameters according to our stability analysis, we obtained the desired results with θ in the middle part growing (the first plot in Fig. 2.17 and Fig. 2.18) or decaying (the lower bottom plot in Fig. 2.17). The solution will approach the steady states after some time.

From the computed results, we can conclude that the two methods, (1) the WENO and central finite difference scheme and (2) the IIM approach, give qualitatively the same results in the solution except at the discontinuities. The IIM approach eliminates the non-physical oscillations by enforcing the jump conditions in the finite difference scheme.



(5)



(6)

Figure 2.15: Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\sigma = 1$ by the upwind scheme along with CFDM. (5) – the solutions of θ (left) and velocity (right) of Case 5 with $\psi = 2.5$ and fine meshes, $N = 640$, $T = 5$; (6) – the solutions of θ (left) and velocity (right) of Case 6 with $\psi = 1$ and coarse meshes, $N = 80$, $T = 20$. In plots of Case 5, the dashed lines indicate the computed solutions while passing the neutral stable state at $t = 0.9$ (referring the dashed line in Fig. 2.2).

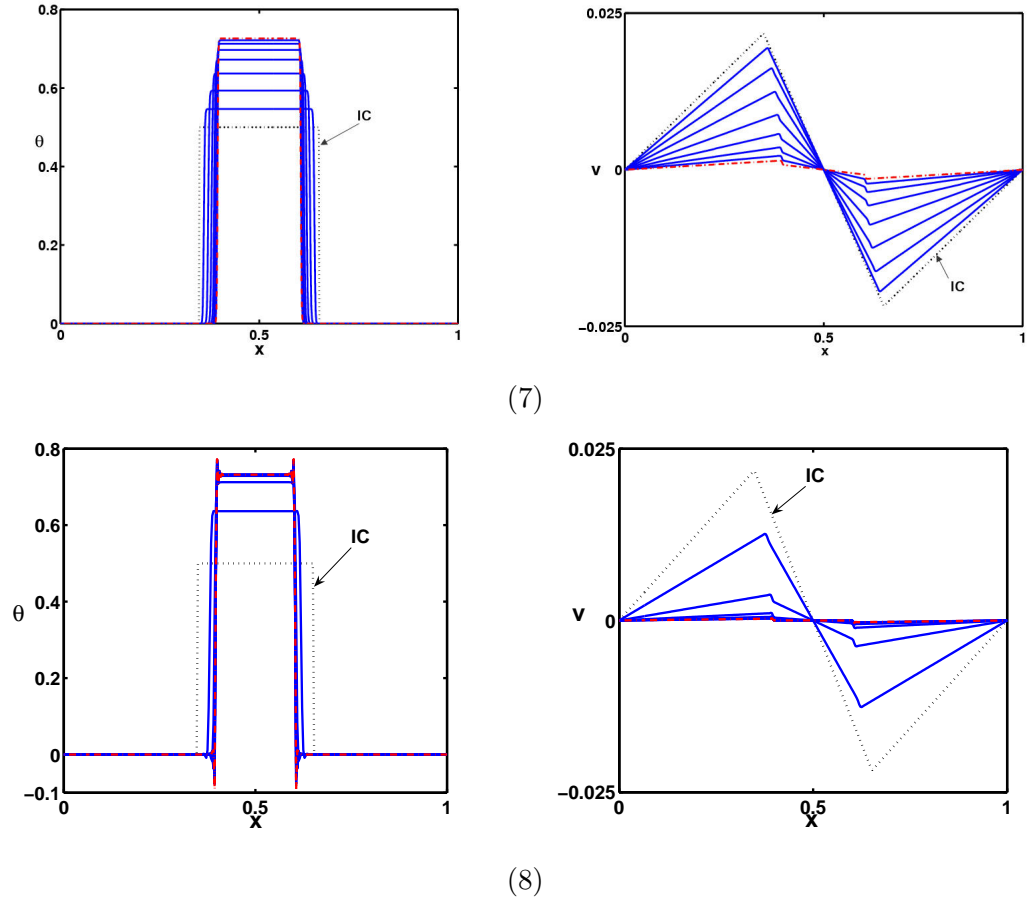
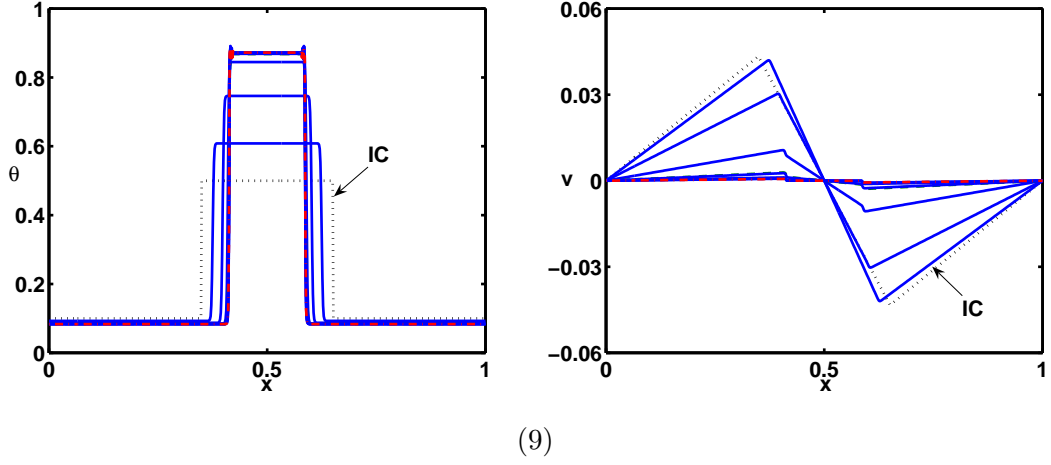
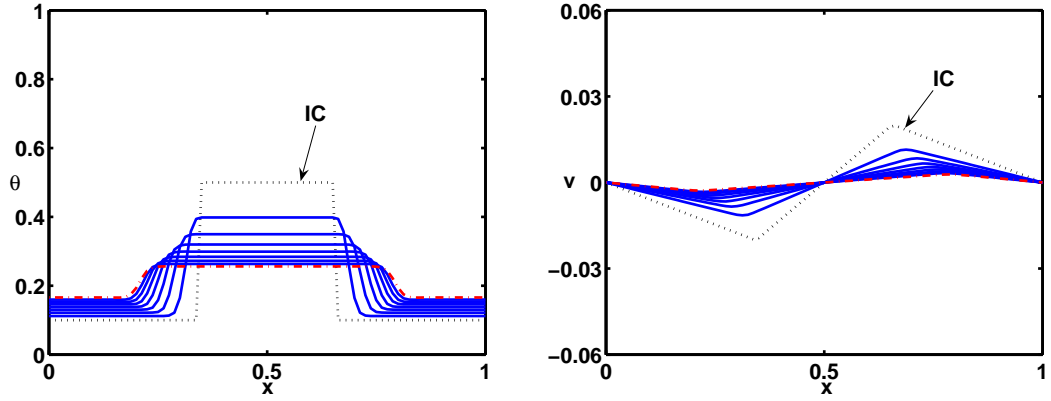


Figure 2.16: Computed solutions of (2.31), (2.32) along with IC in (2.6) and $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$ by the WENO-Roe scheme along with TVD Runge-Kutta method and CFDM. (7) – the solutions of θ (left) and velocity (right) of Case 7 with $N = 640$, $T = 5$; (8) – the solutions of θ (left) and velocity (right) of Case 8 with $N = 320$, $T = 15$.



(9)



(10)

Figure 2.17: Computed solutions using the WENO scheme and central finite difference scheme with different parameters. The top two plots are computed with $\psi = 2.5$, $\sigma = 1$, and $N = 640$. The solution θ grows in the middle. Some oscillations developed near the discontinuities. The bottom plots are computed with $\psi = 1$, $\sigma = 1$, and $N = 80$. The solution θ decays and no oscillations occurred so we take a coarse grid. The final time is $T = 5$ for the top plots while it is $T = 20$ for the bottom ones since it takes a longer time to reach the steady state solution.

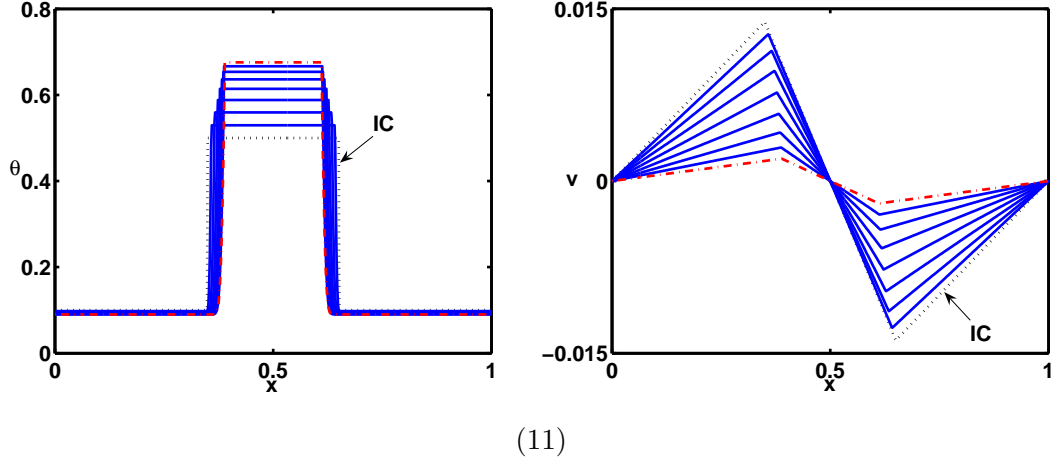
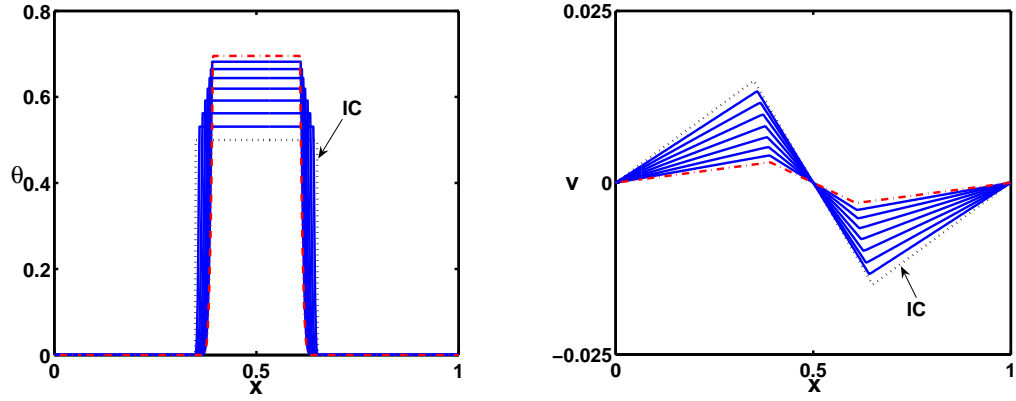


Figure 2.18: Computed solution using the IIM scheme. The results are computed with $\psi = 1.8$, $\sigma = 1$, and $N = 640$. The final time is $T = 5$. There are no oscillations in the computed solution.

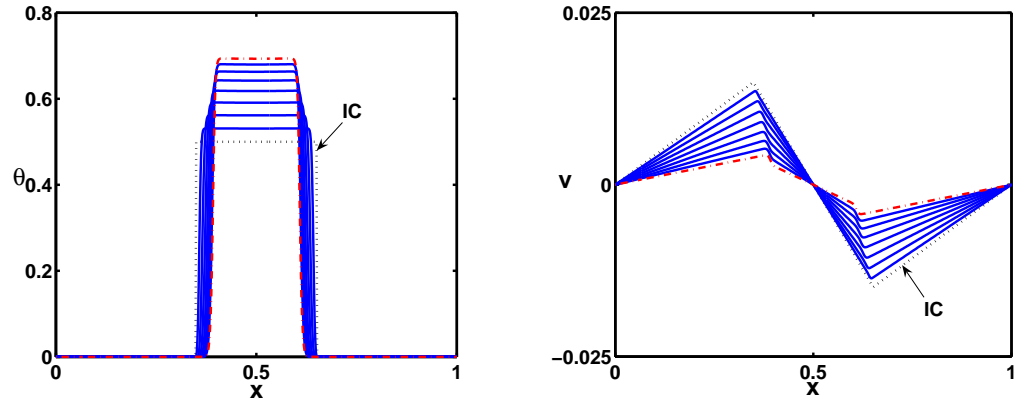
2.4.4 Cases with parameters of Model 3

Now we consider the system, (2.31) - (2.32), with the parameters of Model 3 in Table 2.2. Again we choose $x_1 = 0.35$, $x_2 = 0.65$, $\theta_l = 0$, initial $\theta_u = 0.5$ and use the same approaches as used for Section 2.4.2 to solve the system for comparison. Note that the assumptions about M and φ imply that those parameters are discontinuous across the interface and implicitly depend on time since they are functions of θ . For this model, it is difficult to find a set of parameters that make system at the steady state. Computed solutions to the system solved by the upwind scheme along with IIM (Case 12) and upwind scheme along with CFDM (Case 13 with $T = 5$ and Case 14 with $T = 15$) are shown in Fig. 2.19.

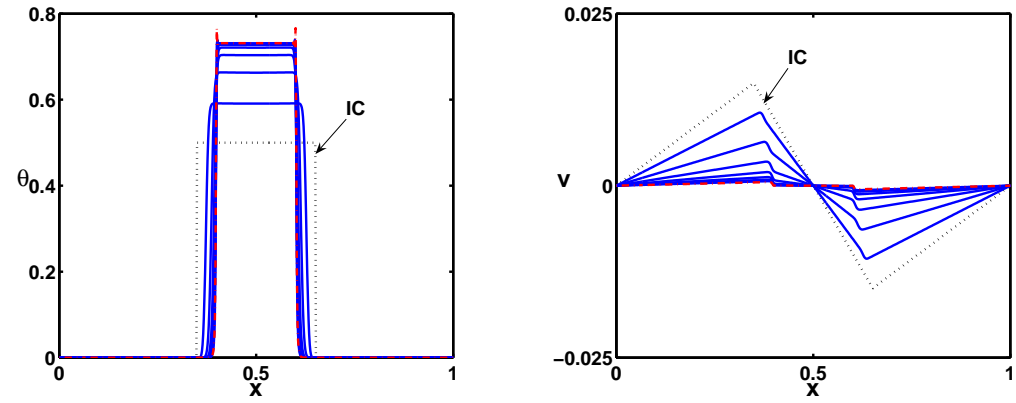
The maximum difference of θ between two time steps near final time step is less than $1.6e^{-4}$ for the upwind scheme along with IIM, i.e., the solution is also approaching the steady state. The maximum difference of θ between two time steps is less than $1e^{-4}$ for the upwind scheme along with CFDM. Compared with the solutions to the system with constant parameters (see Section 2.4.2), θ , the computed solution to the system with Model 3 in Table 2.2, grows more slowly. Case 14 in Fig. 2.19 shows the solutions by the upwind scheme along with CFDM for a long time run ($T=15$). Oscillation is observed again at a later time. However, the amplitude of oscillation becomes smaller in this case.



(12)



(13)



(14)

Figure 2.19: Computed solutions of (2.31), (2.32) along with IC in (2.6) and parameters of Model 3 in Table 2.2. (12) – the solutions of θ (left) and velocity (right) of Case 12 with $T = 5$ and $N = 640$, solved by the upwind scheme along with IIM; (13) – the solutions of θ (left) and velocity (right) of Case 13 with $T = 5$ and $N = 640$, solved by the upwind scheme along with CFDM; (14) – the solutions of θ (left) and velocity (right) of Case 14 with $T = 15$, solved by the upwind scheme along with CFDM.

2.4.5 Cases with parameters of Model 4

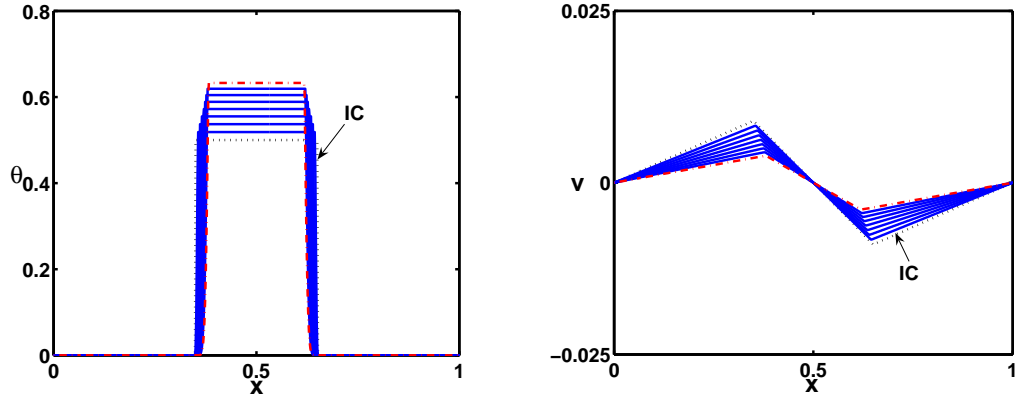
In this example, we consider the system, (2.31) - (2.32), with the parameters of Model 4 in Table 2.2. We assume that the contractility and swelling coefficients, ψ and σ , exponentially decrease with θ . The solutions solved by the upwind scheme along with IIM and upwind scheme along with CFDM are shown in Fig. 2.20.

It is quite obvious that θ in the middle part of the domain is approaching the steady state with the slowest speed comparing with other choices of parameters. In case 15, θ_u is far away from the equilibrium at $T = 5$. The maximum difference of θ between two time steps is about $2.1e^{-4}$ for the upwind scheme along with IIM. Case 16 shows the computed solution to the system by the upwind scheme along with CFDM for $T = 5$. No oscillations are observed by $T = 5$. In this case, θ_u is also far away from the equilibrium at $T = 5$. The maximum difference of θ between two time steps is about $1.1e^{-4}$ by the upwind scheme along with CFDM. Case 17 shows the solution by the upwind scheme along with CFDM for $T = 15$. The oscillation has not appeared by the final computation time ($T = 15$ in this case) because the θ are still away from the equilibrium. But unfortunately, oscillations will occur for the long-time evolution [85] as long as we are using CFDM to solve the elliptic equation for v . The numerical error from v will contaminate the solution of the hyperbolic equation for θ .

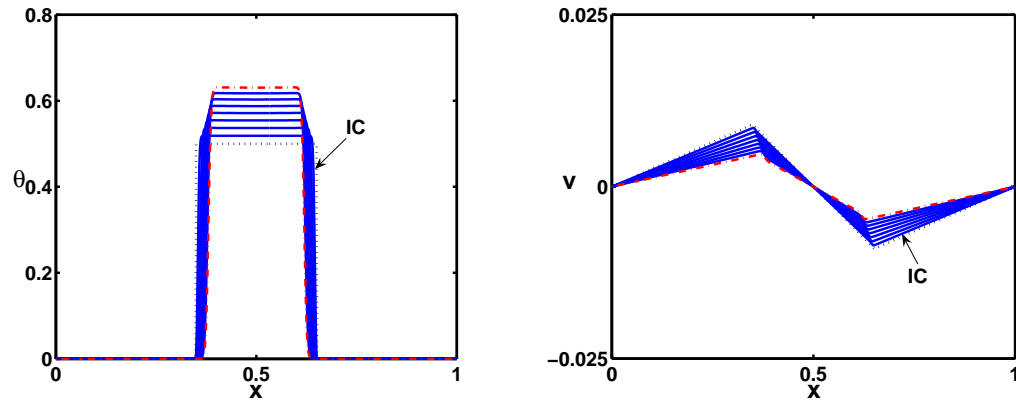
Fig. 2.21 shows the computed solutions solved by WENO-Roe scheme along with TVD Runge-Kutta method and CFDM for the system with parameter of Model 4 in Table 2.2.

2.5 Conclusions

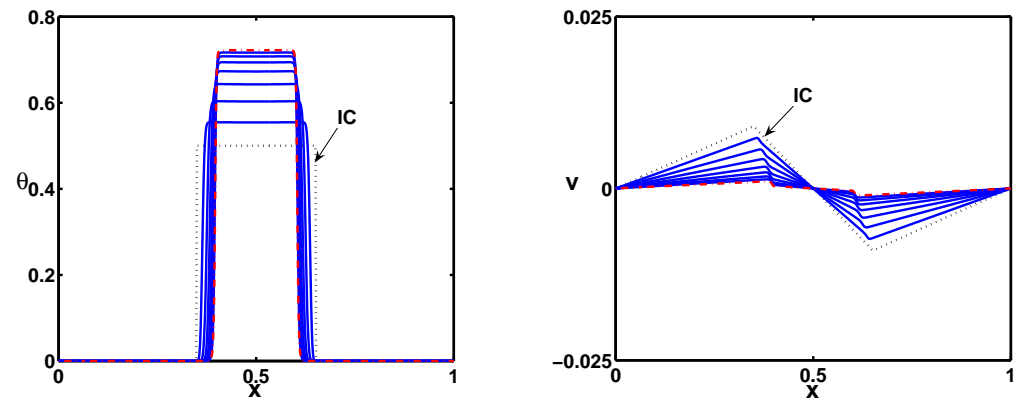
In this chapter, we have studied a one dimensional mixed model for tissue deformation. The linear stability analysis has been conducted which gives the range of the parameters for the stability and their relations with parameters. Many numerical experiments have been completed to support our stability analysis. Several different numerical methods have been studied. The first one is the standard first order upwind scheme for the volume fraction of cells and fibers θ and a central finite difference scheme for the velocity v . The second one is the standard high order (fifth order) WENO scheme with strong stability preserving optimal third order TVD Runge-Kutta method for the volume fraction of cells



(15)

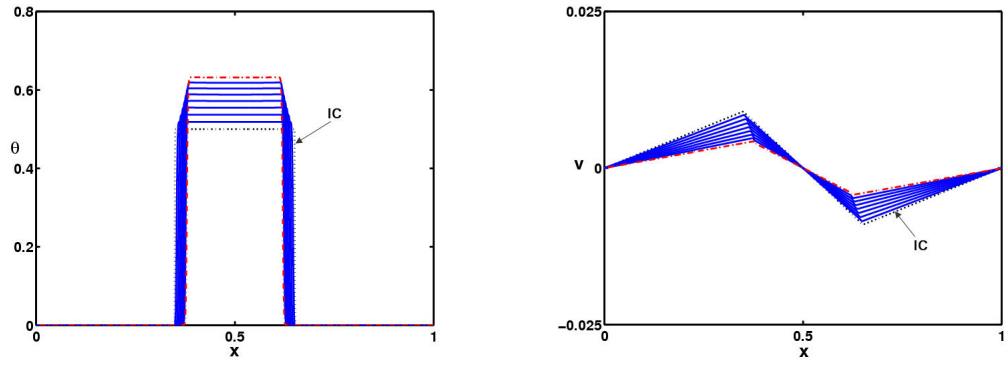


(16)

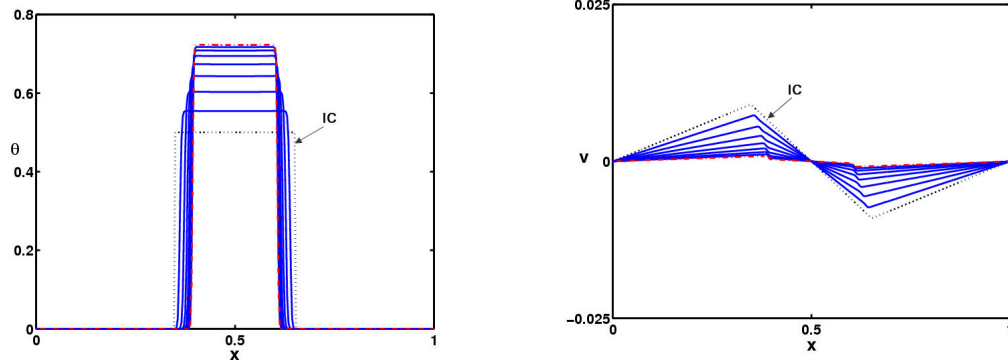


(17)

Figure 2.20: Computed solutions of (2.31), (2.32) along with IC in (2.6) and parameters in (2.28). (15) – the solutions of θ (left) and velocity (right) of Case 15 with $T = 5$ and $N = 640$, solved by the upwind scheme along with IIM; (16) – the solutions of θ (left) and velocity (right) of Case 16 under same conditions as Case 15 but solved by the upwind scheme along with CFDM; (17) – the solutions of θ (left) and velocity (right) of Case 17 similar to Case 16 but with $T = 15$.



(18)



(19)

Figure 2.21: Computed solutions of (2.31), (2.32) along with IC in (2.6) and parameters in (2.28) solved via the WENO-Roe scheme along with TVD Runge-Kutta method and CFDM. (18) – the solutions of θ (left) and velocity (right) of Case 18 with $T = 5$ and $N = 640$; (19) – the solutions of θ (left) and velocity (right) of Case 19 with $T = 15$ and $N = 320$.

and fibers θ and a central finite difference scheme for the velocity v . In the third approach, we used the immersed interface method to enforce the jump conditions. The three methods give qualitatively the same results, but the third method using IIM for velocity v eliminates non-physical oscillations by using the knowledge of the locations of the shocks.

Chapter 3

IIM for a Linear Advection Equation

3.1 A Linear Advection Equation

In this chapter, we consider a linear scalar advection equation (in 1D):

$$q_t + vq_x = 0, \quad 0 \leq x \leq L, \quad (3.1)$$

where q represents some physical quantity, v is the speed of the advection, $v = \text{constant}$. We choose $L = 1$ and $v = 1$ in the computations. We know that the analytical solution to this advection equation is

$$q(x, t) = q_0(x - vt), \quad (3.2)$$

no matter whether the initial condition q_0 is continuous or not. For this toy problem, we consider the cases that q is discontinuous, i.e., we impose nonhomogeneous jump conditions

$$[q] \neq 0, \quad [q_x] \neq 0, \quad \text{and/or} \quad [q_{xx}] \neq 0,$$

at $x = \alpha$.

To avoid other sources of error, for instance, improper boundary conditions, we use the analytical solution at $x = 0$ as the boundary conditions.

Three types of initial condition (IC) are considered in this study; see Fig. 3.1:

$$q(x, 0) = \begin{cases} 0, & \text{if } 0 \leq x \leq \alpha \\ 0.5, & \text{if } \alpha < x \leq L, \end{cases} \quad (3.3)$$

$$q(x, 0) = \begin{cases} 0.2 \sin(2\pi x), & \text{if } 0 \leq x \leq \alpha, \\ 0.5 + 0.2 \sin(2\pi x), & \text{if } \alpha < x \leq L. \end{cases} \quad (3.4)$$

and

$$q(x, 0) = \begin{cases} 0.2 \sin(\pi x), & \text{if } 0 \leq x \leq \alpha, \\ 0.5 + 0.2 \sin(2\pi x), & \text{if } \alpha < x \leq L. \end{cases} \quad (3.5)$$

In Fig. 3.1, the top left plot shows piecewise constant initial condition described in (3.3) with $[q] \neq 0$, $[q_x] = 0$ and $[q_{xx}] = 0$. The top right plot shows piecewise C^∞ initial condition described in (3.4) with $[q] \neq 0$ only. The bottom left plot shows piecewise C^∞ initial condition described in (3.5) with $[q] \neq 0$, $[q_x] \neq 0$ and $[q_{xx}] \neq 0$. The bottom right plot shows derivatives of the initial condition described in (3.5). The interfaces in all the plots are at $\alpha = 1/3$.

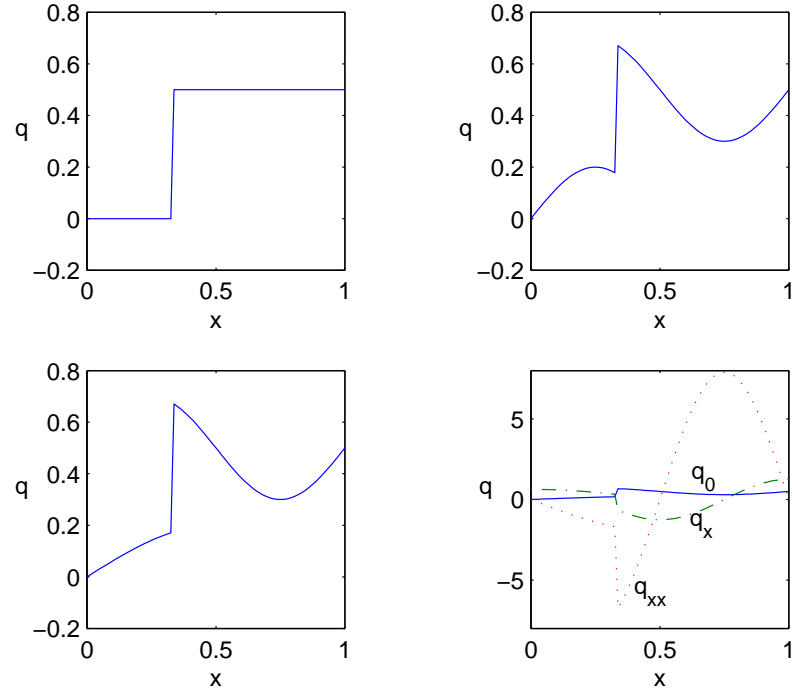


Figure 3.1: Initial condition for the computations. Top left: piecewise constant initial condition described in (3.3). Top right: piecewise C^∞ initial condition described in (3.4) with $[q] \neq 0$ only. Bottom left: piecewise C^∞ initial condition described in (3.5) with $[q] \neq 0$, $[q_x] \neq 0$ and $[q_{xx}] \neq 0$. Bottom right: derivatives of the initial condition described in (3.5).

Note that the jump conditions for the linear advection problem with the initial conditions described in (3.3) and (3.4) are the same:

$$[q] = 0.5, \quad [q_x] = 0, \quad [q_{xx}] = 0,$$

whereas the jump conditions for the linear advection problem with the initial condition described in (3.5) are all non-homogeneous:

$$\begin{aligned} [q] &= 0.5, & [q_x] &= 0.2\pi(2\cos(2\pi\alpha) - \cos(\pi\alpha)), \\ [q_{xx}] &= -0.2\pi^2(4\sin(2\pi\alpha) - \sin(\pi\alpha)). \end{aligned}$$

We discretize the space uniformly

$$0 = x_0 < x_1 < \cdots < x_{N-1} < x_N = L = 1, \quad (3.6)$$

where $x_i = x_0 + i\Delta x$.

There are many ways to solve a advection equation. We use the Lax-Wendroff method, known as a second order method, to illustrate our basic idea. The Lax-Wendroff method gives

$$Q_i^{n+1} = Q_i^n - \frac{v\Delta t}{2\Delta x}(Q_{i+1}^n - Q_{i-1}^n) + \frac{1}{2}\left(\frac{v\Delta t}{\Delta x}\right)^2(Q_{i-1}^n - 2Q_i^n + Q_{i+1}^n), \quad (3.7)$$

where $Q_i^n \approx Q(x_i, t^n)$, $t^n = n\Delta t$.

The Lax-Wendroff method is based on the Taylor series expansion. So the above difference expression for Q_i^{n+1} is valid for all the regular grid points, i.e., the solution is continuous in the interval of the three-point stencil. Suppose the interface lies in

$$x_J \leq \alpha < x_{J+1}.$$

Then the irregular grid points are J and $J+1$. See Fig. 3.2.

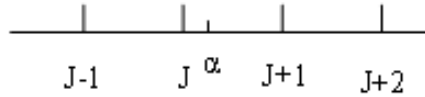


Figure 3.2: Diagram of the stencils for irregular grid points.

3.2 The Immersed Interface Method - Modification for Lax-Wendroff Method

We know that the Lax-Wendroff method works well to approximate Q_i^{n+1} to second order accuracy when the solution is smooth on the supporting stencil $\{x_{i-1}, x_i, x_{i+1}\}$. However, the central finite difference approximations in (3.7) for grid points J and $J+1$ yields no accuracy at all. Hence, we need to find correction terms to assure at least first order accuracy at those irregular grid points.

From the jump conditions, we have the following interface relations

$$\begin{aligned} q(\alpha^+, t) &= q(\alpha^-, t) + [q]_{\alpha, t}, \\ q_x(\alpha^+, t) &= q_x(\alpha^-, t) + [q_x]_{\alpha, t}, \\ q_{xx}(\alpha^+, t) &= q_{xx}(\alpha^-, t) + [q_{xx}]_{\alpha, t}. \end{aligned} \quad (3.8)$$

For the irregular grid point J , we use the following expression to approximate Q_J^{n+1}

$$\frac{Q_J^{n+1} - Q_J^n}{\Delta t} = -\frac{v}{2\Delta x}(Q_{J+1}^n - Q_{J-1}^n) + \frac{1}{2} \frac{v^2 \Delta t}{(\Delta x)^2}(Q_{i-1}^n - 2Q_i^n + Q_{i+1}^n) + C_J, \quad (3.9)$$

where C_J is the correction term to improve the order of accuracy for the approximation.

Consider the local truncation error at x_J :

$$\begin{aligned} T_J &= \frac{q(x_J, t^{n+1}) - q(x_J, t^n)}{\Delta t} + \frac{v}{2\Delta x}(q(x_{J+1}, t^n) - q(x_{J-1}, t^n)) \\ &\quad - \frac{1}{2} \frac{v^2 \Delta t}{(\Delta x)^2}(q(x_{J-1}, t^n) - 2q(x_J, t^n) + q(x_{J+1}, t^n)) - C_J. \end{aligned} \quad (3.10)$$

Consider two cases as shown in Fig. 3.3. In Case 1, the interface stays in the interval, $x_J \leq \alpha < x_{J+1}$, as time evolves from t^n to t^{n+1} . In case 2, the interface crosses the grid line at x_{J+1} at some time τ , $t^n < \tau < t^{n+1}$, i.e., the interface locates in the interval, $x_{J+1} \leq \alpha < x_{J+2}$, at time t^{n+1} .

Correction for Case 1 - no crossing (nc)

We first consider the case without crossing. Taylor series expansion at $x = \alpha$ and

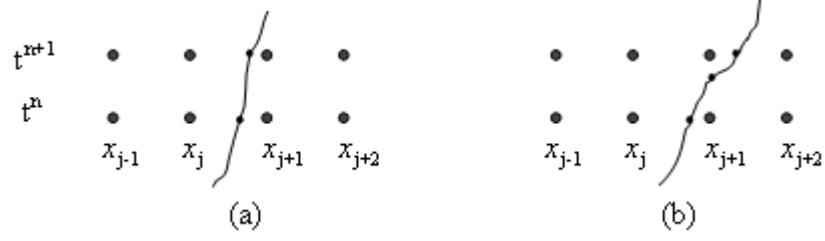


Figure 3.3: Diagram of the interface location. (a) no crossing, (b) crossing a grid line.

t^n gives

$$\begin{aligned}
T_J^{nc} = & \frac{1}{\Delta t} (q(\alpha^-, t^n) + q_x(\alpha^-, t^n)A_J + q_t(\alpha^-, t^n)\Delta t + q_{xx}(\alpha^-, t^n)B_J \\
& + \frac{1}{2}q_{tt}(\alpha^-, t^n)(\Delta t)^2 + q_{xt}(\alpha^-, t^n)A_J\Delta t + O((\Delta x)^3, (\Delta t)^3) \\
& - (q(\alpha^-, t^n) + q_x(\alpha^-, t^n)A_J + q_{xx}(\alpha^-, t^n)B_J) + O((\Delta x)^3) \\
& + \frac{v}{2\Delta x} (q(\alpha^+, t^n) + q_x(\alpha^+, t^n)A_{J+1} + q_{xx}(\alpha^+, t^n)B_{J+1} + O((\Delta x)^3) \\
& - (q(\alpha^-, t^n) + q_x(\alpha^-, t^n)A_{J-1} + q_{xx}(\alpha^-, t^n)B_{J-1} + O((\Delta x)^3))) \\
& - \frac{1}{2} \frac{v^2 \Delta t}{(\Delta x)^2} (q(\alpha^-, t^n) + q_x(\alpha^-, t^n)A_{J-1} + q_{xx}(\alpha^-, t^n)B_{J-1} + O((\Delta x)^3) \\
& - 2(q(\alpha^-, t^n) + q_x(\alpha^-, t^n)A_J + q_{xx}(\alpha^-, t^n)B_J + O((\Delta x)^3)) \\
& + q(\alpha^+, t^n) + q_x(\alpha^+, t^n)A_{J+1} + q_{xx}(\alpha^+, t^n)B_{J+1} + O((\Delta x)^3)) - C_J^{nc},
\end{aligned}$$

where $A_k = x_k - \alpha$, $B_k = (x_k - \alpha)^2/2$ for $k = J-1, J, J+1$, and $J+2$.

Applying the interface relations in (3.8) gives

$$\begin{aligned}
T_J^{nc} = & q_t(\alpha^-, t^n) + \frac{1}{2}q_{tt}(\alpha^-, t^n)(\Delta t) + q_{xt}(\alpha^-, t^n)A_J + O((\Delta x)^2, (\Delta t)^2) \\
& + \frac{v}{2\Delta x} ([q]_{\alpha,n} + [q_x]_{\alpha,n}A_{J+1} + [q_{xx}]_{\alpha,n}B_{J+1}) + vq_x(\alpha^-, t^n) + vq_{xx}(\alpha^-, t^n)A_J \\
& - \frac{1}{2} \frac{v^2 \Delta t}{(\Delta x)^2} ([q]_{\alpha,n} + [q_x]_{\alpha,n}A_{J+1} + [q_{xx}]_{\alpha,n}B_{J+1}) - \frac{1}{2}v^2 \Delta t q_{xx}(\alpha^-, t^n) - C_J^{nc} \\
= & \frac{1}{2} \frac{\Delta x - v\Delta t}{(\Delta x)^2} v ([q]_{\alpha,n} + [q_x]_{\alpha,n}A_{J+1} + [q_{xx}]_{\alpha,n}B_{J+1}) - C_J^{nc} + O((\Delta x)^2, (\Delta t)^2).
\end{aligned}$$

So it will suffice for second order accuracy if the correction term for grid points J is

$$C_J^{nc} = \frac{1}{2} \frac{\Delta x - v\Delta t}{(\Delta x)^2} v ([q]_{\alpha,n} + [q_x]_{\alpha,n}A_{J+1} + [q_{xx}]_{\alpha,n}B_{J+1}). \quad (3.11)$$

Similarly, at the grid point $J+1$ we obtain the correction term as follows

$$C_{J+1}^{nc} = \frac{1}{2} \frac{\Delta x + v\Delta t}{(\Delta x)^2} v ([q]_{\alpha,n} + [q_x]_{\alpha,n}A_J + [q_{xx}]_{\alpha,n}B_J). \quad (3.12)$$

Correction for Case 2 - crossing

Now we consider the case that the interface location cross the grid line at some time τ as time evolves from t^n to t^{n+1} . In our numeric experiments, we just consider the cases with $v > 0$, i.e., wave travels to the right. For the cases that have $v < 0$, all the processes are similar. Looking at the diagram part (b) in Fig. 3.3, we understand that crossing does not affect the correction for grid point J when the wave travels to the right, i.e.,

$$C_J^c = C_J^{mc} = \frac{1}{2} \frac{\Delta x - v\Delta t}{(\Delta x)^2} v ([q]_{\alpha,n} + [q_x]_{\alpha,n} A_{J+1} + [q_{xx}]_{\alpha,n} B_{J+1}). \quad (3.13)$$

The grid crossing only affects the grid $J + 1$. Here we use l instead of $J + 1$ to simplify the expression. Taylor series expansion at $x = \alpha$ and τ gives

$$\begin{aligned} q(x_l, t^{n+1}) &= q(x_l, \tau^+) + q_t(x_l, \tau^+)(t^{n+1} - \tau) + \frac{1}{2} q_{tt}(x_l, \tau^+)(t^{n+1} - \tau)^2 + O((\Delta t)^3) \\ &= q(x_l, \tau^+) - v q_x(x_l, \tau^+)(t^{n+1} - \tau) + \frac{1}{2} v^2 q_{xx}(x_l, \tau^+)(t^{n+1} - \tau)^2 + O((\Delta t)^3) \\ &= q(\alpha^-, t^n) + q_x(\alpha^-, t^n) A_l + q_t(\alpha^-, t^n)(\tau - t^n) + q_{xx}(\alpha^-, t^n) B_l \\ &\quad + \frac{1}{2} q_{tt}(\alpha^-, t^n)(\tau - t^n)^2 + q_{xt}(\alpha^-, t^n) A_l(\tau - t^n) \\ &\quad - v(q_x(\alpha^-, t^n) + q_{xx}(\alpha^-, t^n) A_l + q_{xt}(\alpha^-, t^n)(\tau - t^n))(t^{n+1} - \tau) \\ &\quad + \frac{1}{2} v^2 q_{xx}(\alpha^-, t^n)(t^{n+1} - \tau)^2 + O((\Delta x)^3, (\Delta t)^3) \\ &= q(\alpha^-, t^n) + q_x(\alpha^-, t^n) A_l - v q_x(\alpha^-, t^n)(\tau - t^n) + q_{xx}(\alpha^-, t^n) B_l \\ &\quad + \frac{v^2}{2} q_{xx}(\alpha^-, t^n)(\tau - t^n)^2 - v q_{xx}(\alpha^-, t^n) A_l(\tau - t^n) \\ &\quad - v(q_x(\alpha^-, t^n) + q_{xx}(\alpha^-, t^n) A_l - v q_{xx}(\alpha^-, t^n)(\tau - t^n))(t^{n+1} - \tau) \\ &\quad + \frac{1}{2} v^2 q_{xx}(\alpha^-, t^n)(t^{n+1} - \tau)^2 + O((\Delta x)^3, (\Delta t)^3), \\ &= q(\alpha^-, t^n) + q_x(\alpha^-, t^n) A_l - v q_x(\alpha^-, t^n) \Delta t + q_{xx}(\alpha^-, t^n) B_l \\ &\quad + \frac{v^2}{2} q_{xx}(\alpha^-, t^n)(\Delta t)^2 - v q_{xx}(\alpha^-, t^n) A_l \Delta t + O((\Delta x)^3, (\Delta t)^3), \\ q(x_l, t^n) &= q(\alpha^+, t^n) + q_x(\alpha^+, t^n) A_l + q_{xx}(\alpha^+, t^n) B_l + O((\Delta x)^3) \end{aligned}$$

Applying those expressions and the interface relations in (3.8) gives

$$\begin{aligned}
T_l^c &= \frac{q(x_l, t^{n+1}) - q(x_l, t^n)}{\Delta t} + \frac{v}{2\Delta x}(q(x_{l+1}, t^n) - q(x_{l-1}, t^n)) \\
&\quad - \frac{1}{2} \frac{v^2 \Delta t}{(\Delta x)^2} (q(x_{l-1}, t^n) - 2q(x_l, t^n) + q(x_{l+1}, t^n)) - C_l^c, \\
&= -\frac{1}{\Delta t} ([q]_{\alpha,n} + [q_x]_{\alpha,n} A_l + [q_{xx}]_{\alpha,n} B_l) + v([q_x]_{\alpha,n} + [q_{xx}]_{\alpha,n} A_l - \frac{v\Delta t}{2} [q_{xx}]_{\alpha,n}) \\
&\quad + \frac{1}{2} \frac{\Delta x + v\Delta t}{(\Delta x)^2} v([q]_{\alpha,n} + [q_x]_{\alpha,n} A_{l-1} + [q_{xx}]_{\alpha,n} B_{l-1}) - C_l^c + O((\Delta x)^2, (\Delta t)^2)
\end{aligned}$$

So it will suffice for second order accuracy if the correction term for grid points $J + 1$ is

$$\begin{aligned}
C_{J+1}^c &= -\frac{1}{\Delta t} ([q]_{\alpha,n} + [q_x]_{\alpha,n} A_{J+1} + [q_{xx}]_{\alpha,n} B_{J+1}) \\
&\quad + v([q_x]_{\alpha,n} + [q_{xx}]_{\alpha,n} A_{J+1} - \frac{v\Delta t}{2} [q_{xx}]_{\alpha,n}) \\
&\quad + \frac{1}{2} \frac{\Delta x + v\Delta t}{(\Delta x)^2} v([q]_{\alpha,n} + [q_x]_{\alpha,n} A_J + [q_{xx}]_{\alpha,n} B_J).
\end{aligned} \tag{3.14}$$

For the cases with $[q_x] = [q_{xx}] = 0$, (3.11), (3.12) and (3.14) can be simplified as follows:

$$\begin{aligned}
C_J^{mc} &= C_J^c = \frac{1}{2} \frac{\Delta x - v\Delta t}{(\Delta x)^2} v[q]_{\alpha,n}, \\
C_{J+1}^{mc} &= \frac{1}{2} \frac{\Delta x + v\Delta t}{(\Delta x)^2} v[q]_{\alpha,n}, \\
C_{J+1}^c &= -\frac{1}{\Delta t} [q]_{\alpha,n} + \frac{1}{2} \frac{\Delta x + v\Delta t}{(\Delta x)^2} v[q]_{\alpha,n}.
\end{aligned}$$

3.3 Numerical Examples

We take $L = 1$ for simplicity in this section. Three numerical computations have been completed to demonstrate how well the immersed interface method works for linear advection equations. In all cases presented here, we use solid lines to represent the exact solutions and the symbol 'o' to represent the numerical solutions. The initial interface is located at $\alpha = 1/3$.

Example 1

First we consider the piecewise constant initial condition described in (3.3) for this example.

Fig. 3.4 shows the exact solution and numerical solutions at $t = 0.1$, $t = 0.2$, $t = 0.3$, and $t = 0.4$ respectively. The immersed interface method keeps the interface sharp as time evolves. No errors are observed in the computed solution at all the grid points.

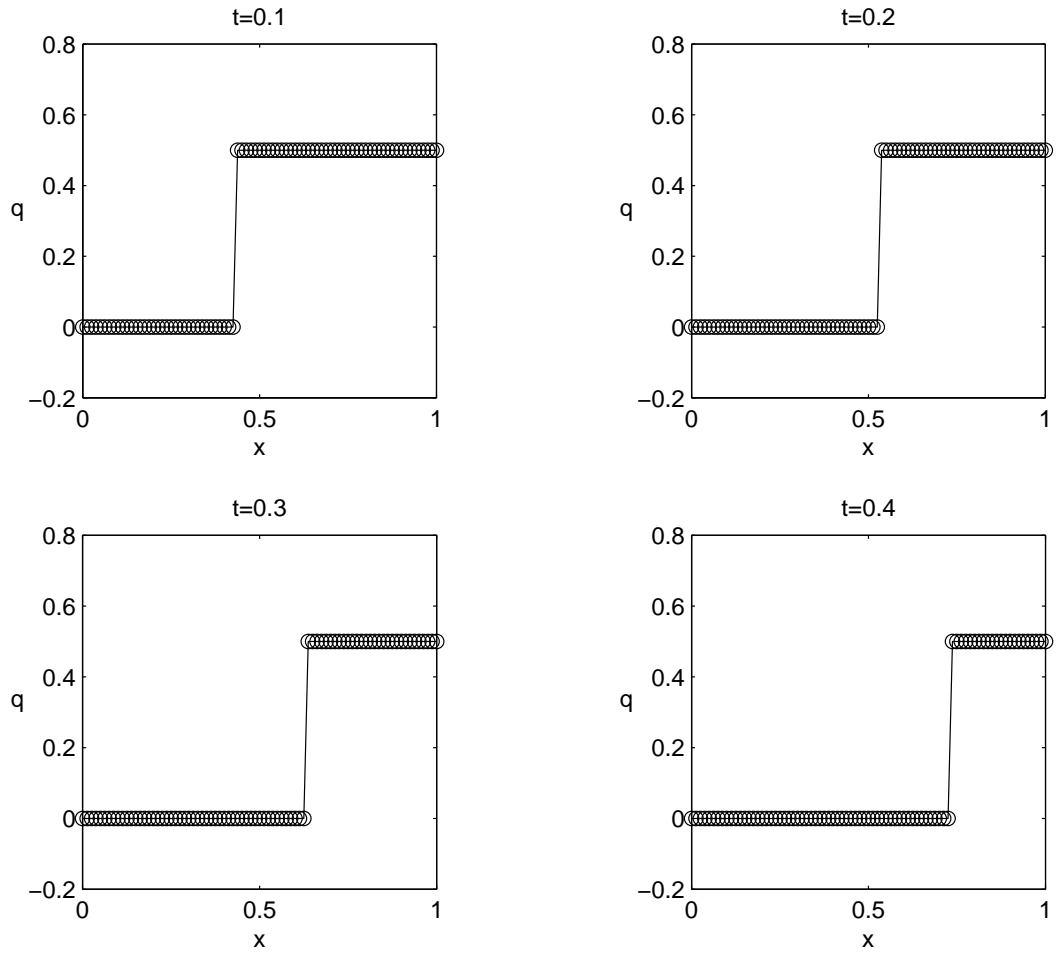


Figure 3.4: 1-D linear advection with $v = 1$ and initial condition in (3.4). Comparison of computed solution and exact solution at $t = 0.1, 0.2, 0.3$, and 0.4 respectively. The solid line represents the exact solution and the symbol 'o' represents the numerical solutions. The mesh size for the computation is $h = 1/80$.

Example 2

In this example, we consider the cases with the piecewise C^∞ initial condition described in (3.4). We employ the immersed interface method (modified Lax-Wendroff method) to compute the partial differential equation for $T = 0.5$. Fig. 3.5 shows the exact solution and computed solution at $t = 0.1$, $t = 0.2$, $t = 0.3$, and $t = 0.4$ respectively. Table 3.1 shows the grid refinement result at $T = 0.5$ which shows second order accuracy in L_∞ , L_1 and L_2 norms obtained when the immersed interface method is used. The linear regression analysis of the convergence order in log-log scale for Example 2 at $t = 0.5$ is shown in Fig. 3.6. The average convergence order for the L_{infy} , L_1 and L_2 errors are 2, 2.02 and 2.03, respectively.

Table 3.1: Accuracy of IIM for solving a linear advection equation with initial condition in (3.4).

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio	L_2 error	L_2 ratio
40	1.54e-3	—	6.81e-4	—	8.22e-4	—
80	3.91e-4	3.95	1.64e-4	4.15	2.00e-4	4.10
160	9.73e-5	4.02	4.02e-5	4.09	4.93e-5	4.07
320	2.42e-5	4.01	9.93e-6	4.04	1.22e-5	4.03
640	6.06e-6	4.00	2.47e-6	4.02	3.04e-6	4.01
1280	1.51e-6	4.00	6.16e-7	4.01	7.59e-7	4.01

Example 3

In this example, we consider the cases with the piecewise C^∞ initial condition described in (3.5) with which the solution of the linear advection equation has jumps in the solution, the first and second derivatives of the solution. We employ the immersed interface method (modified Lax-Wendroff method) to compute the partial differential equation for $T = 0.5$. Fig. 3.7 shows the exact solution and computed solution at $t = 0.1$, $t = 0.2$, $t = 0.3$, and $t = 0.4$ respectively. Table 3.2 shows the grid refinement result at $T = 0.5$ which shows second order accuracy in L_∞ , L_1 and L_2 norms obtained when the immersed interface method is used. The linear regression analysis of the convergence order in log-log scale for Example 3 at $t = 0.5$ is shown in Fig. 3.8. The average convergence order for the L_{infy} , L_1 and L_2 errors are 1.99, 2.04 and 2.03, respectively.

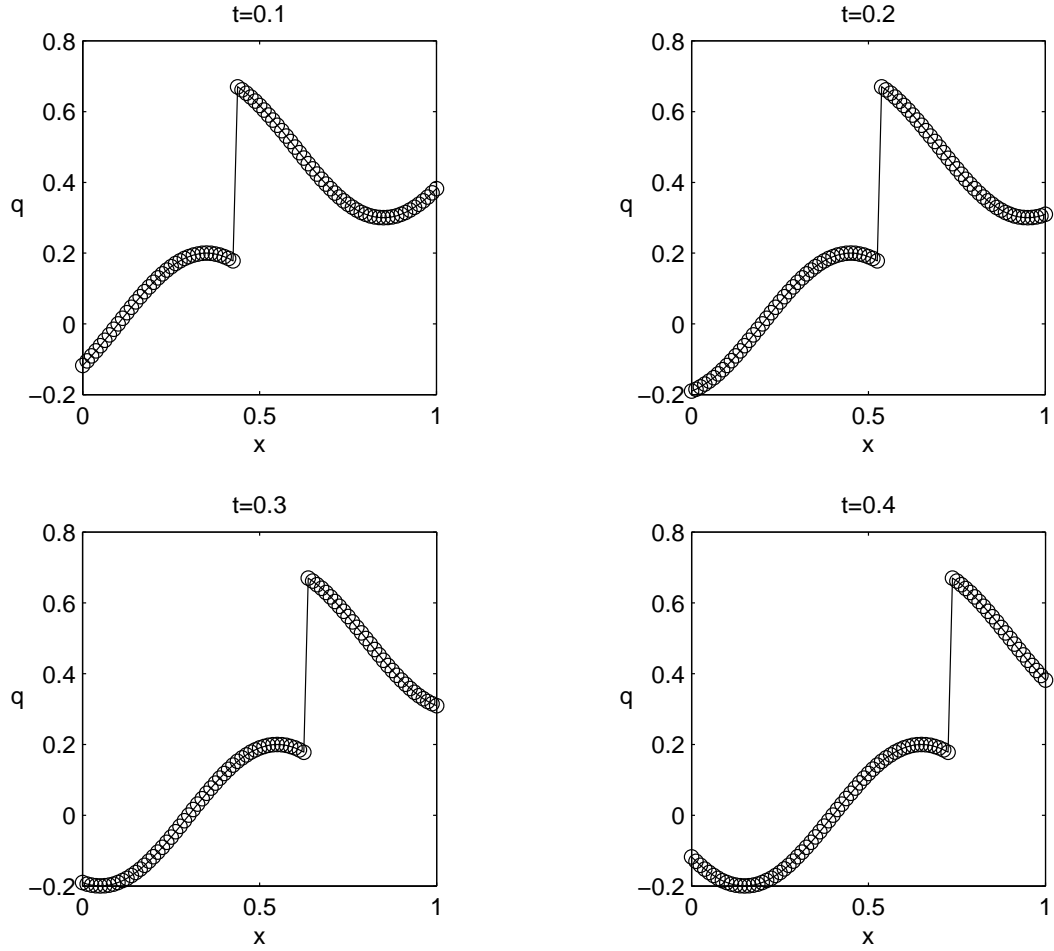


Figure 3.5: 1-D linear advection with $v = 1$ and initial condition in (3.4). Comparison of computed solution and exact solution at $t = 0.1, 0.2, 0.3$, and 0.4 respectively. The solid line represents the exact solution and the symbol 'o' represents the numerical solutions. The mesh size for the computation is $h = 1/80$.

Table 3.2: Accuracy of IIM for solving a linear advection equation with initial condition in (3.5).

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio	L_2 error	L_2 ratio
40	2.50e-3	—	4.45e-4	—	7.67e-4	—
80	6.48e-4	3.89	1.02e-4	4.37	1.80e-4	4.27
160	1.62e-4	4.00	2.47e-5	4.12	4.37e-5	4.11
320	4.03e-5	4.02	6.11e-6	4.04	1.08e-5	4.04
640	1.01e-5	4.00	1.51e-6	4.04	2.68e-6	4.03
1280	2.52e-6	4.00	3.76e-7	4.03	6.67e-7	4.02

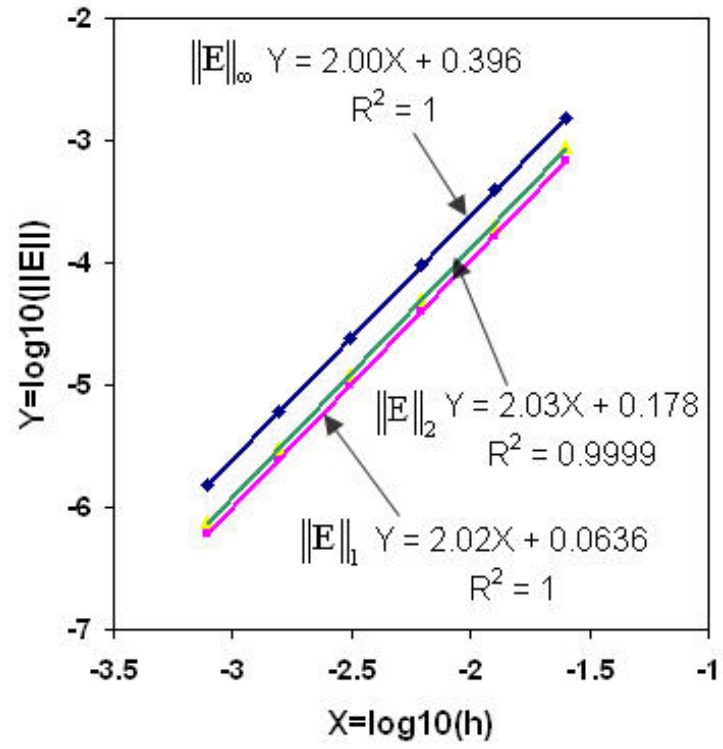


Figure 3.6: Linear regression analysis of the convergence order in log-log scale for 1-D linear advection with $v = 1$ and initial condition in (3.4) at $t = 0.5$. The average convergence order for the L_{∞} , L_1 and L_2 errors are 2, 2.02 and 2.03, respectively.

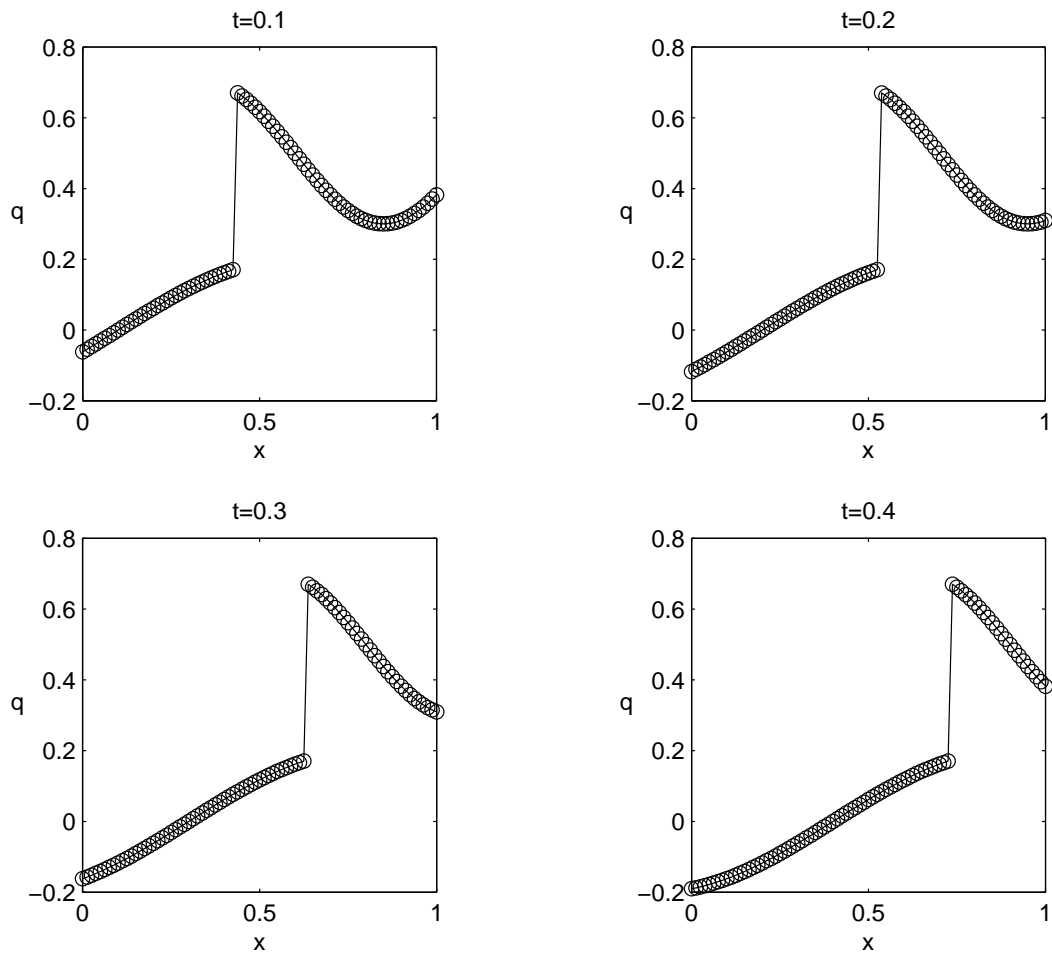


Figure 3.7: 1-D linear advection with $v = 1$ and initial condition in (3.5). Comparison of computed solution and exact solution at $t = 0.1, 0.2, 0.3$, and 0.4 respectively. The solid line represents the exact solution and the symbol 'o' represents the numerical solutions. The mesh size for the computation is $h = 1/80$.

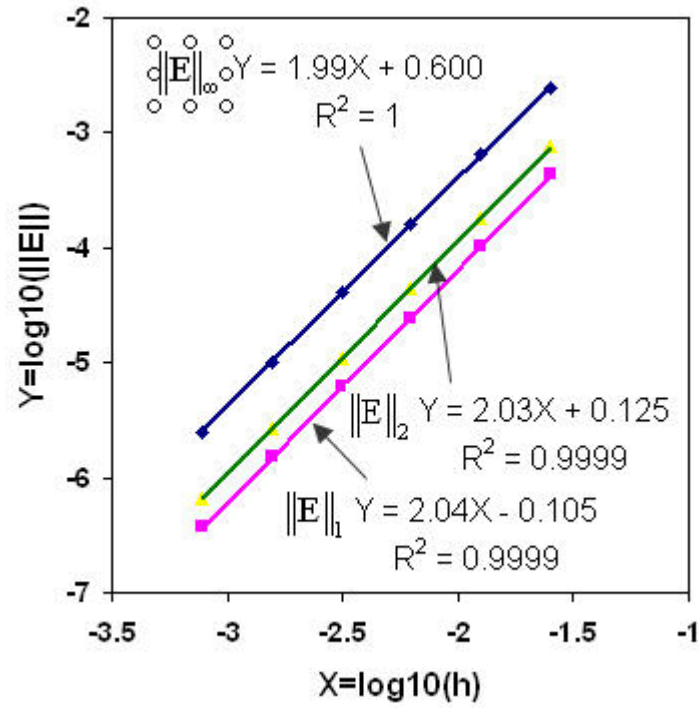


Figure 3.8: Linear regression analysis of the convergence order in log-log scale for 1-D linear advection with $v = 1$ and initial condition in (3.5) at $t = 0.5$. The average convergence order for the L_{∞} , L_1 and L_2 errors are 1.99, 2.04 and 2.03, respectively.

3.4 Conclusions

In this chapter, we have studied the accuracy of the immersed interface method for solving a one dimensional linear advection equation with three different types of discontinuous initial conditions. The well-known Lax-Wendroff method is used for the regular grid points. We have derived the correction terms for the irregular grid points near the moving interface and the corrections due to the grid crossing as time evolves from t^n to t^{n+1} whenever it is applicable. By enforcing the jump conditions across the interfaces, the immersed interface method improves both the resolution and accuracy of the numerical solution to the linear advection problem with moving interface comparing to other conventional methods. For the advection problem with piecewise constant initial condition, no errors at grid points have been observed. The other two numerical experiments with piecewise C^∞ initial conditions have shown that second order accuracy is obtained.

Chapter 4

IIM for a 1D Fluid Mixture Model of Tissue Deformation with Constant Physical Parameters

One of the main interests of this study is to use the immersed interface method to solve the fluid mixture model of tissue deformation with discontinuous initial conditions. As we discussed in Chapter 1, the immersed interface method is a fine method for solving a variety of interface problems. Here we extend this method to handle some nonlinear system combined with hyperbolic and elliptic equations.

Note that prior knowledge of the locations of the shocks (interfaces) is required to apply the immersed interface method. The proper jump conditions are extremely important in formulating the special finite difference scheme at the irregular grid points near the interfaces. For a general fluid mixture model of tissue deformations with four physical parameters, the drag coefficient φ , the contractility coefficient ψ , the swelling coefficient σ , and the viscosity coefficient of the cell-fiber fraction M , may depend on time, space, and θ . The system is very complicated to deduce the necessary jump conditions across the interface from the partial differential equations to meet the need when the immersed interface method is applied in order to achieve second order accurate solutions. Hence, in this chapter, we consider implementing the immersed interface method to solve the most simple case, a one dimensional fluid mixture model of tissue deformation with constant

physical parameters:

$$\begin{aligned}\theta_t + (v\theta)_x &= 0, \\ v_{xx} + (\psi\theta)_x + (\sigma \ln(1 - \theta))_x &= 0,\end{aligned}\tag{4.1}$$

with a periodic boundary condition

$$\theta(0, t) = \theta(1, t), v(0, t) = v(1, t),$$

and a piecewise constant initial condition

$$\theta_0(x) = \begin{cases} \theta_u, & \text{if } \alpha_1 \leq x \leq \alpha_2, \\ \theta_l, & \text{if } 0 \leq x < \alpha_1 \text{ or } \alpha_2 < x \leq 1, \end{cases}\tag{4.2}$$

where we use θ_l for the smaller constant (lower), and θ_u for the larger constant (upper), and α_1 and α_2 indicate the locations of two interfaces, ψ and σ are constants.

4.1 Numerical Methods

In Chapter 2, we have already shown the work required to apply the immersed interface method only to the elliptic equation for the velocity v . So we will not repeat that explanation in this chapter. We focus here on how to apply the immersed interface method to the hyperbolic equation. Applying the immersed interface method to only the elliptic equation gives more accurate results than applying the conventional finite difference methods to the system for both θ and v and eliminates non-physical oscillations (the Gibbs phenomenon). But the order of accuracy is poor (first order accuracy when the errors are measured 0.1 away from the interfaces) when the shock capturing method, WENO, along with the Runge-Kutta method for time evolution, is used to solve the hyperbolic equation for the volume fraction of the cell and fiber phase θ .

4.1.1 The Conventional Finite Difference Method for Regular Grid Points

As we discussed in Chapter 1, we cannot take advantage of the conventional two-level second order accurate methods, such as Lax-Wendroff method, or Beam-Warming method, to apply to the hyperbolic equation at the regular grid points due to a lack of information of v_t . In order to achieve a second order algorithm, we need a second order method for the regular grid points. The three-level method, the leapfrog method [80, 81, 14]

shown in (4.3), and two-step MacCormack's method [86, 13], shown in (4.4), respectively, are used for the regular grid points:

$$\frac{\Theta_i^{n+1} - \Theta_i^{n-1}}{2\Delta t} = -\frac{(\Theta V)_{i+1}^n - (\Theta V)_{i-1}^n}{2\Delta x}, \quad (4.3)$$

$$\begin{aligned} \Theta_i^* &= \Theta_i^n - \frac{\Delta t}{\Delta x}((\Theta V)_{i+1}^n - (\Theta V)_i^n), \\ \Theta_i^{**} &= \Theta_i^* - \frac{\Delta t}{\Delta x}((\Theta V)_i^* - (\Theta V)_{i-1}^*), \\ \Theta_i^{n+1} &= \frac{1}{2}(\Theta_i^n + \Theta_i^{**}). \end{aligned} \quad (4.4)$$

Note that we do not have information at $-\Delta t$. So we need a two-level method to generate the numerical initial data when we use the leapfrog method. We use the first order upwind scheme for the first time step.

$$\Theta_j^{n+1} = \begin{cases} \Theta_j^n - \frac{\Delta t}{h}((\Theta V)_j^n - (\Theta V)_{j-1}^n), & \text{if } V_j^n > 0, \\ \Theta_j^n - \frac{\Delta t}{h}((\Theta V)_{j+1}^n - (\Theta V)_j^n), & \text{if } V_j^n < 0, \end{cases} \quad (4.5)$$

When the two-step MacCormack method is used, note that we need to solve the elliptic equation at each step to maintain second order accuracy for the method. The method would be of first order accuracy if we use V^n instead of V^* at the second step. For the proof, see Appendix A. This implies extra work to solve the system of difference equations for velocity at each time step. When the viscosity coefficient of the cell-fiber fraction is constant, as we study in this chapter, this does not increase computing cost a lot since we can find the inverse of the coefficient matrix for the system of difference equations once at the beginning and do the matrix-vector multiplication for later time steps. However, when the viscosity coefficient of the cell-fiber fraction depends on θ , we need to find the inverse of the coefficient matrix for the system of difference equations at each time step. Then MacCormack's method doubles the cost of the computation.

4.1.2 Applying the Immersed Interface Method

We apply the immersed interface method to the irregular grid points based on the methods used for the regular grid points.

Recall that for the given system in (4.1) with chosen physical parameters and the initial condition, the square-wave pulse in θ increases as time evolves (see numerical examples in Chapter 2). This implies that the jump conditions change with time. We

use $[\theta]_{k,n}$, $[\theta_t]_{k,n}$ and $[\theta_{tt}]_{k,n}$ to represent the jump conditions of θ and its first and second derivatives at interface α_k and time t^n . So the interface relations are

$$\begin{aligned}\theta(\alpha_k^+, t^n) &= \theta(\alpha_k^-, t^n) + [\theta]_{k,n}, \\ \theta_t(\alpha_k^+, t^n) &= \theta_t(\alpha_k^-, t^n) + [\theta_t]_{k,n}, \\ (\theta v)_x(\alpha_k^+, t^n) &= (\theta v)_x(\alpha_k^-, t^n) + [(\theta v)_x]_{k,n}, \\ \theta_{tt}(\alpha_k^+, t^n) &= \theta_{tt}(\alpha_k^-, t^n) + [\theta_{tt}]_{k,n}.\end{aligned}\tag{4.6}$$

To maintain accuracy for the volume fraction of cell and fiber phase θ near the discontinuities, we apply the immersed interface method. The idea of the immersed interface method mainly counts on Taylor expansion and builds jump conditions into the methods for the irregular grid points near the interface. Note that correction terms to the conventional numerical methods for the irregular grid points depend on those methods. The correction terms are found via the Taylor series expansion by enforcing the jump conditions across the interfaces.

Consider the upwind scheme. Suppose θ_i^n and v_i^n , denoting the θ and v at the grid point i and time t^n , are known for all i in the domain. And suppose the interfaces are located at $x = \alpha_i$, $i = 1, 2$, which lie in the intervals between x_{J_i} and x_{J_i+1} at time t^n ; see Fig. 4.1. There are three possible cases as time evolves: no crossing, crossing to left or crossing to the right.

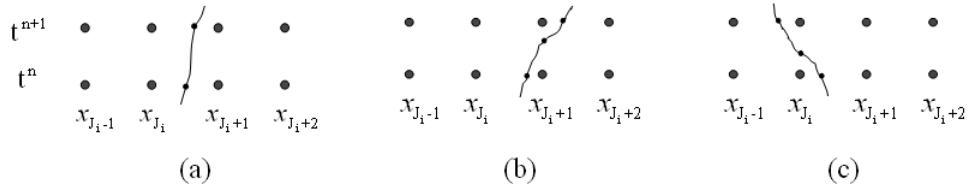


Figure 4.1: Diagram of the interface location for a two-level method. (a) no crossing, (b) crossing a grid line as $\alpha_i(t)$ increases with time, (c) crossing a grid line as $\alpha_i(t)$ decreases with time.

The irregular grid points for the upwind scheme are J_i or $J_i + 1$ at time t^n .

$$\frac{\Theta_{l+1}^{n+1} - \Theta_{l+1}^n}{\Delta t} = -\frac{1}{\Delta x}((\Theta V)_{l+1}^n - (\Theta V)_j^n) + C_{l+1}, \quad \text{if } V_{l+1}^n > 0, \tag{4.7}$$

$$\frac{\Theta_l^{n+1} - \Theta_l^n}{\Delta t} = -\frac{1}{\Delta x}((\Theta V)_{l+1}^n - (\Theta V)_l^n) + C_l, \quad \text{if } V_l^n < 0, \tag{4.8}$$

where $l = J_i$.

In order to find the correction terms we do the Taylor series expansion at the interfaces. Consider the irregular point at l first. The local truncation error is

$$T_l = \frac{\theta(x_l, t^{n+1}) - \theta(x_l, t^n)}{\Delta t} + \frac{1}{\Delta x}((\theta v)(x_{l+1}, t^n) - (\theta v)(x_l, t^n)) - C_l, \quad \text{if } V_l^n < 0, \quad (4.9)$$

We want to make the magnitude of the truncation error as small as possible by choosing the correction term. In order to obtain second order global accuracy, we need to ensure an $O(\Delta x)$ local truncation error at those irregular points [38]. Consider the Case (a) in Fig. 4.1 first. Expanding all the terms in (4.9) at α_k and t^n and applying the interface relations in (4.6) gives

$$\begin{aligned} T_l^{nc} &= \frac{1}{\Delta t}(\theta(\alpha_k^-, t^n) + \theta_x(\alpha_k^-, t^n)(x_l - \alpha_k) + \theta_t(\alpha_k^-, t^n)\Delta t + O((\Delta x)^2, (\Delta t)^2)) \\ &\quad - (\theta(\alpha_k^-, t^n) + \theta_x(\alpha_k^-, t^n)(x_l - \alpha_k) + O((\Delta x)^2)) \\ &\quad + \frac{1}{\Delta x}(((\theta v)(\alpha_k^+, t^n) + (\theta v)_x(\alpha_k^+, t^n)(x_{l+1} - \alpha_k) + O((\Delta x)^2)) \\ &\quad - ((\theta v)(\alpha_k^-, t^n) + (\theta v)_x(\alpha_k^-, t^n)(x_l - \alpha_k) + O((\Delta x)^2))) - C_l^{nc} \\ &= \theta_t(\alpha_k^-, t^n) + \frac{1}{\Delta x}[\theta v]_{k,n} + \frac{x_{l+1} - \alpha_k}{\Delta x}[(\theta v)_x]_{k,n} + (\theta v)_x(\alpha_k^-, t^n) - C_l^{nc} + O(\Delta x, \Delta t) \\ &= \frac{1}{\Delta x}[\theta v]_{k,n} + \frac{x_{l+1} - \alpha_k}{\Delta x}[(\theta v)_x]_{k,n} - C_l^{nc} + O(\Delta x, \Delta t), \end{aligned}$$

So it will suffice for first order accuracy if the correction term for grid points l is

$$C_l^{nc} = \frac{1}{\Delta x}[\theta v]_{k,n} + \frac{x_{l+1} - \alpha_k}{\Delta x}[(\theta v)_x]_{k,n}. \quad (4.10)$$

We get $O(\Delta x, \Delta t)$ for the local truncation error at the irregular grid point x_l .

Likewise, we can find the correction term for the irregular grid point x_{l+1} when $V_{l+1}^n > 0$ by analyzing the local truncation error at that point if no grid line crossing happens:

$$C_{l+1}^{nc} = \frac{1}{\Delta x}[\theta v]_{k,n} + \frac{x_l - \alpha_k}{\Delta x}[(\theta v)_x]_{k,n}. \quad (4.11)$$

Now consider the cases where grid line crossing happens as time evolves from t^n to t^{n+1} . Note that Case (b) in Fig. 4.1 happens only when $V_{l+1}^n > 0$ and Case (c) happens only when $V_{l+1}^n < 0$. Again, using Taylor series expansion at α_k and t^n and enforcing the jump conditions across the interfaces, we obtain the following correction terms for l (Case (c)) and $l+1$ (Case (b))

$$C_l^c = \frac{1}{\Delta t}[\theta]_{k,n} + \frac{x_l - \alpha_k}{\Delta t}[\theta_x]_{k,n} + \frac{1}{\Delta x}[\theta v]_{k,n} + \frac{x_l - \alpha_k}{\Delta x}[(\theta v)_x]_{k,n}, \quad (4.12)$$

$$C_{l+1}^c = -\frac{1}{\Delta t}[\theta]_{k,n} - \frac{x_{l+1} - \alpha_k}{\Delta t}[\theta_x]_{k,n} + \frac{1}{\Delta x}[\theta v]_{k,n} + \frac{x_{l+1} - \alpha_k}{\Delta x}[(\theta v)_x]_{k,n}. \quad (4.13)$$

Note that all the analysis above for the correction terms for the irregular grid points when the upwind scheme is used are suitable when MacCormack's method is used. We need to use (4.10) or (4.12) for the first step and (4.11) or (4.13) for the second step as time evolves from t^n to t^{n+1} .

The leapfrog method is a three-level method. Hence there are more possible crossing cases than a two-level method; see Fig. 4.2. For Case (a), no crossing, the correction comes from the space discretization. For Cases (b) - (g), different possible crossings happen, and the correction comes from both the space and time discretization.

The irregular grid points for the leapfrog method are J_i and $J_i + 1$ at time t^n .

$$\frac{\Theta_l^{n+1} - \Theta_l^{n-1}}{2\Delta t} = -\frac{1}{2\Delta x}((\Theta V)_{l+1}^n - (\Theta V)_{l-1}^n) + C_l, \quad (4.14)$$

where $l = J_i, J_i + 1$. The local truncation error at l is

$$T_l = \frac{\theta(x_l, t^{n+1}) - \theta(x_l, t^{n-1})}{2\Delta t} + \frac{1}{2\Delta x}((\theta v)(x_{l+1}, t^n) - (\theta v)(x_{l-1}, t^n)) - C_l.$$

Expanding all the terms and applying the interface relations for $l = J_i$ gives

$$\begin{aligned} T_l^{(a)} &= \frac{1}{2\Delta t}(\theta(\alpha_k^-, t^n) + \theta_x(\alpha_k^-, t^n)(x_l - \alpha_k) + \theta_t(\alpha_k^-, t^n)\Delta t + \frac{1}{2}\theta_{xx}(\alpha_k^-, t^n)(x_l - \alpha_k)^2 \\ &\quad + \frac{1}{2}\theta_{tt}(\alpha_k^-, t^n)(\Delta t)^2 + \theta_{xt}(\alpha_k^-, t^n)(x_l - \alpha_k)\Delta t) \\ &\quad - (\theta(\alpha_k^-, t^n) + \theta_x(\alpha_k^-, t^n)(x_l - \alpha_k) - \theta_t(\alpha_k^-, t^n)\Delta t + \frac{1}{2}\theta_{xx}(\alpha_k^-, t^n)(x_l - \alpha_k)^2 \\ &\quad + \frac{1}{2}\theta_{tt}(\alpha_k^-, t^n)(\Delta t)^2 - \theta_{xt}(\alpha_k^-, t^n)(x_l - \alpha_k)\Delta t) + O((\Delta x)^2, (\Delta t)^2) \\ &\quad + \frac{1}{2\Delta x}(((\theta v)(\alpha_k^+, t^n) + (\theta v)_x(\alpha_k^+, t^n)(x_{l+1} - \alpha_k) + \frac{1}{2}(\theta v)_{xx}(\alpha_k^+, t^n)(x_{l+1} - \alpha_k)^2 \\ &\quad - ((\theta v)(\alpha_k^-, t^n) + (\theta v)_x(\alpha_k^-, t^n)(x_{l-1} - \alpha_k) + \frac{1}{2}(\theta v)_{xx}(\alpha_k^-, t^n)(x_{l-1} - \alpha_k)^2))) - C_l^{(a)} \\ &= \frac{1}{2\Delta x}[\theta v]_{k,n} + \frac{x_{l+1} - \alpha_k}{2\Delta x}[(\theta v)_x]_{k,n} + \frac{(x_{l+1} - \alpha_k)^2}{4\Delta x}[(\theta v)_{xx}]_{k,n} - C_l^{(a)} + O((\Delta x)^2, (\Delta t)^2). \end{aligned}$$

So it will suffice for second order accuracy if the correction term for grid points J_i is

$$C_{J_i}^{(a)} = \frac{1}{2\Delta x}[\theta v]_{k,n} + \frac{x_{l+1} - \alpha_k}{2\Delta x}[(\theta v)_x]_{k,n} + \frac{(x_{l+1} - \alpha_k)^2}{4\Delta x}[(\theta v)_{xx}]_{k,n}. \quad (4.15)$$

We get $O((\Delta x)^2, (\Delta t)^2)$ for the local truncation error at the irregular grid point x_{J_i} .

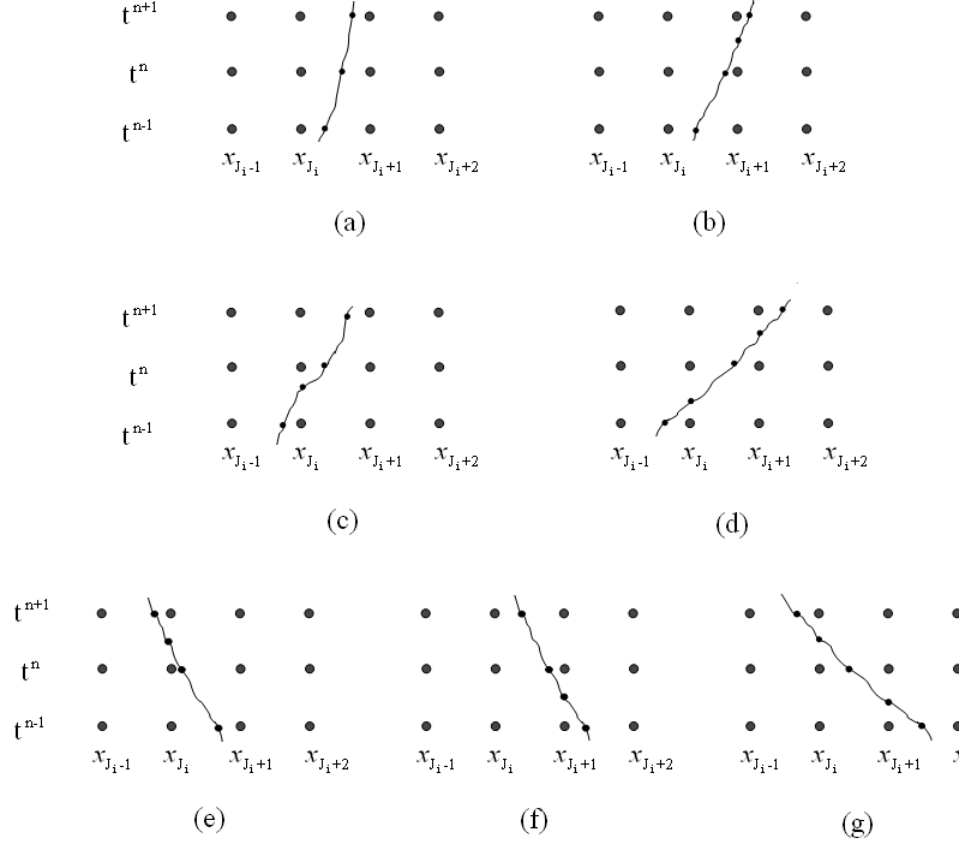


Figure 4.2: Diagram of the interface location for a three-level method. (a) no crossing, (b) crossing a grid line as $\alpha_i(t)$ increases with time at $t^n < \tau_2 < t^{n+1}$ only, (c) crossing a grid line as $\alpha_i(t)$ increases with time at $t^{n-1} < \tau_1 < t^n$ only, (d) crossing a grid line as $\alpha_i(t)$ increases with time at both $t^{n-1} < \tau_1 < t^n$ and $t^n < \tau_2 < t^{n+1}$, (e) crossing a grid line as $\alpha_i(t)$ decreases with time at $t^n < \tau_2 < t^{n+1}$ only, (f) crossing a grid line as $\alpha_i(t)$ decreases with time at $t^{n-1} < \tau_1 < t^n$ only, (g) crossing a grid line as $\alpha_i(t)$ decreases with time at both $t^{n-1} < \tau_1 < t^n$ and $t^n < \tau_2 < t^{n+1}$.

Likewise, we can find the correction term for the irregular grid point x_{J_i+1} by analyzing the local truncation error at that point if no grid line crossing happens:

$$C_{J_i+1}^{(a)} = \frac{1}{2\Delta x}[\theta v]_{k,n} + \frac{x_l - \alpha_k}{2\Delta x}[(\theta v)_x]_{k,n} + \frac{(x_l - \alpha_k)^2}{4\Delta x}[(\theta v)_{xx}]_{k,n}. \quad (4.16)$$

Dealing with grid crossing

If there is no grid crossing (see Case (a) in Fig. 4.2) as time evolves from t^{n-1} to t^{n+1} , the above work shows that the corrections to the leapfrog method come from the space discretization for the irregular grid points ($l = J_i, J_i + 1$) and

$$\frac{\theta(x_l, t^{n+1}) - \theta(x_l, t^{n-1})}{2\Delta t} = \theta_t(x_l, t^n) + O((\Delta t)^2)$$

However, if the interface crosses the grid line x_{J_i} when a shock travels to the left or crosses the grid line x_{J_i+1} when a shock travels to the right at some time τ as time evolves (see Cases (b)-(g) in Fig. 4.2), then the time discretization need to be corrected. Consider Case (b) first. Let $D_l^{(b)}$ denote the correction from the time discretization, where $l = J_i, J_i + 1$. The correction from the time discretization is needed for grid point $J_i + 1$ only, i.e., $D_{J_i}^{(b)} = 0$, but $D_{J_i+1}^{(b)} \neq 0$. Expanding $\theta(x_l, t^{n+1})$, where $l = J_i + 1$ for simplicity, in Taylor series at α_k and t^n gives

$$\begin{aligned} \theta(x_l, t^{n+1}) = & \theta(\alpha_k^-, t^n) + \theta_x(\alpha_k^-, t^n)(x_l - \alpha_k) + \theta_t(\alpha_k^-, t^n)\Delta t + \frac{1}{2}\theta_{xx}(\alpha_k^-, t^n)((x_l - \alpha_k))^2 \\ & + \frac{1}{2}\theta_{tt}(\alpha_k^-, t^n)(\Delta t)^2 + \theta_{xt}(\alpha_k^-, t^n)(x_l - \alpha_k)\Delta t + O((\Delta x)^3, (\Delta t)^3). \end{aligned}$$

Applying the interface relations gives

$$\begin{aligned} \theta(x_l, t^{n+1}) = & \theta(\alpha_k^+, t^n) + \theta_x(\alpha_k^+, t^n)(x_l - \alpha_k) + \theta_t(\alpha_k^+, t^n)\Delta t + \frac{1}{2}\theta_{xx}(\alpha_k^+, t^n)((x_l - \alpha_k))^2 \\ & + \frac{1}{2}\theta_{tt}(\alpha_k^+, t^n)(\Delta t)^2 + \theta_{xt}(\alpha_k^-, t^n)(x_l - \alpha_k)\Delta t + [*]_{k,n} + O((\Delta x)^3, (\Delta t)^3), \end{aligned}$$

where

$$\begin{aligned} [*]_{k,n} = & -[\theta]_{k,n} - [\theta_x]_{k,n}(x_l - \alpha) - [\theta_t]\Delta t - \frac{1}{2}[\theta_{xx}]_{k,n}(x_l - \alpha)^2 \\ & - \frac{1}{2}[\theta_{tt}]_{k,n}(\Delta t)^2 - [\theta_{xt}]_{k,n}(x_l - \alpha)\Delta t \\ = & -[\theta]_{k,n} - [\theta_x]_{k,n}(x_l - \alpha) + [(\theta v)_x]\Delta t - \frac{1}{2}[\theta_{xx}]_{k,n}(x_l - \alpha)^2 \\ & + \frac{1}{2}[(\theta v)_{xt}]_{k,n}(\Delta t)^2 + [(\theta v)_{xx}]_{k,n}(x_l - \alpha)\Delta t. \end{aligned}$$

Now expanding $\theta(\alpha_k^+, t^n)$, $\theta_x(\alpha_k^+, t^n)$, $\theta_t(\alpha_k^+, t^n)$, $\theta_{xx}(\alpha_k^+, t^n)$, $\theta_{tt}(\alpha_k^+, t^n)$, and $\theta_{xt}(\alpha_k^-, t^n)$ at x_l and t^n gives

$$\begin{aligned}\theta(x_l, t^{n+1}) &= \theta(x_l, t^n) + \theta_x(x_l, t^n)(\alpha_k - x_l) + \frac{1}{2}\theta_{xx}(x_l, t^n)(\alpha_k - x_l)^2 \\ &\quad + \theta_x(x_l, t^n)(x_l - \alpha_k) + \theta_{xx}(x_l, t^n)(\alpha_k - x_l)(x_l - \alpha_k) \\ &\quad + \theta_t(x_l, t^n)\Delta t + \theta_{tx}(x_l, t^n)(\alpha_k - x_l)\Delta t + \frac{1}{2}\theta_{xx}(x_l, t^n)((x_l - \alpha_k))^2 \\ &\quad + \frac{1}{2}\theta_{tt}(x_l, t^n)(\Delta t)^2 + \theta_{xt}(x_l, t^n)(x_l - \alpha_k)\Delta t + [*]_{k,n} + O((\Delta x)^3, (\Delta t)^3) \\ &= \theta(x_l, t^n) + \theta_t(x_l, t^n)\Delta t + \frac{1}{2}\theta_{tt}(x_l, t^n)(\Delta t)^2 + [*]_{k,n} + O((\Delta x)^3, (\Delta t)^3),\end{aligned}$$

Expanding $\theta(x_l, t^{n-1})$ in Taylor series at x_l and t^n gives

$$\theta(x_l, t^{n-1}) = \theta(x_l, t^n) - \theta_t(x_l, t^n)\Delta t + \frac{1}{2}\theta_{tt}(x_l, t^n)(\Delta t)^2$$

Thus it follows that

$$\frac{\theta(x_l, t^{n+1}) - \theta(x_l, t^{n-1})}{2\Delta t} = \theta_t(x_l, t^n) + \frac{[*]_{k,n}}{2\Delta t} + O((\Delta x)^2, (\Delta t)^2).$$

So the correction from the time discretization for Case (b) is

$$\begin{aligned}D_{J_i+1}^{(b)} &= \frac{[*]_{k,n}}{2\Delta t} \\ &= -\frac{[\theta]_{k,n}}{2\Delta t} - \frac{x_{J_i+1} - \alpha}{2\Delta t}[\theta_x]_{k,n} + \frac{[(\theta v)_x]}{2} - \frac{(x_{J_i+1} - \alpha)^2}{4\Delta t}[\theta_{xx}]_{k,n} \\ &\quad + \frac{\Delta t}{4}[(\theta v)_{xt}]_{k,n} + \frac{x_{J_i+1} - \alpha}{2}[(\theta v)_{xx}]_{k,n}.\end{aligned}$$

Hence, the correction for Case (b) is

$$\begin{aligned}C_{J_i}^{(b)} &= C_{J_i}^{(a)}, \\ C_{J_i+1}^{(b)} &= C_{J_i+1}^{(a)} + D_{J_i+1}^{(b)},\end{aligned}\tag{4.17}$$

Likewise, we can find the correction term for Cases (c) - (g) by analyzing the local truncation errors:

$$\begin{aligned}C_{J_i}^{(c)} &= C_{J_i}^{(a)} + D_{J_i}^{(c)}, \\ C_{J_i+1}^{(c)} &= C_{J_i+1}^{(a)},\end{aligned}\tag{4.18}$$

where

$$\begin{aligned}D_{J_i}^{(c)} &= -\frac{[\theta]_{k,n}}{2\Delta t} - \frac{x_{J_i} - \alpha}{2\Delta t}[\theta_x]_{k,n} - \frac{[(\theta v)_x]}{2} - \frac{(x_{J_i} - \alpha)^2}{4\Delta t}[\theta_{xx}]_{k,n} \\ &\quad + \frac{\Delta t}{4}[(\theta v)_{xt}]_{k,n} - \frac{x_{J_i} - \alpha}{2}[(\theta v)_{xx}]_{k,n}.\end{aligned}$$

$$\begin{aligned} C_{J_i}^{(d)} &= C_{J_i}^{(a)} + D_{J_i}^{(c)}, \\ C_{J_{i+1}}^{(d)} &= C_{J_{i+1}}^{(a)} + D_{J_{i+1}}^{(b)}, \end{aligned} \quad (4.19)$$

$$\begin{aligned} C_{J_i}^{(e)} &= C_{J_i}^{(a)} + D_{J_i}^{(e)}, \\ C_{J_{i+1}}^{(e)} &= C_{J_{i+1}}^{(a)}, \end{aligned} \quad (4.20)$$

where

$$\begin{aligned} D_{J_i}^{(e)} &= \frac{[\theta]_{k,n}}{2\Delta t} + \frac{x_{J_i} - \alpha}{2\Delta t} [\theta_x]_{k,n} - \frac{[(\theta v)_x]}{2} + \frac{(x_{J_i} - \alpha)^2}{4\Delta t} [\theta_{xx}]_{k,n} \\ &\quad - \frac{\Delta t}{4} [(\theta v)_{xt}]_{k,n} - \frac{x_{J_i} - \alpha}{2} [(\theta v)_{xx}]_{k,n}. \end{aligned}$$

$$\begin{aligned} C_{J_i}^{(f)} &= C_{J_i}^{(a)}, \\ C_{J_{i+1}}^{(f)} &= C_{J_{i+1}}^{(a)} + D_{J_{i+1}}^{(f)}, \end{aligned} \quad (4.21)$$

where

$$\begin{aligned} D_{J_{i+1}}^{(f)} &= \frac{[\theta]_{k,n}}{2\Delta t} + \frac{x_{J_{i+1}} - \alpha}{2\Delta t} [\theta_x]_{k,n} + \frac{[(\theta v)_x]}{2} + \frac{(x_{J_{i+1}} - \alpha)^2}{4\Delta t} [\theta_{xx}]_{k,n} \\ &\quad - \frac{\Delta t}{4} [(\theta v)_{xt}]_{k,n} + \frac{x_{J_{i+1}} - \alpha}{2} [(\theta v)_{xx}]_{k,n}. \\ C_{J_i}^{(g)} &= C_{J_i}^{(a)} + D_{J_i}^{(e)}, \\ C_{J_{i+1}}^{(g)} &= C_{J_{i+1}}^{(a)} + D_{J_{i+1}}^{(f)}, \end{aligned} \quad (4.22)$$

Note that the PDE system does not supply the information about v_t . So we can not deduce $[(\theta v)_{xt}]_{k,n}$ from the PDE system. And $[(\theta v)_{xt}]_{k,n}$ is hard to approximate correctly by numerical methods due to the explicit methods used in our study. So we drop this term in our correction to make the local truncation errors at those irregular grid points near the interfaces to be $O((\Delta x)^2, \Delta t)$. Away from the interfaces, the local truncation errors for the finite difference methods (the leapfrog method and MacCormack's method) are $O((\Delta x)^2, (\Delta t)^2)$. The correction terms will assure the local truncation errors for the grid points near the interfaces are $O((\Delta x)^2, \Delta t)$. The global accuracy of the finite difference method is second order, based on the fact that the local truncation errors of a finite difference method on a boundary can be one order lower than those of interior points without affecting global second order accuracy [38, 48]. But the above claim may not be true for some problems with moving interface.

4.2 Numerical Example

In our computations, we choose $\psi = 1.8$, $\sigma = 1.0$, and initially $\theta_u = 0.5$, $\theta_u = 0$, $\alpha_1 = 1/3$ and $\alpha_2 = 2/3$. Since we do not know the exact solution of the system in (4.1), we compare the computed results against the one computed from the finest grid which is $N = 1280$.

4.2.1 The IIM and the Leapfrog Method

Fig. 4.3 and 4.4 show the solutions of θ and v solved by the immersed interface method based on the leapfrog method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$. The computed solutions exhibit high resolution near the interfaces.

Fig. 4.5 shows the computed solutions ($T = 0.1$) solved by the immersed interface method based on the leapfrog method. Those on the bottom row have high resolution near the interface whereas, on the top row, the computed solutions solved by the leapfrog method without enforcing the jump conditions at the irregular grid points have poor resolution near the interface (oscillations at the discontinuities were observed in θ). Fig. 4.6 shows the computed solutions solved by the immersed interface method based on the leapfrog method for $T = 5$.

The grid refinement analysis has been completed to check the convergence rate of our approach. We did not obtain second order convergence as we expected from the local truncation error analysis. As discussed earlier in this chapter, we have second order local truncation error at regular grid points and first order truncation error at a few irregular grid points. We expected the one order lower truncation error at the irregular grid points would not affect the global error. However, no good results have been obtained although the numerical error with the immersed interface method has much smaller magnitude compared to the method without enforcing the jump conditions to the stencils with interfaces. See Tables 4.1 - 4.3.

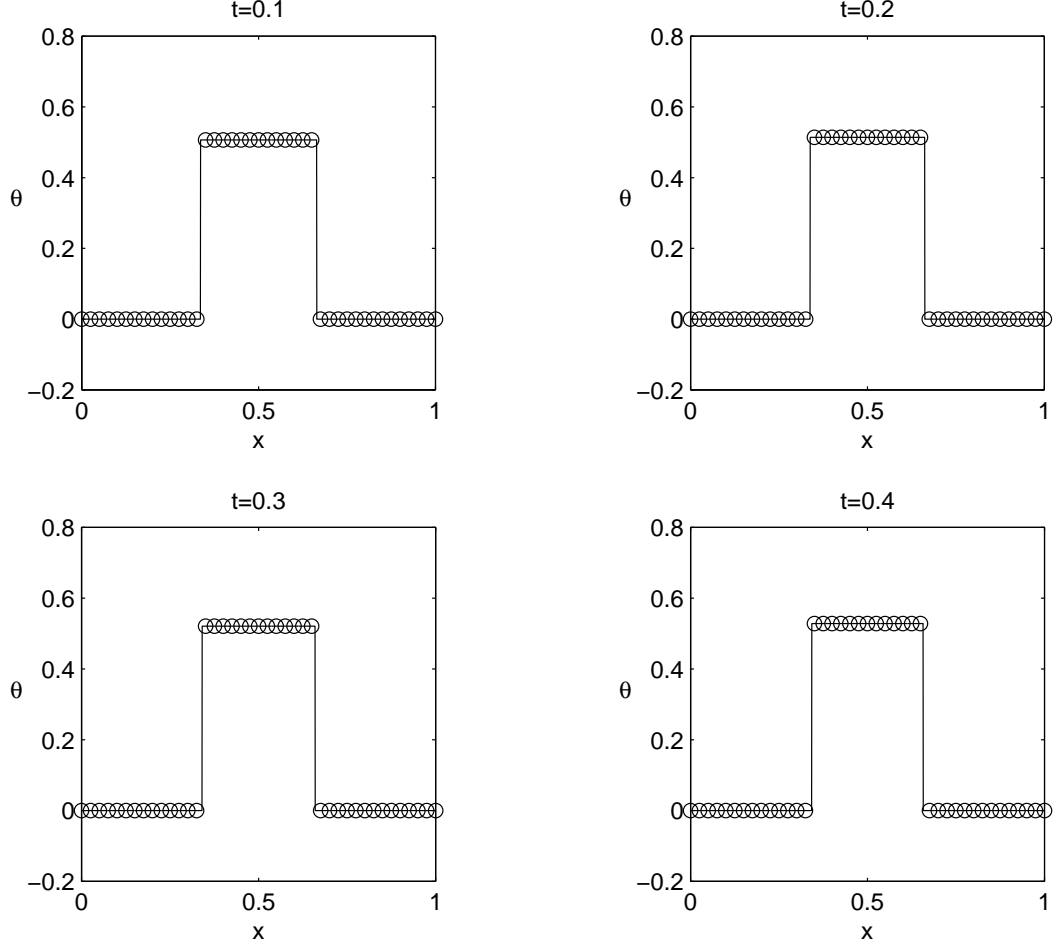


Figure 4.3: Comparison of solution of θ solved by the IIM based on the leapfrog method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$.

Table 4.1: Accuracy of θ solved by the IIM based on the leapfrog method for the system in (4.1) with constant physical parameters.

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio	L_2 error	L_2 ratio
40	8.74e-4	—	2.84e-4	—	4.98e-4	—
80	1.33e-8	6.57e4	4.49e-9	6.33e4	7.72e-9	6.45e4
160	5.42e-9	2.45	1.80e-9	2.50	3.12e-9	2.48
320	1.09e-4	4.97e-5	3.58e-5	5.01e-5	6.25e-5	4.99
640	5.46e-5	2.00	1.80e-5	1.99	3.13e-5	2.00

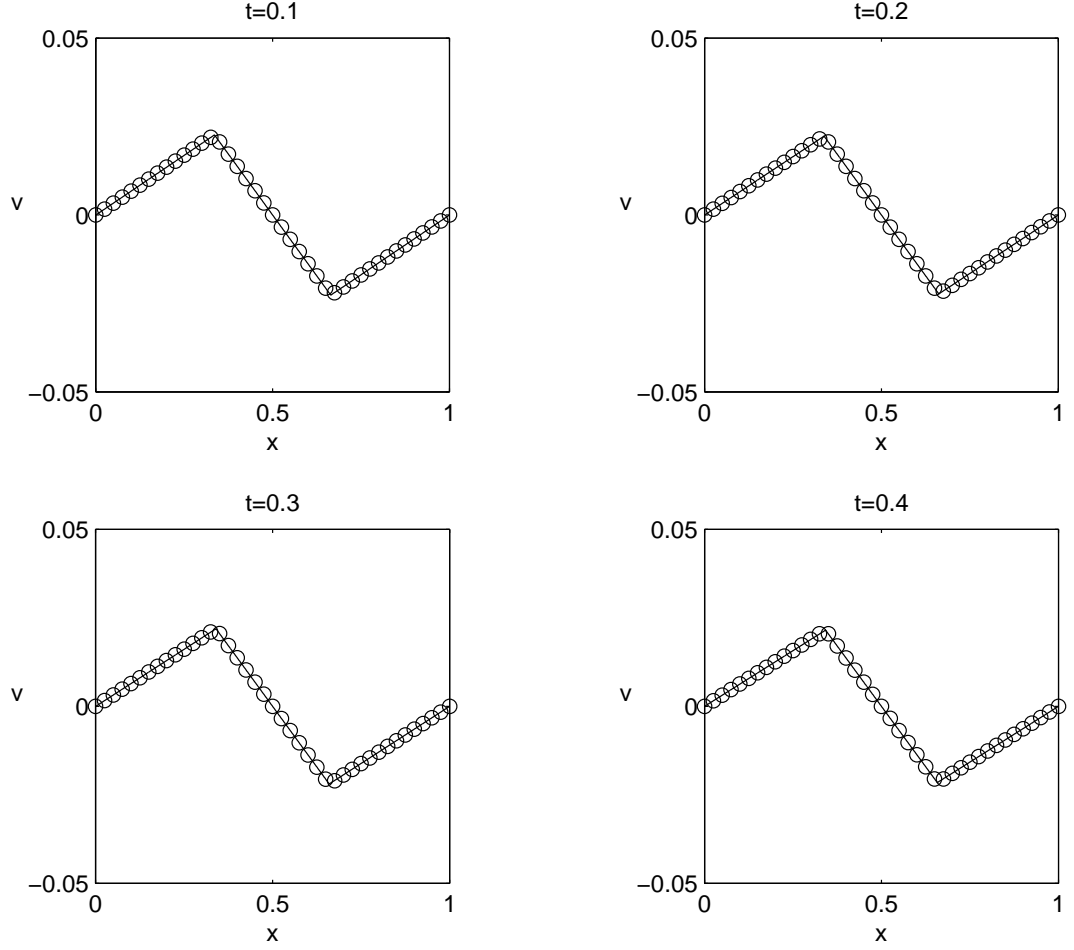


Figure 4.4: Comparison of solution of v solved by the IIM based on the leapfrog method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$.

Table 4.2: Accuracy of v solved by the IIM based on the leapfrog method for the system in (4.1) with constant physical parameters.

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio	L_2 error	L_2 ratio
40	5.95e-5	—	2.13e-5	—	2.93e-5	—
80	1.26e-7	4.71e2	5.36e-8	3.97e2	6.43e-8	4.56e2
160	6.01e-8	2.10	2.52e-8	2.13	3.03e-8	2.12
320	7.63e-6	7.87e-3	2.63e-6	9.58e-3	3.63e-6	8.35e-3
640	3.81e-6	2.00	1.31e-6	2.01	1.81e-6	2.01

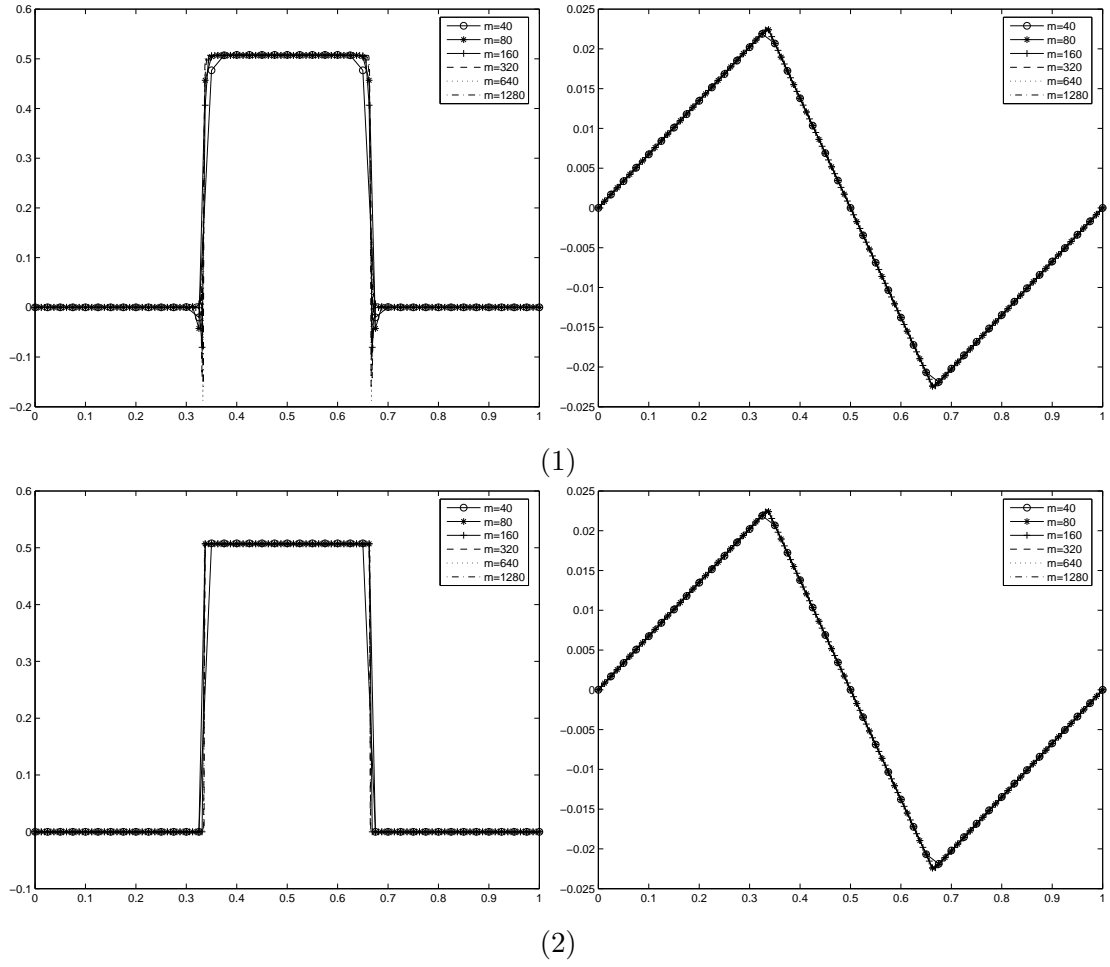


Figure 4.5: Computed solutions of (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ for $T = 0.1$. (1) – the solutions of θ (left) and v (right) solved via the leapfrog method; (2) – the solutions of θ (left) and v (right) solved via the IIM based on leapfrog method. Fine meshes, $N = 640$.

Table 4.3: Accuracy of θ solved by the leapfrog method for the hyperbolic equation in (4.1) with constant physical parameters.

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio	L_2 error	L_2 ratio
40	3.00e-2	–	2.75e-3	–	8.11e-3	–
80	4.39e-2	0.685	2.27e-3	1.21	9.68e-3	0.838
160	9.39e-2	0.467	1.64e-3	1.38	1.08e-2	0.895
320	2.66e-1	0.353	2.39e-3	0.685	2.16e-2	0.501
640	1.88e-1	1.42	1.57e-3	1.53	1.41e-2	1.53

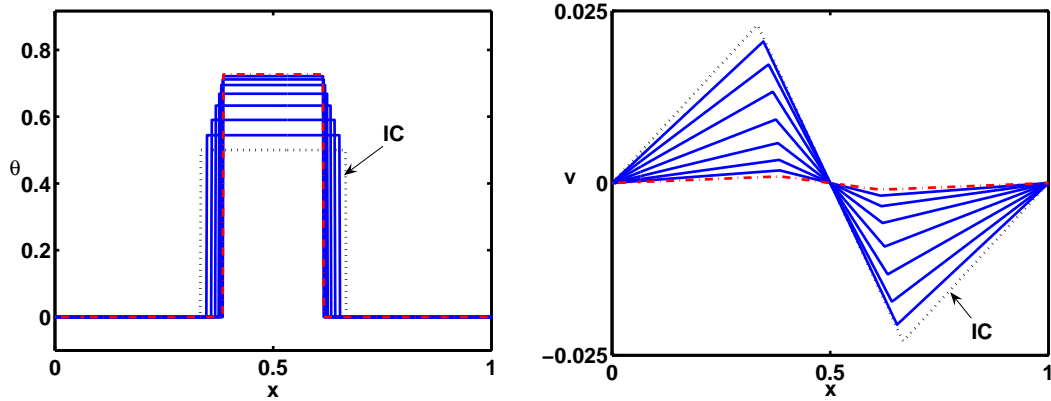


Figure 4.6: Computed solutions of (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ for $T = 5$ solved by the IIM based on the leapfrog method (the solutions of θ : left plot, and v : right plot). The dotted lines represent the initial data, dash-dotted lines represent the solution at the final time ($T = 5$), and solid lines represent the intermediate solution between initial and final time. Fine meshes, $N = 640$.

4.2.2 The IIM and MacCormack's Method

As we discussed early in this chapter, MacCormack's method doubles the work of computing v in order to make the algorithm to be of second order accuracy for our problem. To save the computing time, we used the formula for the exact solution of v we derived in the Chapter 2 to compute the velocity and determine the jump conditions. We used this information to solve the hyperbolic equation using the immersed interface method based on MacCormack's method.

Figs. 4.7 and 4.8 show the solutions of θ and v solved by the immersed interface method based on MacCormack's method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$. The computed solutions show good resolution near the interfaces. However, oscillations are observed near the interface when the figure is enlarged (see Fig. 4.9). The oscillation develops quickly with time. This may be due to the nature of MacCormack's method: taking the simple averaging process of MacCormack's method after the two-step predictor-corrector procedure. It will introduce big errors whenever the interface crosses grid lines. This could be approved by the grid refinement analysis

informally. See the Table 4.4. Note that finer mesh has more opportunities to experience the grid line crossing. Hence the simple averaging process will bring in more errors for the grid points near the interfaces.

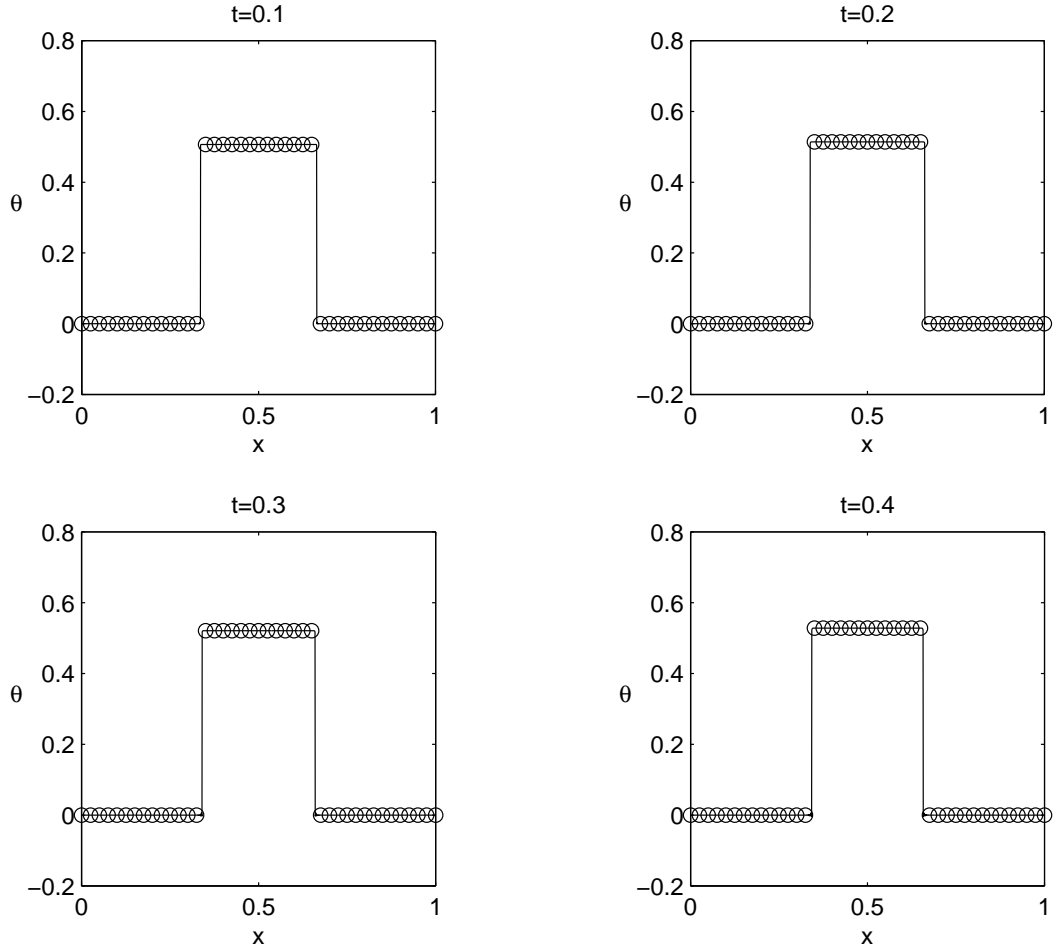


Figure 4.7: Comparison of solution of θ solved by the IIM based on MacCormack's method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$.

Fig. 4.10 shows solutions computed by regular MacCormack's method (left) and the immersed interface method based on MacCormack's method (right) for $T = 5$. Unfortunately, non-physical oscillations are observed in both algorithms. However, the immersed interface method obviously damps down the oscillation more than regular MacCormack's

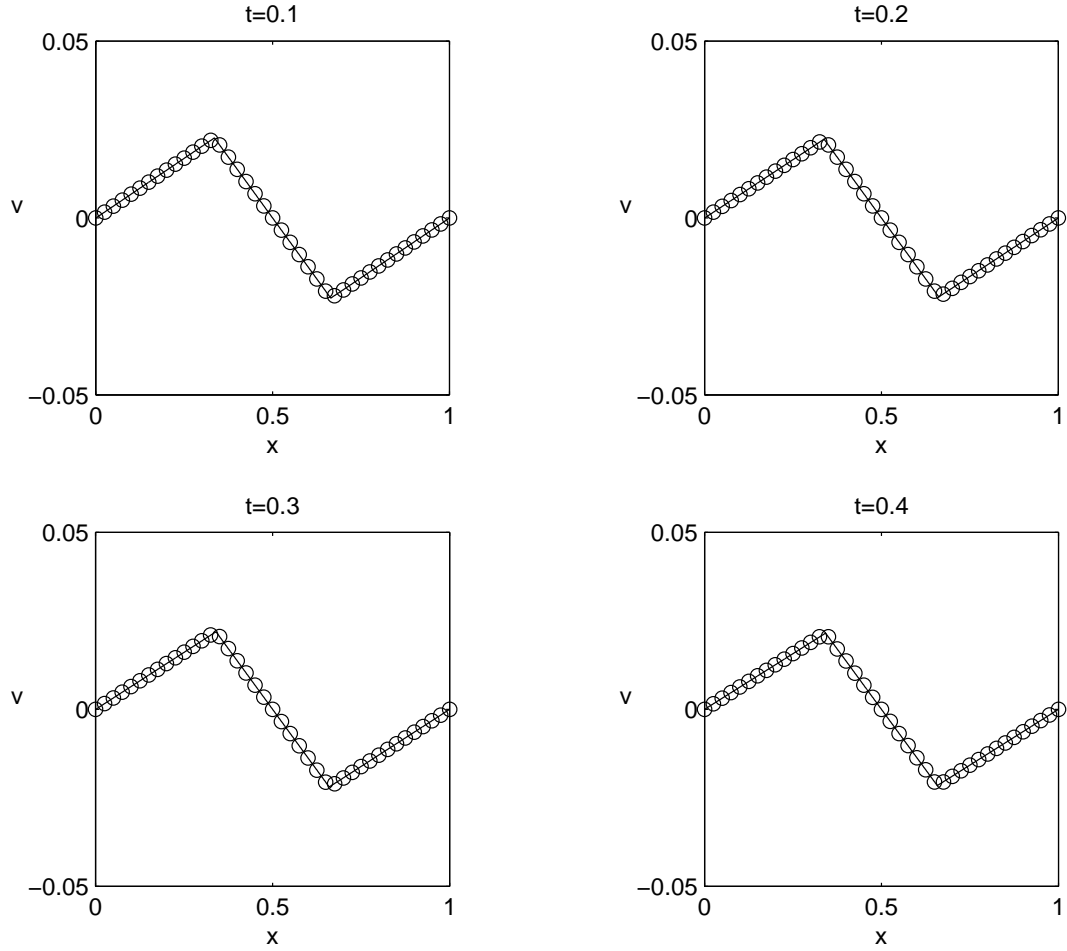


Figure 4.8: Comparison of solution of v solved by the IIM based on MacCormack's method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$.

Table 4.4: Accuracy of θ solved by the IIM based on MacCormack's method for the hyperbolic equation in (4.1) with constant physical parameters.

N	L_∞ error	L_∞ ratio	L_1 error	L_1 ratio	L_2 error	L_2 ratio
40	8.74e-4	—	2.84e-4	—	4.98e-4	—
80	2.65e-4	3.30	6.64e-6	42.8	4.18e-5	11.9
160	4.19e-4	0.633	8.53e-6	0.778	5.53e-5	0.756
320	3.10e-3	0.135	6.86e-5	0.124	2.76e-4	0.200
640	2.59e-3	1.20	4.16e-5	1.65	2.25e-4	1.23

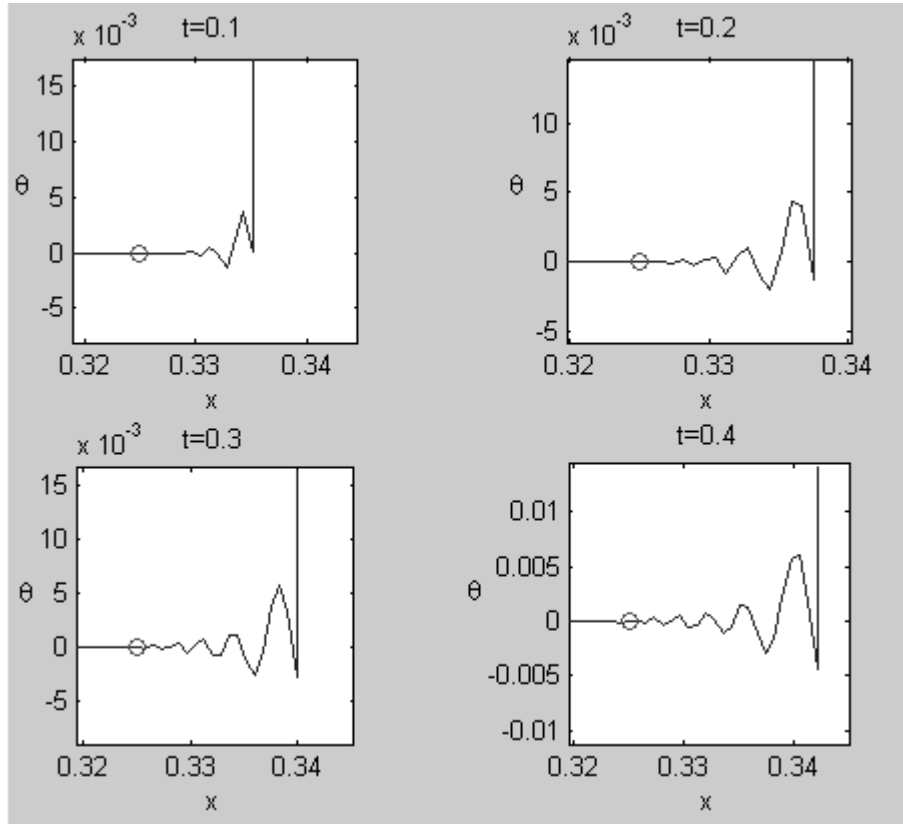


Figure 4.9: Enlargement of simulations from Fig. 4.7. Partial computed solution of θ near the interface solved by the IIM based on MacCormack's method for the system in (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ at $t = 0.1$ (top left), $t = 0.2$ (top right), $t = 0.3$ (bottom left), and $t = 0.4$ (bottom right). The solid line represents the solution computed with mesh size $h = 1/1280$ and the symbol 'o' represents the numerical solutions computed with the mesh size $h = 1/40$. Oscillations are observed in the enlarged plots.

method.

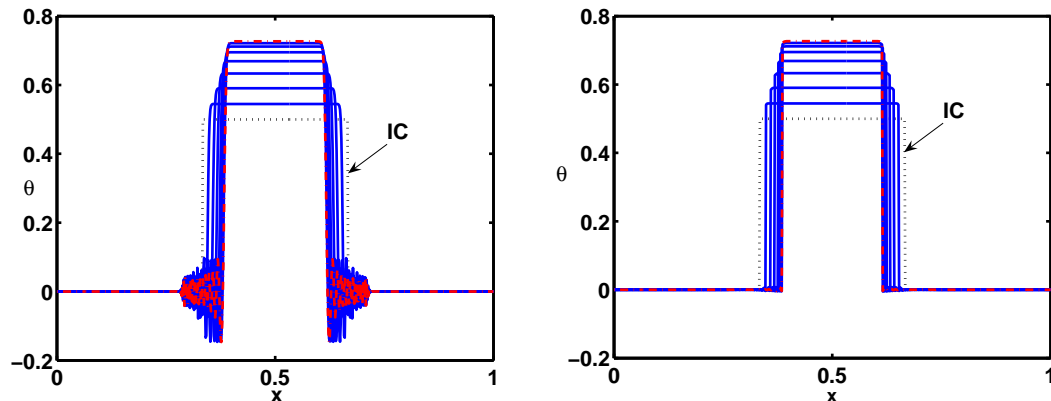


Figure 4.10: Computed solutions θ of (4.1) with $M = 0.5$, $\varphi = 0$, $\psi = 1.8$, $\sigma = 1$, and $\theta_l = 0$, $\theta_u = 0.5$ for $T = 5$ solve by two algorithms. Left: using MacCormack's method; right: using the IIM based on MacCormack's method. The dotted lines represent the initial data, dash-dotted lines represent the solution at the final time ($T = 5$), and solid lines represent the intermediate solution between initial and final time. Fine meshes, $N = 640$. Oscillations are observed.

4.3 Conclusions

In this chapter, the immersed interface method has been applied to a one dimensional fluid mixture model of tissue deformations with all constant physical parameters based on the three-level leapfrog method and two-step MacCormack's method. Deriving the jump conditions for both hyperbolic and elliptic equations is necessary and important. We have found out the correction terms for different cases corresponding to the different conventional finite difference methods used for the regular grid points away from the interfaces. The solutions of high resolution near the interface have been obtained. However, we did not get solutions of second order accuracy as we expected from the local truncation error analysis. And when MacCormack's method is used, oscillation is observed near the interface, and the oscillation developed with time. The immersed interface method mostly damps down the oscillation.

Chapter 5

A 2D Fluid Mixture Model of Tissue Deformations

5.1 Introduction

In this chapter, we consider a three dimensional mathematical model developed in [2], [3] for modelling tissue deformations in two dimensions. The tissues are considered to be composed of two inter-penetrating material phases: an aqueous phase and a cell-fiber phase. The aqueous phase is composed of all the water and dissolved extracellular components of the tissues. The cell-fiber phase consists of the cells and the remaining, generally fibrous, extracellular components. It is assumed that: (1) the two phases occupy complementary portions of the space, (2) the aqueous phase behaves as a Stokes fluid, (3) the stresses in the cell-fiber phase are dissipated by permanent deformation on the relevant time scale and can also be treated as a Stokes flow. These assumptions lead to the following three dimensional system of partial differential equations:

$$\frac{\partial \theta}{\partial t} + \nabla \cdot (\theta \underline{V}) = 0, \quad (5.1)$$

$$\nabla \cdot (\theta \underline{V} + (1 - \theta) \underline{W}) = 0, \quad (5.2)$$

$$\nabla \cdot (\theta(\lambda(\nabla \cdot \underline{V})\underline{I} + 2\mu\underline{E})) - \theta \nabla p - \varphi(\theta)(\underline{V} - \underline{W}) = \nabla(\theta(\Upsilon - \psi)), \quad (5.3)$$

$$-(1 - \theta) \nabla p - \varphi \frac{\theta}{1 - \theta} (\underline{V} - \underline{W}) = \nabla((1 - \theta)\Upsilon). \quad (5.4)$$

where θ is volume fraction of cells and fibers, \underline{V} is velocity of cell-fiber phase, \underline{W} is velocity of aqueous phase, p is interphase pressure, φ is specific drag coefficient, ψ is contractility

coefficient, Υ is the solvation stress. μ is specific shear modulus of viscosity of the cell-fiber fraction, λ is specific bulk modulus of viscosity of cell-fiber phase, $\underline{\underline{I}}$ is the identity tensor, and $\underline{\underline{E}} = \frac{1}{2}(\nabla \underline{V} + (\nabla \underline{V})^T)$ is the strain rate tensors for the cell-fiber phase.

Note that $0 \leq \theta \leq 1$ and parameters φ , μ , λ , ψ , Υ and h are nonnegative. Also note that these parameters may vary in space and change with the time, and they may explicitly depend on θ .

Note also that either phase alone, $\theta = 0$ or 1 , would be incompressible. But when the phases are mixed, only the mixture is incompressible; the individual phases in the mixture are compressible.

The boundary conditions (BC) are given as follows

$$\underline{V}|_{\partial\Omega} = \underline{W}|_{\partial\Omega} = 0 \quad (5.5)$$

$$\nabla\theta \cdot \hat{n}|_{\partial\Omega} = 0 \quad (5.6)$$

$$\nabla p \cdot \hat{n}|_{\partial\Omega} = 0 \quad (5.7)$$

Three types of initial condition (IC) are considered in this study. One is with continuous data as follows

$$\theta(x, y, 0) = \theta_0(1 + \varepsilon \cos(\pi x) \cos(\pi y)), \quad (5.8)$$

where we choose $\varepsilon = 0.01$ for the computations. The second type of initial condition is with random perturbation as follows

$$\theta(x, y, 0) = \theta_0(1 + \sum_{j=1}^{j=n_{qf}} \varepsilon_j \cos(\lambda_{1j}\pi x) \cos(\lambda_{2j}\pi y)), \quad (5.9)$$

where $\lambda_{ij} = \text{integer of } (n_{amp} * \zeta_{ij})$, n_{qf} and n_{amp} are integers, ε_j and ζ_{ij} are the random numbers generated from a uniform $(0, 1)$ distribution.

The third type of initial condition is with random perturbation at each grid point as follows

$$\theta(x_i, y_j, 0) = \theta_0(1 + \prod_{k=1}^{k=n_{qf}} (\zeta_k - 0.5)), \quad (5.10)$$

where n_{qf} is integer, and ζ_k is the random number generated from a uniform $(0, 1)$ distribution.

A reasonable range of dimensional and non-dimensional parameters is presented in Table 5.1.

Table 5.1: Expected ranges of parameter values [2, 3].

parameter	symbol	units	range
specific drag coefficient	φ	$kg/m^3\text{-sec}$	$10^{12} - 10^{14}$
tissue viscosity	μ, λ	$kg/m\text{-sec}$	10^5
specific contractility coefficient	ψ	$kg/m\text{-sec}^2$	$10^3 - 10^4$
thickness of gel	h	cm	~ 0.3
volume fraction of cell-fiber phase	θ_0	-	0.1 - 0.3
gel size	x_f, y_f	cm	5

The mathematical model is a non-linear, mixed (hyperbolic and elliptic) system of differential equations. Shocks will be developed in the time evolution process. The purpose of this study is to investigate the stability of the equilibria, and to show the utility of our numerical methods in simulating this stiff pattern-formation problem.

5.1.1 Deriving Body Force

In this chapter, we will focus on simulating pattern formation by contractile force in a gel (two dimensions). Starting with the three dimensional system, we need to find the proper body force by reducing the system to two dimensions with a lubrication approximation assuming the third dimension is shallow [87], [88], [89]. Solving (5.4) for \underline{W} , one gets the following expression

$$\underline{W} = \underline{V} - \frac{(1 - \theta)\nabla p + \nabla((1 - \theta)\Upsilon)}{\varphi(\theta)}.$$

Substituting the above expression into (5.2),

$$\nabla \cdot \left(\underline{V} - \frac{(1 - \theta)^2 \nabla p + (1 - \theta) \nabla(1 - \theta)\Upsilon}{\varphi(\theta)} \right) = 0. \quad (5.11)$$

Combining (5.3) and (5.4), one can obtain

$$\nabla \cdot (\theta(\lambda(\nabla \cdot \underline{V})\underline{I} + 2\mu\underline{E})) - \nabla p = \nabla(\Upsilon - \theta\psi). \quad (5.12)$$

Let the thickness of the gel be $h(x, y)$, and $\underline{V} = (v_1, v_2, v_3)$. One can scale z, v_3 to

$$z = \delta z', \quad v_3 = \delta v'_3 \quad \text{with} \quad \delta \ll 1.$$

The following boundary conditions were used in derivation

$$(v_1, v_2, v_3)(z' = 0) = (0, 0, 0), \quad \text{no motion,}$$

$$\frac{\partial v_1}{\partial z}(z' = h) = \frac{\partial v_2}{\partial z}(z' = h) = 0, \quad \text{no shear stress,}$$

$$\frac{\partial v_3}{\partial x}(z' = h) = \frac{\partial v_3}{\partial y}(z' = h) = 0, \quad \text{no } \perp \text{ stress.}$$

Note that $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right) = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{1}{\delta} \frac{\partial}{\partial z'} \right)$. We apply the scaling to the simplified system. The steady form of (5.1) becomes

$$\frac{\partial}{\partial x}(\theta v_1) + \frac{\partial}{\partial y}(\theta v_2) + \frac{\partial}{\partial z'}(\theta v'_3) = 0. \quad (5.13)$$

Equation (5.11) becomes

$$\begin{aligned} & \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v'_3}{\partial z'} - \frac{\partial}{\partial x} \left\{ \frac{1}{\varphi(\theta)} \left[(1-\theta)^2 \frac{\partial p}{\partial x} + (1-\theta) \frac{\partial}{\partial x}((1-\theta)\Upsilon) \right] \right\} \\ & - \frac{\partial}{\partial y} \left\{ \frac{1}{\varphi(\theta)} \left[(1-\theta)^2 \frac{\partial p}{\partial y} + (1-\theta) \frac{\partial}{\partial y}((1-\theta)\Upsilon) \right] \right\} \\ & - \frac{1}{\delta^2} \frac{\partial}{\partial z'} \left\{ \frac{1}{\varphi(\theta)} \left[(1-\theta)^2 \frac{\partial p}{\partial z'} + (1-\theta) \frac{\partial}{\partial z'}((1-\theta)\Upsilon) \right] \right\} = 0. \end{aligned} \quad (5.14)$$

Equation (5.12) becomes

$$\begin{aligned} & \nabla \cdot \left\{ \theta \left[\lambda \left(\frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v'_3}{\partial z'} \right) \underline{I} + 2\mu \begin{pmatrix} \frac{\partial v_1}{\partial x} & \frac{(\partial v_1 + \partial v_2)}{\partial y} & \frac{\frac{1}{\delta} \frac{\partial v_1}{\partial z'} + \delta \frac{\partial v'_3}{\partial x}}{2} \\ \frac{\partial v_1 + \partial v_2}{\partial y} & \frac{\partial v_2}{\partial y} & \frac{\frac{1}{\delta} \frac{\partial v_2}{\partial z'} + \delta \frac{\partial v'_3}{\partial y}}{2} \\ \frac{\frac{1}{\delta} \frac{\partial v_1}{\partial z'} + \delta \frac{\partial v'_3}{\partial x}}{2} & \frac{\frac{1}{\delta} \frac{\partial v_2}{\partial z'} + \delta \frac{\partial v'_3}{\partial y}}{2} & \frac{\partial v'_3}{\partial z'} \end{pmatrix} \right] \right\} \\ & - \begin{pmatrix} \frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial y} \\ \frac{1}{\delta} \frac{\partial p}{\partial z'} \end{pmatrix} = \begin{pmatrix} \frac{\partial(\Upsilon - \theta\psi)}{\partial x} \\ \frac{\partial(\Upsilon - \theta\psi)}{\partial y} \\ \frac{1}{\delta} \frac{\partial(\Upsilon - \theta\psi)}{\partial z'} \end{pmatrix}. \end{aligned} \quad (5.15)$$

Expanding (5.15), one obtains

$$\begin{aligned} & \frac{\partial}{\partial x} \left\{ \theta \lambda \left(\frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v'_3}{\partial z'} \right) \right\} + 2 \frac{\partial}{\partial x} \left\{ \theta \mu \frac{\partial v_1}{\partial x} \right\} + \frac{\partial}{\partial y} \left\{ \theta \mu \left(\frac{\partial v_2}{\partial x} + \frac{\partial v_1}{\partial y} \right) \right\} \\ & + \frac{1}{\delta} \frac{\partial}{\partial z'} \left\{ \theta \mu \left(\frac{1}{\delta} \frac{\partial v_1}{\partial z'} + \delta \frac{\partial v'_3}{\partial x} \right) \right\} - \frac{\partial p}{\partial x} = \frac{\partial}{\partial x}(\Upsilon - \theta\psi), \end{aligned} \quad (5.16)$$

$$\begin{aligned} & \frac{\partial}{\partial y} \left\{ \theta \lambda \left(\frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v'_3}{\partial z'} \right) \right\} + \frac{\partial}{\partial x} \left\{ \theta \mu \left(\frac{\partial v_2}{\partial x} + \frac{\partial v_1}{\partial y} \right) \right\} + 2 \frac{\partial}{\partial y} \left\{ \theta \mu \frac{\partial v_2}{\partial y} \right\} \\ & + \frac{1}{\delta} \frac{\partial}{\partial z'} \left\{ \theta \mu \left(\frac{1}{\delta} \frac{\partial v_2}{\partial z'} + \delta \frac{\partial v'_3}{\partial y} \right) \right\} - \frac{\partial p}{\partial y} = \frac{\partial}{\partial y}(\Upsilon - \theta\psi), \end{aligned} \quad (5.17)$$

and

$$\begin{aligned} \frac{1}{\delta} \frac{\partial}{\partial z'} \left\{ \theta \lambda \left(\frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v'_3}{\partial z'} \right) \right\} + \frac{\partial}{\partial x} \left\{ \theta \mu \left(\delta \frac{\partial v'_3}{\partial x} + \frac{1}{\delta} \frac{\partial v_1}{\partial z'} \right) \right\} + \frac{2}{\delta} \frac{\partial}{\partial z'} \left\{ \theta \mu \frac{\partial v'_3}{\partial z'} \right\} \\ \frac{\partial}{\partial y} \left\{ \theta \mu \left(\frac{1}{\delta} \frac{\partial v_2}{\partial z'} + \delta \frac{\partial v'_3}{\partial y} \right) \right\} - \frac{1}{\delta} \frac{\partial p}{\partial z'} = \frac{1}{\delta} \frac{\partial}{\partial z'} (\Upsilon - \theta \psi). \end{aligned} \quad (5.18)$$

Equation (5.18) can be simplified further by multiplying δ

$$\begin{aligned} \frac{\partial}{\partial z'} \left\{ \theta \lambda \left(\frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v'_3}{\partial z'} \right) \right\} + \frac{\partial}{\partial x} \left\{ \theta \mu \left(\delta^2 \frac{\partial v'_3}{\partial x} + \frac{\partial v_1}{\partial z'} \right) \right\} + 2 \frac{\partial}{\partial z'} \left\{ \theta \mu \frac{\partial v'_3}{\partial z'} \right\} \\ \frac{\partial}{\partial y} \left\{ \theta \mu \left(\frac{\partial v_2}{\partial z'} + \delta^2 \frac{\partial v'_3}{\partial y} \right) \right\} - \frac{\partial p}{\partial z'} = \frac{\partial}{\partial z'} (\Upsilon - \theta \psi). \end{aligned} \quad (5.19)$$

The only way that (5.16), (5.17), and (5.19) will not reduce to a triviality as $\delta \rightarrow 0$ is if p is rescaled with $\frac{1}{\delta^2}$. Thus we write $p = \frac{p'}{\delta^2}$ and, to lowest order, the equations are

$$\frac{\partial}{\partial x} (\theta v_1) + \frac{\partial}{\partial y} (\theta v_2) + \frac{\partial}{\partial z'} (\theta v'_3) = 0, \quad (5.20)$$

$$\frac{\partial}{\partial z'} \left\{ \frac{1}{\varphi(\theta)} (1 - \theta)^2 \frac{\partial p'}{\partial z'} \right\} = 0, \quad (5.21)$$

$$\frac{\partial}{\partial z'} \left\{ \theta \mu \frac{\partial v_1}{\partial z'} \right\} - \frac{\partial p'}{\partial x} = 0, \quad (5.22)$$

$$\frac{\partial}{\partial z'} \left\{ \theta \mu \frac{\partial v_2}{\partial z'} \right\} - \frac{\partial p'}{\partial y} = 0, \quad (5.23)$$

$$\frac{\partial p'}{\partial z'} = 0. \quad (5.24)$$

Note that (5.21) is always true due to (5.24). From (5.24), we know that

$$p' = p'(x, y). \quad (5.25)$$

By integrating (5.22) over z' from 0 to $h(x, y)$, we obtain

$$\theta \mu \frac{\partial v_1}{\partial z'} = \frac{\partial p'}{\partial x} z' + c_1.$$

Applying the boundary condition that $\frac{\partial v_1}{\partial z'}(z = h) = 0$, we can obtain $c_1 = -\frac{\partial p'}{\partial x} h$. So

$$\frac{\partial v_1}{\partial z'} = \frac{1}{\theta \mu} \frac{\partial p'}{\partial x} (z' - h).$$

Assume μ is a constant, and $\theta = \theta(x, y, t)$. Integrating the above expression over z' from 0 to $h(x, y)$ again, we obtain

$$v_1 = \int_0^h \frac{1}{\theta \mu} \frac{\partial p'}{\partial x} (z' - h) dz' = \frac{1}{\theta \mu} \frac{\partial p'}{\partial x} \left[\frac{(z')^2}{2} - h z' \right]_0^h.$$

Hence,

$$v_1 = \frac{1}{\theta\mu} \frac{\partial p'}{\partial x} \left[-\frac{(h)^2}{2} \right]. \quad (5.26)$$

Likewise, we can obtain

$$v_2 = \frac{1}{\theta\mu} \frac{\partial p'}{\partial y} \left[-\frac{(h)^2}{2} \right]. \quad (5.27)$$

Rewriting (5.26) and (5.27), we get

$$-\begin{pmatrix} \frac{\partial p'}{\partial x} \\ \frac{\partial p'}{\partial y} \end{pmatrix} = \frac{2\theta\mu}{h^2} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}. \quad (5.28)$$

Hence, we find the proper body force term, i.e., $\frac{2\mu}{h^2}\theta\underline{V}$.

Taking the body force into account and ignoring solvation stress, one obtains the two dimensional version model of tissue deformations as follows:

$$\frac{\partial \theta}{\partial t} + \nabla \cdot (\theta \underline{V}) = 0, \quad x_0 < x < x_f, \quad y_0 < y < y_f \quad (5.29)$$

$$\nabla \cdot (\theta \underline{V} + (1 - \theta) \underline{W}) = 0, \quad (5.30)$$

$$\nabla \cdot (\theta(\lambda(\nabla \cdot \underline{V})\underline{I} + 2\mu\underline{E})) - \theta \nabla p - \varphi(\theta)(\underline{V} - \underline{W}) = \nabla(-\theta\psi) + \frac{2\mu}{h^2}\theta\underline{V}, \quad (5.31)$$

$$(1 - \theta)\nabla p - \varphi \frac{\theta}{1 - \theta}(\underline{V} + \underline{W}) = 0. \quad (5.32)$$

5.1.2 Simplification

Solving (5.32), one can obtain

$$\underline{W} = \underline{V} - \frac{(1 - \theta)^2}{\varphi\theta} \nabla p.$$

\underline{W} can be eliminated entirely by plugging the expression of \underline{W} into (5.30)

$$\nabla \cdot \left(\underline{V} - \frac{(1 - \theta)^3}{\varphi\theta} \nabla p \right) = 0. \quad (5.33)$$

Note that (5.31) and (5.32) can be combined to give

$$\nabla \cdot (\theta(\lambda(\nabla \cdot \underline{V})\underline{I} + 2\mu\underline{E})) - \nabla p = -\nabla(\theta\psi) + \frac{2\mu}{h^2}\theta\underline{V}. \quad (5.34)$$

So the two dimensional version system can be summarized as

$$\frac{\partial \theta}{\partial t} + \nabla \cdot (\theta \underline{V}) = 0, \quad x_0 < x < x_f, \quad y_0 < y < y_f, \quad (5.35)$$

$$\nabla \cdot \left(\underline{V} - \frac{(1 - \theta)^3}{\varphi\theta} \nabla p \right) = 0, \quad (5.36)$$

$$\nabla \cdot (\theta(\lambda(\nabla \cdot \underline{V})\underline{I} + 2\mu\underline{E})) - \nabla p = -\nabla(\theta\psi) + \frac{2\mu}{h^2}\theta\underline{V}. \quad (5.37)$$

5.1.3 Nondimensionalization

We nondimensionalize the model (5.35) - (5.37) by letting $t = \tilde{t}t_0$, $\underline{X} = \tilde{X}X_0$, $\underline{V} = \tilde{V}V_0 = \tilde{V}\frac{X_0}{t_0}$, $p = \tilde{p}p_0$, $\psi = \tilde{\psi}p_0$, where \tilde{t} , \tilde{X} , \tilde{V} , \tilde{p} , and $\tilde{\psi}$ are nondimensional variables, and t_0 , X_0 , V_0 , and p_0 are dimensional constants. Note that $\nabla = \frac{1}{X_0}\tilde{\nabla}$, $\frac{\partial\theta}{\partial t} = \frac{\partial\theta}{\partial\tilde{t}}\frac{\partial\tilde{t}}{\partial t} = \frac{1}{t_0}\frac{\partial\theta}{\partial\tilde{t}}$. Similarly, $\nabla p = \frac{p_0}{X_0}\tilde{\nabla}\tilde{p}$. The nondimensional model is then written

$$\frac{\partial\theta}{\partial\tilde{t}} + \tilde{\nabla} \cdot (\theta\tilde{V}) = 0, \quad (5.38)$$

$$\tilde{\nabla} \cdot \left(\tilde{V} - \frac{(1-\theta)^3}{\theta} \left(\frac{p_0 t_0}{\varphi X_0^2} \right) \tilde{\nabla}\tilde{p} \right) = 0, \quad (5.39)$$

$$\tilde{\nabla} \cdot \left(\theta \left(\frac{\lambda}{p_0 t_0} (\tilde{\nabla} \cdot \tilde{V}) \underline{\underline{I}} + \frac{2\mu}{p_0 t_0} \underline{\underline{\tilde{E}}} \right) \right) - \tilde{\nabla}\tilde{p} = -\tilde{\nabla}(\theta\tilde{\psi}) + \frac{2\mu X_0^2}{p_0 t_0 h^2} \theta \tilde{V}, \quad (5.40)$$

where $\underline{\underline{\tilde{E}}} = \frac{1}{2}(\tilde{\nabla}\tilde{v} + (\tilde{\nabla}\tilde{v})^T)$.

Note that $\frac{p_0 t_0}{\varphi X_0^2}$, $\frac{\lambda}{p_0 t_0}$, $\frac{2\mu}{p_0 t_0}$, and $\frac{2\mu X_0^2}{p_0 t_0 h^2}$ must be nondimensional ratios.

Choose X_0 , p_0 , t_0 to simplify above nondimensional ratios. Choose $X_0 = \hbar =$ average of h , then $\frac{p_0 t_0}{\varphi \hbar^2}$, $\frac{\lambda}{p_0 t_0}$, $\frac{2\mu}{p_0 t_0}$, and $\frac{2\mu}{p_0 t_0} \frac{\hbar^2}{h^2}$. Now choose either $p_0 t_0 = \lambda$ or $p_0 t_0 = \varphi \hbar^2$ or $p_0 t_0 = 2\mu$. Pick $p_0 t_0 = \lambda$, then

$$\begin{aligned} \frac{p_0 t_0}{\varphi \hbar^2} &= \frac{\lambda}{\varphi \hbar^2} \equiv \beta, \\ \frac{2\mu}{p_0 t_0} &= \frac{2\mu}{\lambda} \equiv \alpha, \end{aligned}$$

where β is known as dimensionless permeability. Dropping tildes, the nondimensional model is then written

$$\frac{\partial\theta}{\partial t} + \nabla \cdot (\theta V) = 0, \quad (5.41)$$

$$\nabla \cdot \left(V - \frac{(1-\theta)^3}{\theta} \beta \nabla p \right) = 0. \quad (5.42)$$

$$\nabla \cdot (\theta((\nabla \cdot V)\underline{\underline{I}} + \alpha \underline{\underline{E}})) - \nabla p = -\nabla(\theta\psi) + \alpha \left(\frac{\hbar}{h} \right)^2 \theta V, \quad (5.43)$$

This nondimensional model, (5.41) - (5.43), has only three dimensionless parameters.

5.1.4 Stability Analysis

There are four physical coefficients, φ , μ , λ and ψ in the system, (5.35) - (5.37). In our study, we assume μ , λ and φ are constants. We assume ψ is nonlinear and θ dependent only. A possible distribution of ψ is $\psi = \bar{\psi} \sin(n\pi\theta)$ with $\bar{\psi}$ a constant. See Fig. 5.1. In

this figure, the distributions with $n = 1.1$, $n = 1.2$, $n = 1.3$, and $n = 1.4$, respectively, are shown. To understand the behavior of the system, (5.35) - (5.37), linear stability analysis was completed.

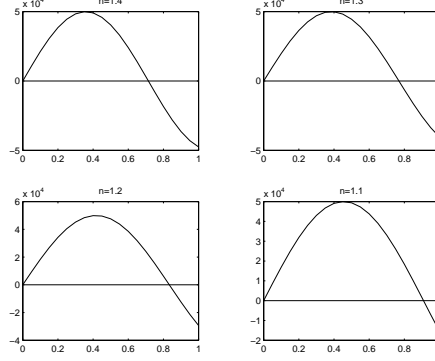


Figure 5.1: Possible distribution of ψ

In 2D, $\underline{V} = \begin{pmatrix} u \\ v \end{pmatrix}$. We write the system, (5.35) - (5.37), in detail under Cartesian coordinate system

$$\frac{\partial \theta}{\partial t} + \frac{\partial}{\partial x}(\theta u) + \frac{\partial}{\partial y}(\theta v) = 0, \quad (5.44)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} - \frac{\partial}{\partial x} \left(\frac{(1-\theta)^3}{\varphi \theta} \frac{\partial p}{\partial x} \right) - \frac{\partial}{\partial y} \left(\frac{(1-\theta)^3}{\varphi \theta} \frac{\partial p}{\partial y} \right) = 0, \quad (5.45)$$

$$\frac{\partial}{\partial x} \left[\theta \left((\lambda + 2\mu) \frac{\partial u}{\partial x} + \lambda \frac{\partial v}{\partial y} \right) \right] + \frac{\partial}{\partial y} \left[\theta \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] - \frac{\partial p}{\partial x} = -\frac{\partial}{\partial x}(\theta \psi) + \frac{2\mu}{h^2} \theta u, \quad (5.46)$$

$$\frac{\partial}{\partial x} \left[\theta \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \frac{\partial}{\partial y} \left[\theta \left(\lambda \frac{\partial u}{\partial x} + (\lambda + 2\mu) \frac{\partial v}{\partial y} \right) \right] - \frac{\partial p}{\partial y} = -\frac{\partial}{\partial y}(\theta \psi) + \frac{2\mu}{h^2} \theta v. \quad (5.47)$$

One of the steady states of the model, which can be easily verified, is as follows

$$\begin{bmatrix} \theta \\ u \\ v \\ p \end{bmatrix} = \begin{bmatrix} \theta_0 \\ u_0 \\ v_0 \\ p_0 \end{bmatrix} = \begin{bmatrix} \theta_0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (5.48)$$

where θ_0 is a constant.

Theorem: Given the system, (5.35) - (5.37), with μ , λ , φ and h of constants and ψ depending on θ only, the system is unstable to spatially periodic disturbances if

$$\psi|_{\theta_0} + \theta_0 \frac{\partial \psi}{\partial \theta} |_{\theta_0} > 0. \quad (5.49)$$

Sketch of the proof. Linearizing the system at the steady state to first order of a small quantity ε , i.e., letting

$$\begin{bmatrix} \theta \\ u \\ v \\ p \end{bmatrix} = \begin{bmatrix} \theta_0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \varepsilon \begin{bmatrix} \theta_1 \\ u_1 \\ v_1 \\ p_1 \end{bmatrix}, \quad 0 < \varepsilon \ll 1, \quad (5.50)$$

and plugging it into (5.44), ignoring the higher order term $O(\varepsilon^2)$, we get the simplified form for $O(\varepsilon)$ term

$$\frac{\partial \theta_1}{\partial t} + \theta_0 \frac{\partial(u_1)}{\partial x} + \theta_0 \frac{\partial(v_1)}{\partial y} = 0 \quad (5.51)$$

Expanding the term of $\frac{(1-\theta)^3}{\varphi\theta}$ via Taylor expansion, we get

$$\frac{(1-\theta)^3}{\varphi\theta} = \frac{(1-\theta_0)^3}{\varphi\theta_0} + O(\varepsilon),$$

where φ is assumed as a constant. Substituting above equation along with (5.50) into (5.45) and ignoring the higher order terms similar with what we did to (5.44), we get the $O(\varepsilon)$ terms

$$\frac{\partial u_1}{\partial x} + \frac{\partial v_1}{\partial y} - \frac{(1-\theta_0)^3}{\varphi\theta_0} \frac{\partial^2 p_1}{\partial x^2} - \frac{(1-\theta_0)^3}{\varphi\theta_0} \frac{\partial^2 p_1}{\partial y^2} = 0. \quad (5.52)$$

Now expanding ψ at the steady state via Taylor expansion, we get

$$\psi = \psi(\theta_0) + \frac{\partial \psi}{\partial \theta}(\theta_0)\varepsilon\theta_1 + O(\varepsilon^2).$$

Substituting the above expression along with (5.50) into (5.46) and ignoring the higher order terms similar to what we did to (5.44), we get the $O(1)$ terms

$$\frac{\partial}{\partial x}(\theta_0\psi_0) = 0, \quad (5.53)$$

which is true if ψ depends on θ only, and $O(\varepsilon)$ terms

$$\begin{aligned} & \theta_0(\lambda + 2\mu) \frac{\partial^2 u_1}{\partial x^2} + \theta_0\mu \frac{\partial^2 u_1}{\partial y^2} + \theta_0(\lambda + \mu) \frac{\partial}{\partial x} \frac{\partial v_1}{\partial y} - \frac{\partial p_1}{\partial x} \\ & = -\theta_0 \frac{\partial}{\partial x} \left(\frac{\partial \psi}{\partial \theta}(\theta_0)\theta_1 \right) - \frac{\partial}{\partial x}(\psi(\theta_0)\theta_1) + \frac{2\mu}{h^2} \theta_0 u_1. \end{aligned} \quad (5.54)$$

Similarly, substituting the above expression along with (5.50) into (5.47), we get the $O(1)$ terms

$$\frac{\partial}{\partial y}(\theta_0\psi_0) = 0, \quad (5.55)$$

and $O(\varepsilon)$ terms

$$\begin{aligned} & \theta_0(\lambda + \mu) \frac{\partial}{\partial x} \frac{\partial u_1}{\partial y} + \theta_0 \mu \frac{\partial^2 v_1}{\partial x^2} + \theta_0(\lambda + 2\mu) \frac{\partial^2 v_1}{\partial y^2} - \frac{\partial p_1}{\partial y} \\ &= -\theta_0 \frac{\partial}{\partial y} \left(\frac{\partial \psi}{\partial \theta}(\theta_0) \theta_1 \right) - \frac{\partial}{\partial y} (\psi(\theta_0) \theta_1) + \frac{2\mu}{h^2} \theta_0 v_1. \end{aligned} \quad (5.56)$$

For the purpose of doing linear stability analysis, we assume that

$$\begin{bmatrix} \theta_1 \\ u_1 \\ v_1 \\ p_1 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} e^{\gamma t + ik_1 x + ik_2 y}, \quad (5.57)$$

where c_1, c_2, c_3 and c_4 are constants and k_1 and k_2 are real.

Substituting (5.57) into (5.51), (5.52), (5.54) and (5.56), we obtain the following linearized system

$$\begin{bmatrix} \gamma & ik_1 \theta_0 & ik_2 \theta_0 & 0 \\ 0 & ik_1 & ik_2 & F \\ A & B & C & ik_1 \\ D & C & E & ik_2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (5.58)$$

where

$$\begin{aligned} A &= -i \left(\theta_0 \frac{\partial \psi}{\partial \theta}(\theta_0) + \psi(\theta_0) \right) k_1, \\ B &= \theta_0 \left((\lambda + 2\mu) k_1^2 + \mu k_2^2 + \frac{2\mu}{h^2} \right), \\ C &= \theta_0(\lambda + \mu) k_1 k_2, \\ D &= -i \left(\theta_0 \frac{\partial \psi}{\partial \theta}(\theta_0) + \psi(\theta_0) \right) k_2, \\ E &= \theta_0 \left(\mu k_1^2 + (\lambda + 2\mu) k_2^2 + \frac{2\mu}{h^2} \right), \\ F &= \frac{(1 - \theta_0)^3}{\varphi \theta_0} (k_1^2 + k_2^2). \end{aligned}$$

To obtain non-trivial solutions to the above system, γ needs to satisfy

$$\gamma = \frac{i\theta_0 F(k_1(AE - CD) - k_2(AC - BD))}{BEF - C^2 F + k_1^2 E + k_2^2 B - 2k_1 k_2 C}. \quad (5.59)$$

It follows that there exists a family of solutions to (5.58)

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} \frac{F\theta_0}{\gamma} \\ -\frac{ik_1k_2\gamma+iCF\gamma+k_2AF\theta_0}{\gamma(k_1C-k_2B)} \\ -\frac{ik_1^2\gamma+iBF\gamma+k_1AF\theta_0}{\gamma(k_1C-k_2B)} \\ 1 \end{bmatrix} c_4 \quad (5.60)$$

Substituting the expressions of A , B , C , D , E and F into (5.59), we obtain

$$\gamma = \frac{(1-\theta_0)^3(k_1^2+k_2^2)\left(\psi(\theta_0)+\theta_0\frac{\partial\psi}{\partial\theta}(\theta_0)\right)}{(1-\theta_0)^3\left[(\lambda+2\mu)(k_1^2+k_2^2)+\frac{2\mu}{h}\right]+\varphi} \quad (5.61)$$

Note as a result that γ is always real and $\gamma(k_1^2+k_2^2)$ is bounded. Note also that $(1-\theta)^3$ is always greater than 0 when $0 < \theta < 1$, $k_1^2+k_2^2 > 0$, λ , μ , φ , and h are positive. Hence the sign of γ depends on the following expression:

$$\xi = \psi(\theta_0) + \theta_0 \frac{\partial\psi}{\partial\theta}(\theta_0). \quad (5.62)$$

In the range where $\gamma > 0$ patterns will grow. $\gamma > 0$ when

$$\frac{\psi(\theta_0)}{\theta_0} > -\frac{\partial\psi}{\partial\theta}(\theta_0)$$

Note that the above inequality depends only upon the values of θ_0 , $\psi(\theta_0)$ and $\frac{\partial\psi}{\partial\theta}(\theta_0)$. From this relationship, we can find the region of the θ for a given function of ψ in which the pattern increases.

5.2 Numerical Methods

The system of equations is a combination of hyperbolic equation and elliptic equations. It shows that state variables, u , v and p , are implicitly dependent on t . Numerical computation is based on the nondimensional system, (5.41) - (5.43).

We utilized the upwind scheme in CLAWPACK [13, 90, 75], to solve the hyperbolic equation (5.41) for volume fraction of cells and fibers θ . The central finite difference method, known as a second order scheme, has been used to solve the elliptic equations, (5.42) and (5.43), for velocities and pressure. The multigrid DMGD9V solver [91, 92] has been used to solve for the system of difference equations for velocity.

Following is the upwind scheme for two dimensional hyperbolic equation:

$$\Theta_{i,j}^{n+1} = \begin{cases} \Theta_{i,j}^n - \frac{\Delta t}{\Delta x}((\Theta U)_{i+1,j}^n - (\Theta U)_{i,j}^n) - \frac{\Delta t}{\Delta y}((\Theta V)_{i,j+1}^n - (\Theta V)_{i,j}^n), & \text{if } U_{i,j}^n < 0, V_{i,j}^n < 0, \\ \Theta_{i,j}^n - \frac{\Delta t}{\Delta x}((\Theta U)_{i+1,j}^n - (\Theta U)_{i,j}^n) - \frac{\Delta t}{\Delta y}((\Theta V)_{i,j}^n - (\Theta V)_{i,j-1}^n), & \text{if } U_{i,j}^n < 0, V_{i,j}^n > 0, \\ \Theta_{i,j}^n - \frac{\Delta t}{\Delta x}((\Theta U)_{i,j}^n - (\Theta U)_{i-1,j}^n) - \frac{\Delta t}{\Delta y}((\Theta V)_{i,j+1}^n - (\Theta V)_{i,j}^n), & \text{if } U_{i,j}^n > 0, V_{i,j}^n < 0, \\ \Theta_{i,j}^n - \frac{\Delta t}{\Delta x}((\Theta U)_{i,j}^n - (\Theta U)_{i-1,j}^n) - \frac{\Delta t}{\Delta y}((\Theta V)_{i,j}^n - (\Theta V)_{i,j-1}^n), & \text{if } U_{i,j}^n > 0, V_{i,j}^n > 0, \end{cases} \quad (5.63)$$

where $\Theta_{i,j}^n$ is the numerical solution of volume fraction at time step n and mesh grid (x_i, y_j) , $U_{i,j}^n, V_{i,j}^n$ are the numerical solution of velocities at time step n and mesh grid (x_i, y_j) , Δx and Δy are the mesh sizes and Δt is the time step size.

The central finite difference method, known as a 2^{nd} order scheme, has been used to solve for the velocities and pressure. We apply this scheme to (5.42) and (5.43):

$$\begin{aligned} & \frac{\tilde{\alpha}_{i-1/2,j}}{(\Delta x)^2} U_{i-1,j} + \frac{\tilde{\alpha}_{i+1/2,j}}{(\Delta x)^2} U_{i+1,j} + \frac{\tilde{\beta}_{i,j-1/2}}{(\Delta y)^2} U_{i,j-1} + \frac{\tilde{\beta}_{i,j+1/2}}{(\Delta y)^2} U_{i,j+1} \\ & - \left(\frac{\tilde{\alpha}_{i-1/2,j} + \tilde{\alpha}_{i+1/2,j}}{(\Delta x)^2} + \frac{\tilde{\beta}_{i,j-1/2} + \tilde{\beta}_{i,j+1/2}}{(\Delta y)^2} + 2\tilde{\beta}_{i,j} \right) U_{i,j} \\ & = \frac{P_{i+1,j} - P_{i-1,j}}{2\Delta x} - \frac{\Theta \psi_{i+1,j} - \Theta \psi_{i-1,j}}{2\Delta x} \\ & - \frac{1}{4\Delta x \Delta y} ((\tilde{\gamma}_{i+1,j} + \tilde{\beta}_{i,j+1}) V_{i+1,j+1} - (\tilde{\gamma}_{i+1,j} + \tilde{\beta}_{i,j-1}) V_{i+1,j-1} \\ & - (\tilde{\gamma}_{i-1,j} + \tilde{\beta}_{i,j+1}) V_{i-1,j+1} + (\tilde{\gamma}_{i-1,j} + \tilde{\beta}_{i,j-1}) V_{i-1,j-1}), \end{aligned} \quad (5.64)$$

$$\begin{aligned} & \frac{\tilde{\beta}_{i-1/2,j}}{(\Delta x)^2} V_{i-1,j} + \frac{\tilde{\beta}_{i+1/2,j}}{(\Delta x)^2} V_{i+1,j} + \frac{\tilde{\alpha}_{i,j-1/2}}{(\Delta y)^2} V_{i,j-1} + \frac{\tilde{\alpha}_{i,j+1/2}}{(\Delta y)^2} V_{i,j+1} \\ & - \left(\frac{\tilde{\beta}_{i-1/2,j} + \tilde{\beta}_{i+1/2,j}}{(\Delta x)^2} + \frac{\tilde{\alpha}_{i,j-1/2} + \tilde{\alpha}_{i,j+1/2}}{(\Delta y)^2} + 2\tilde{\beta}_{i,j} \right) V_{i,j} \\ & = \frac{P_{i,j+1} - P_{i,j-1}}{2\Delta y} - \frac{\Theta \psi_{i,j+1} - \Theta \psi_{i,j-1}}{2\Delta y} \\ & - \frac{1}{4\Delta x \Delta y} ((\tilde{\beta}_{i+1,j} + \tilde{\gamma}_{i,j+1}) U_{i+1,j+1} - (\tilde{\beta}_{i+1,j} + \tilde{\gamma}_{i,j-1}) U_{i+1,j-1} \\ & - (\tilde{\beta}_{i-1,j} + \tilde{\gamma}_{i,j+1}) U_{i-1,j+1} + (\tilde{\beta}_{i-1,j} + \tilde{\gamma}_{i,j-1}) U_{i-1,j-1}), \end{aligned} \quad (5.65)$$

$$\begin{aligned} & \frac{\rho_{i-1/2,j}}{(\Delta x)^2} P_{i-1,j} + \frac{\rho_{i+1/2,j}}{(\Delta x)^2} P_{i+1,j} + \frac{\rho_{i,j-1/2}}{(\Delta y)^2} P_{i,j-1} + \frac{\rho_{i,j+1/2}}{(\Delta y)^2} P_{i,j+1} \\ & - \left(\frac{\rho_{i-1/2,j} + \rho_{i+1/2,j}}{(\Delta x)^2} + \frac{\rho_{i,j-1/2} + \rho_{i,j+1/2}}{(\Delta y)^2} \right) P_{i,j} \\ & = \frac{U_{i+1,j} - U_{i-1,j}}{2\Delta x} + \frac{V_{i,j+1} - V_{i,j-1}}{2\Delta y}, \end{aligned} \quad (5.66)$$

where $\tilde{\alpha} = \Theta(1 + \alpha)$, $\tilde{\beta} = \Theta\alpha/2$, $\tilde{\gamma} = \Theta$ and $\rho = \frac{(1-\Theta)^3}{\Theta}\beta$.

Fig. 5.2 shows the numerical solutions for θ , u , v and p at $t = 0, 3, 6, 9$, and 12 (mesh size is 82 by 82) for the case with piecewise constant initial condition which is comparable with the experimental result for a bioartificial tendon shown in Fig. 1.1 [1] in which the cells contract the gel over time.

5.3 Numerical Examples

The simulations have been done on the nondimensional system, (5.41) - (5.43), with a variety of initial conditions. In all the computation, we choose $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$. And we consider the problem in the domain $0 < x < x_f$, $0 < y < 5$, where $x_f = 5$ or $x_f = 15$.

5.3.1 Cases with Regular Initial Data

In all computations under this subsection, we consider the problem with initial conditions described in (5.8).

We employ a central finite difference scheme to set up the discrete system for u, v and p at each time step. Fig. 5.3 - Fig. 5.4 show the numerical solutions of θ , u , v and p at $t = 0, 5, 10, 15$ and 20 (mesh size is 82 by 82) for the system (5.41) -(5.43) with the initial condition in (5.8), in which $\theta_0 = 0.2$ and $\varepsilon = 0.01$. The algorithm crashed when $t > 49.1$ presumably due to the large gradient of θ . We do the calculation for the case with regular initial condition in order to test the numerical methods. As expected, the cells and fibers contract over time.

5.3.2 Cases with Random Initial Data

In the real world, the perturbation to the initial distribution is not of a regular pattern, but randomly distributed. Hence, in this subsection, we consider the problem with initial conditions described in (5.9) and (5.10) for all computations.

Figs. 5.5 - 5.6 show the numerical solutions for θ , u , v and p to the system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (mesh size is 82 by 82) for the case with $n_{qf} = 5$ and $n_{amp} = 1$. The

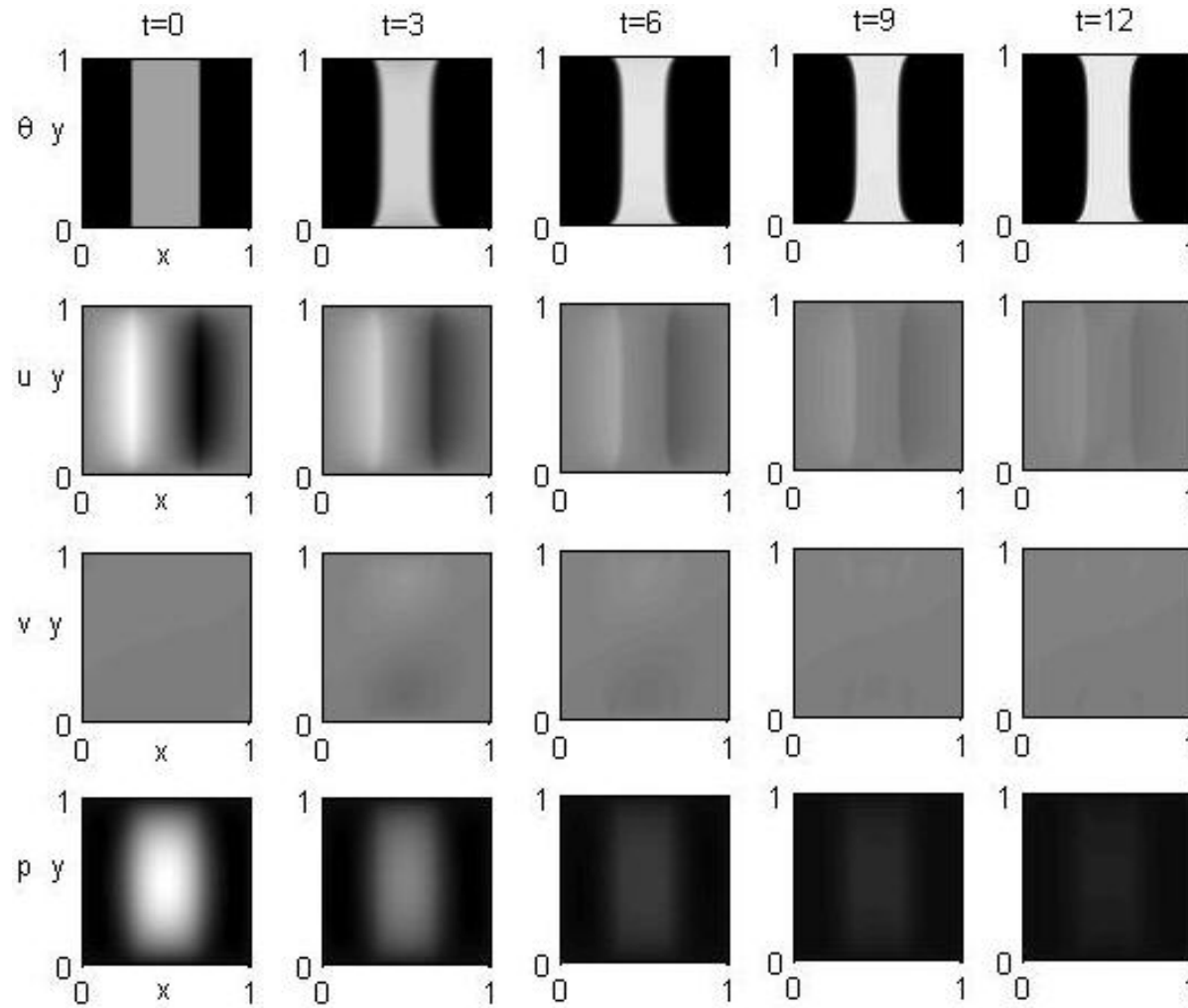


Figure 5.2: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with piecewise constant initial condition, in which $\theta_0 = 0.5$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 3, 6, 9$, and 12 (pcolor plot: 82×82).

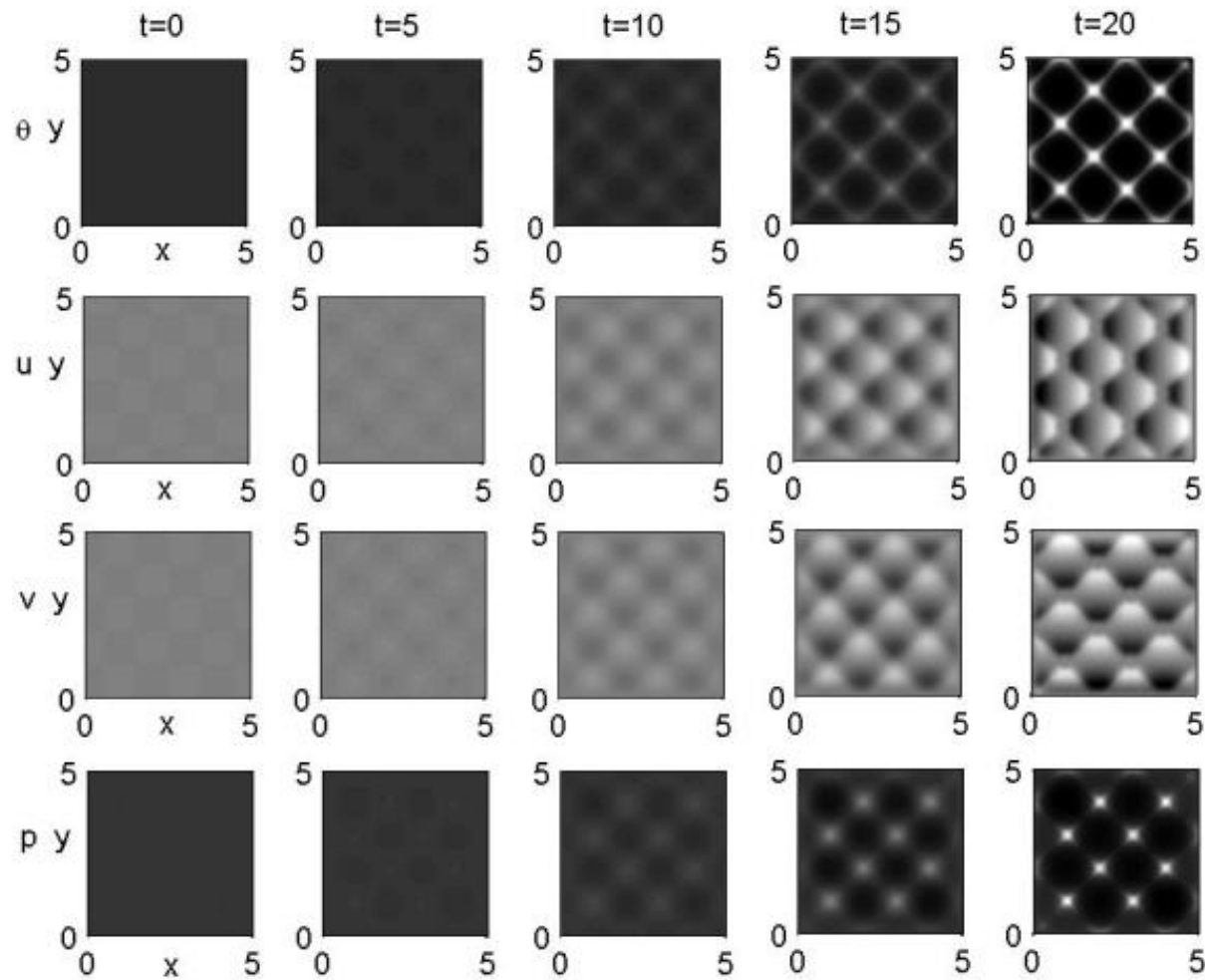


Figure 5.3: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.8), in which $\theta_0 = 0.2$ and $\varepsilon = 0.01$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82).

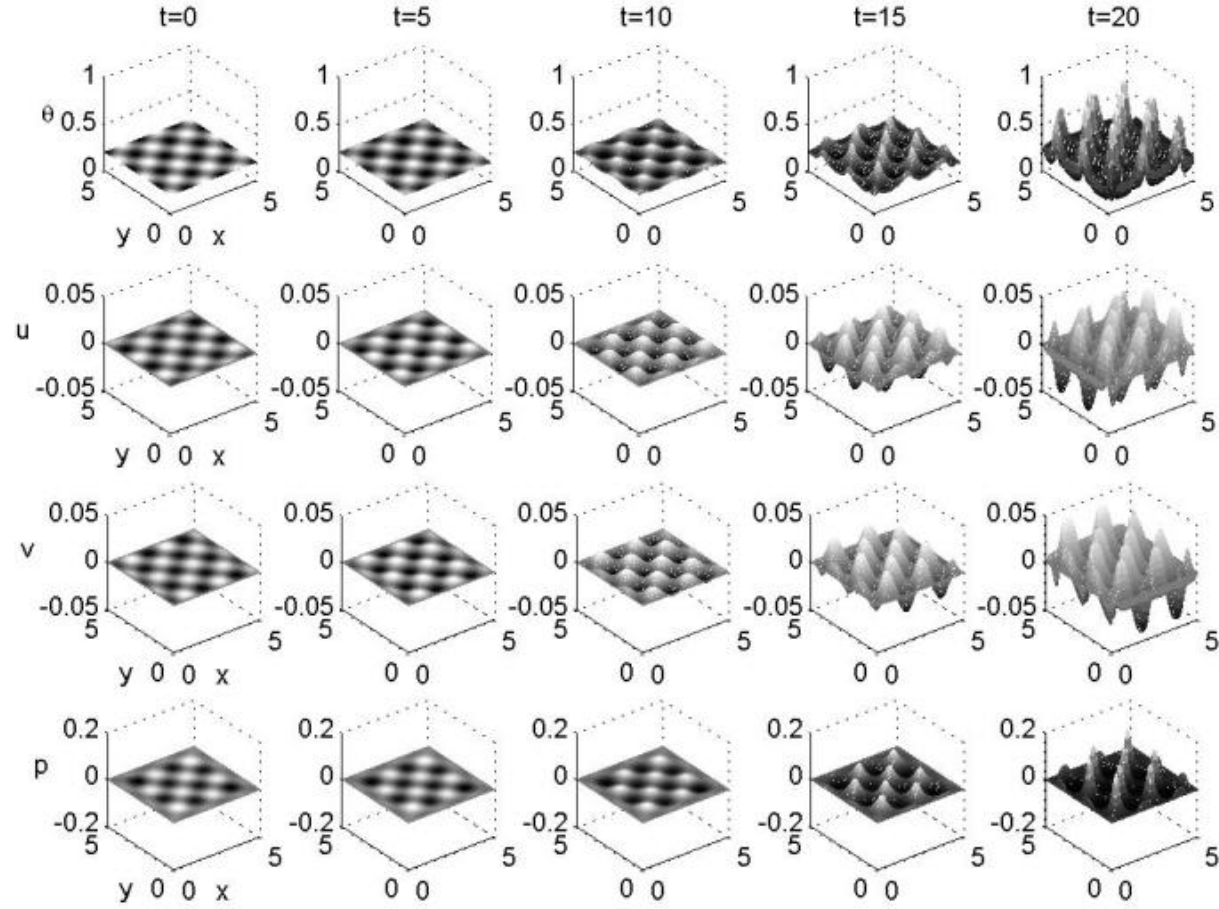


Figure 5.4: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.8), in which $\theta_0 = 0.2$ and $\varepsilon = 0.01$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82).

algorithm crashed when $t > 32.1$ presumably due to the large gradient of θ .

Figs. 5.7 - 5.8 show the numerical solutions for another sample with different initial distribution of θ . This case is similar to the last one except that the magnitudes of perturbations are randomly chosen from a uniform $(0, 1)$ distribution. This results in a more randomly perturbed initial distribution. The contraction of tissue is observed over time.

Figs. 5.9 - 5.10 show the numerical solutions for θ , u , v and p at $t = 0, 5, 10, 15$ and 20 (mesh size is 82 by 82) for the case with $n_{qf} = 7$ on a 5×5 domain. The algorithm crashed when $t > 28.02$ presumably due to the large gradient of θ . In this calculation, random perturbation has been added to each grid point. This may cause numerical difficulty in calculations.

Fig. 5.11 - Fig. 5.12 show the numerical solutions for θ , u , v and p at $t = 0, 5, 10, 15$ and 20 (mesh size is 194 by 66) for the case with $h(x) = 0.1 + \frac{0.4}{15}x$, and $n_{qf} = 7$ on a 15×5 domain. The algorithm crashed when $t > 29.6$ presumably due to the large gradient of θ . In this calculation, we have shown how the thickness of gel affects the pattern formed by contraction of the tissue.

5.4 Conclusions

In this chapter, we have studied a two dimensional fluid mixture model for tissue deformations. The linear stability analysis has been conducted which gives the range of the parameters for the stability of the system. Two different numerical methods were employed for different types of partial differential equations. CLAWPACK was utilized to solve the volume conservation equation for the volume fraction of cells and fibers θ , and a central finite difference scheme for the velocity u , v and pressure p . The expected numerical solutions have been obtained.

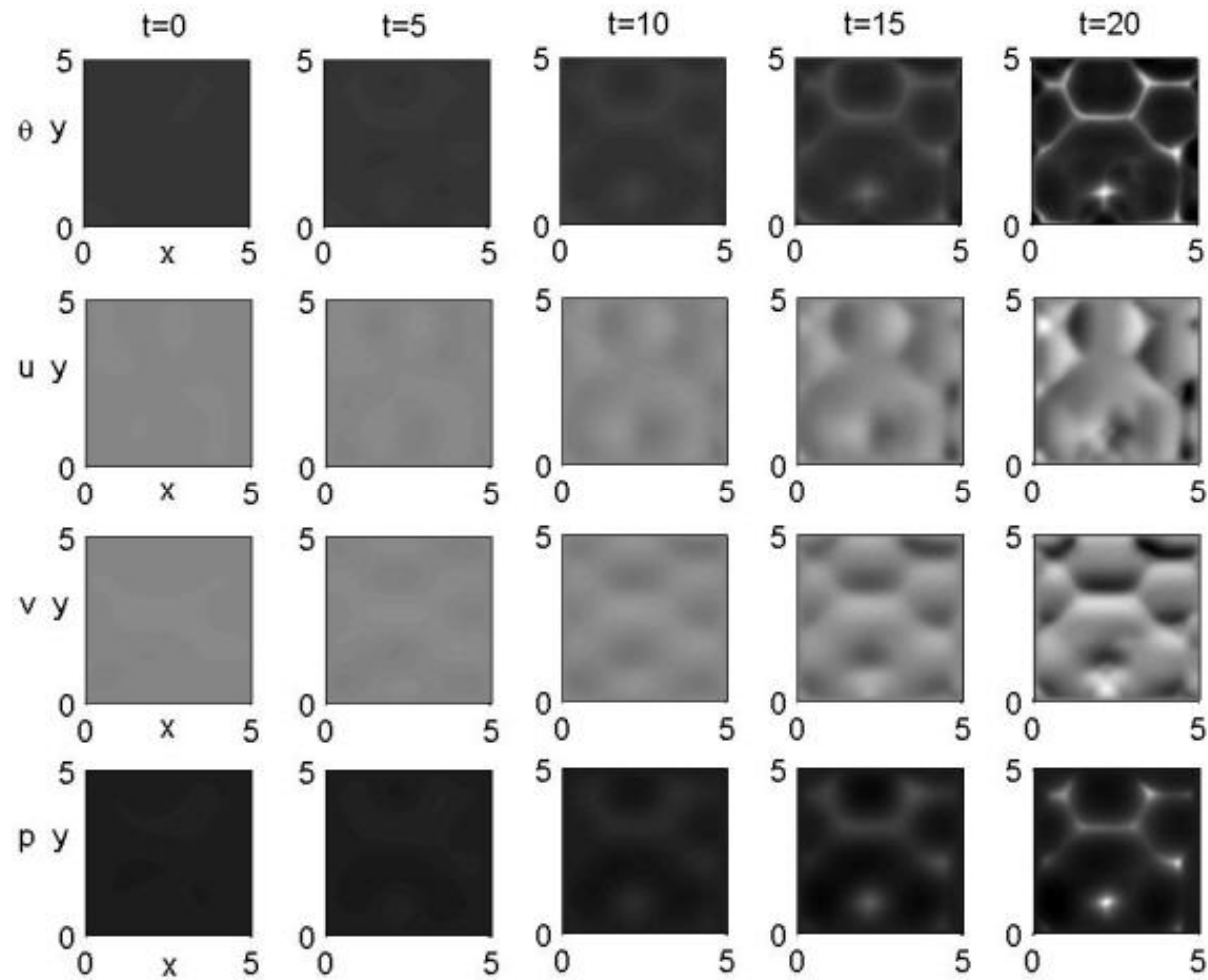


Figure 5.5: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82).

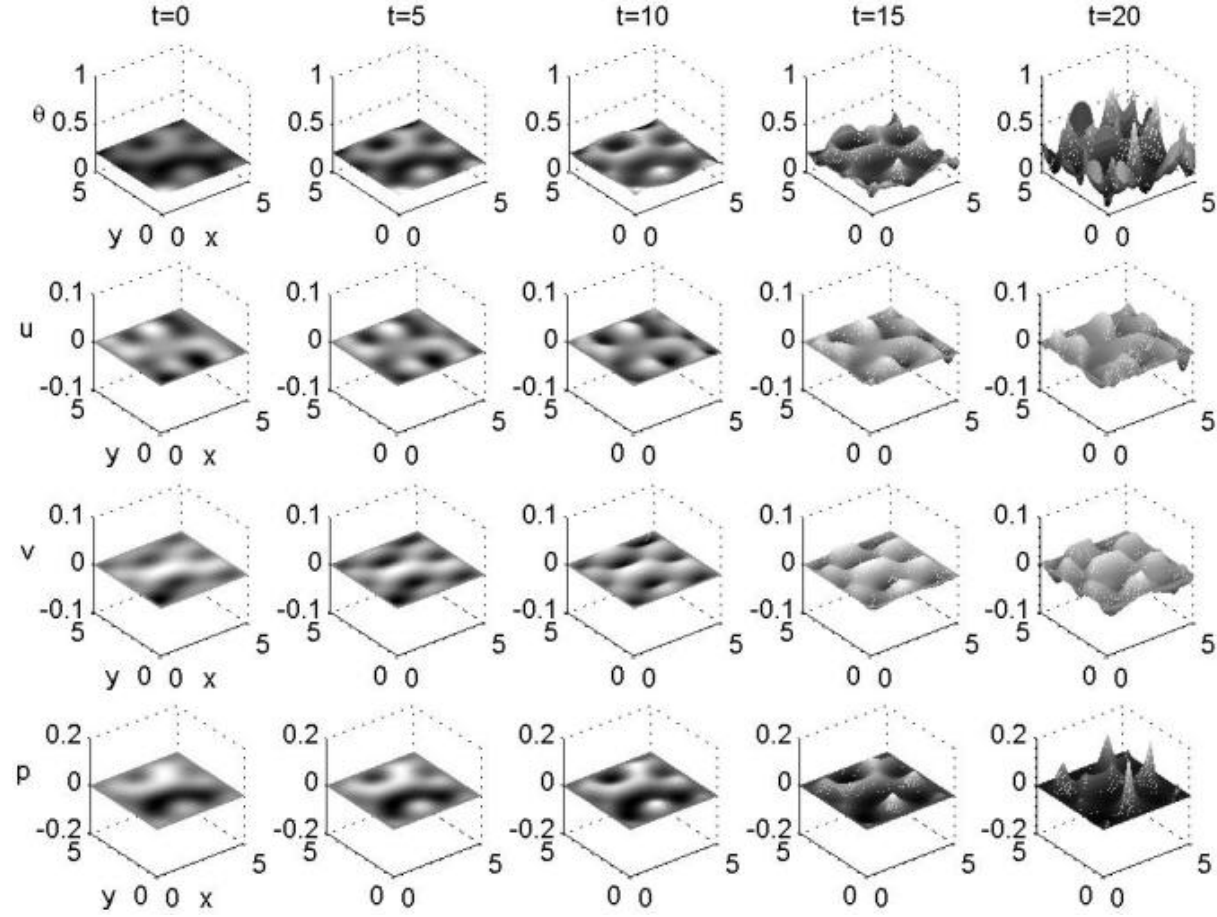


Figure 5.6: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82).

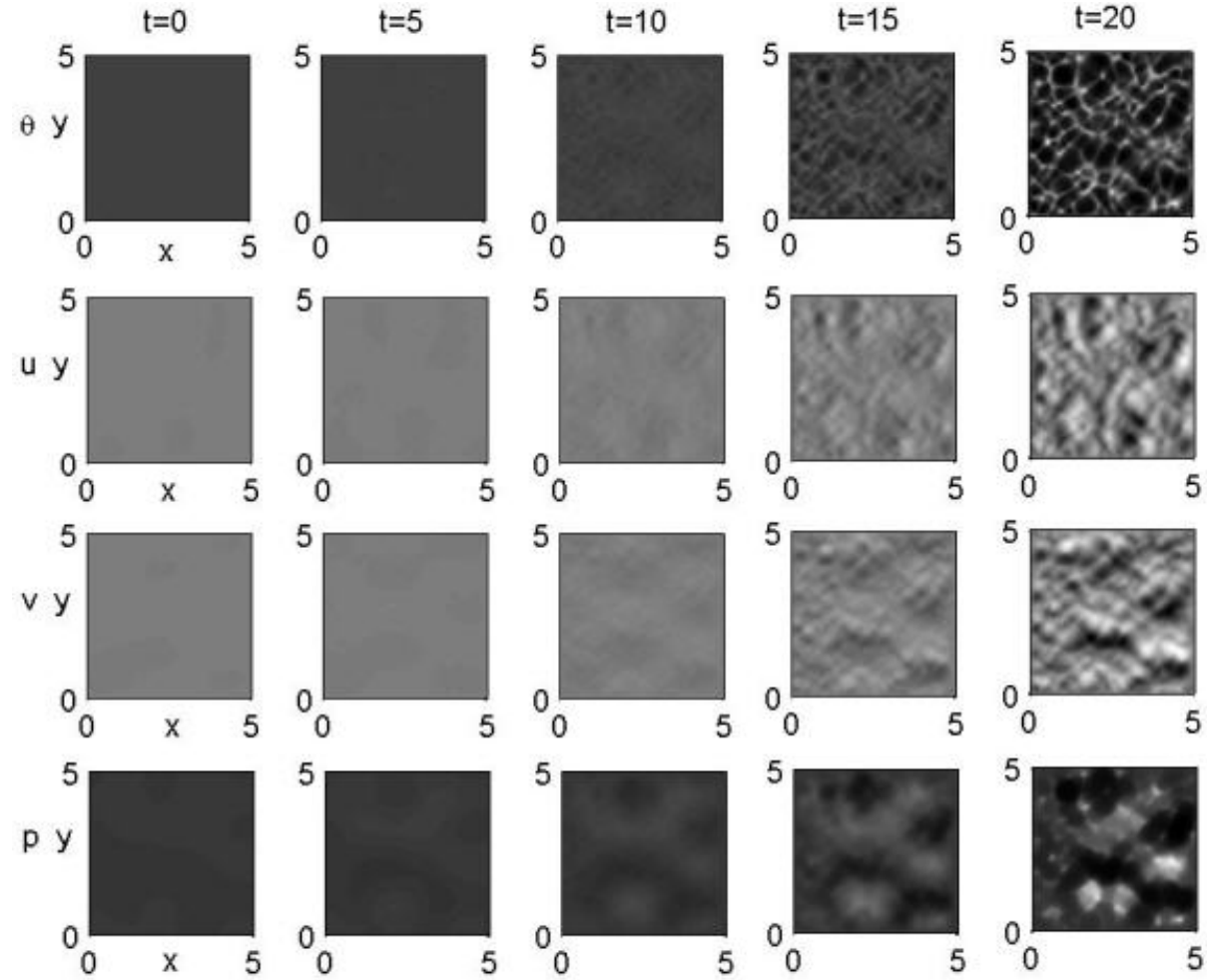


Figure 5.7: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82).

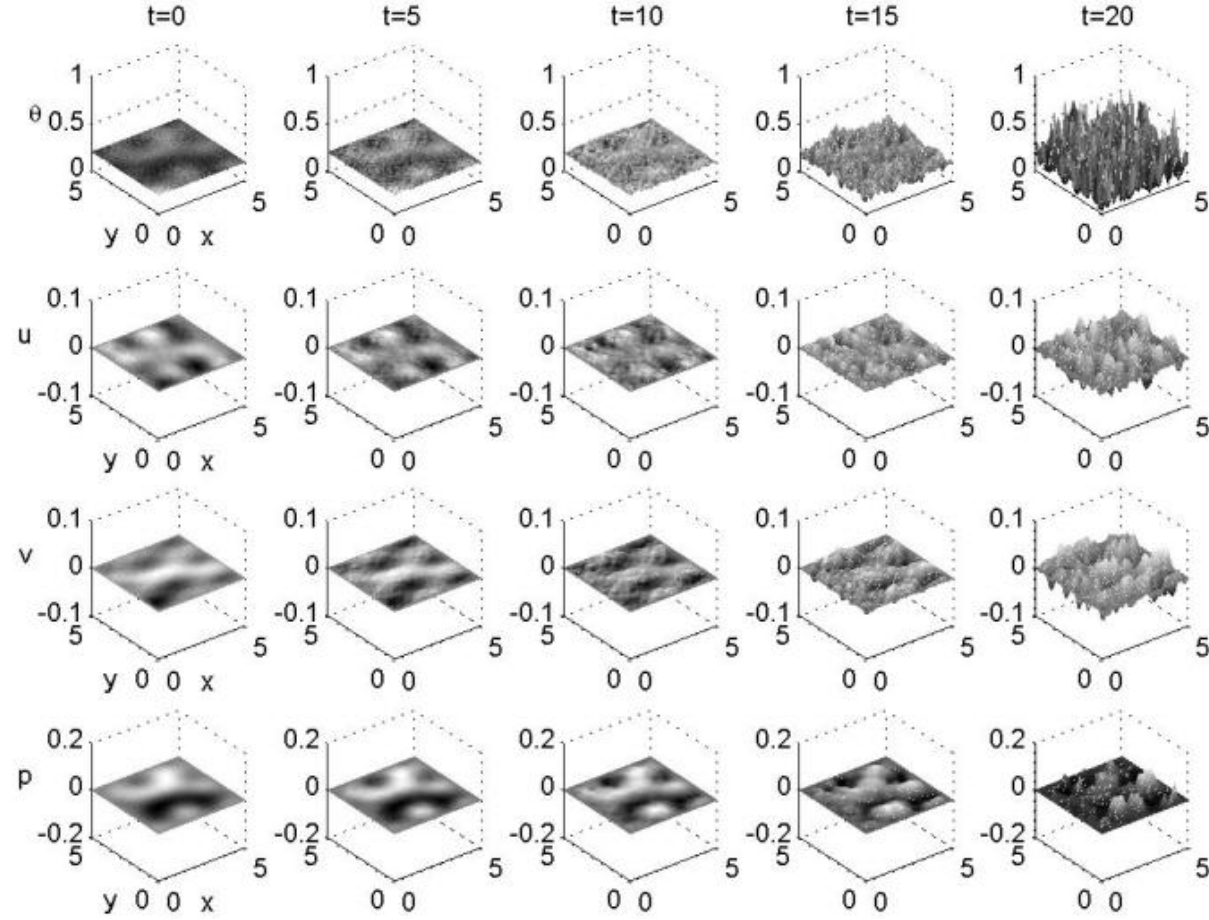


Figure 5.8: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.9), in which $\theta_0 = 0.2$, $n_{qf} = 5$ and $n_{amp} = 1$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82).

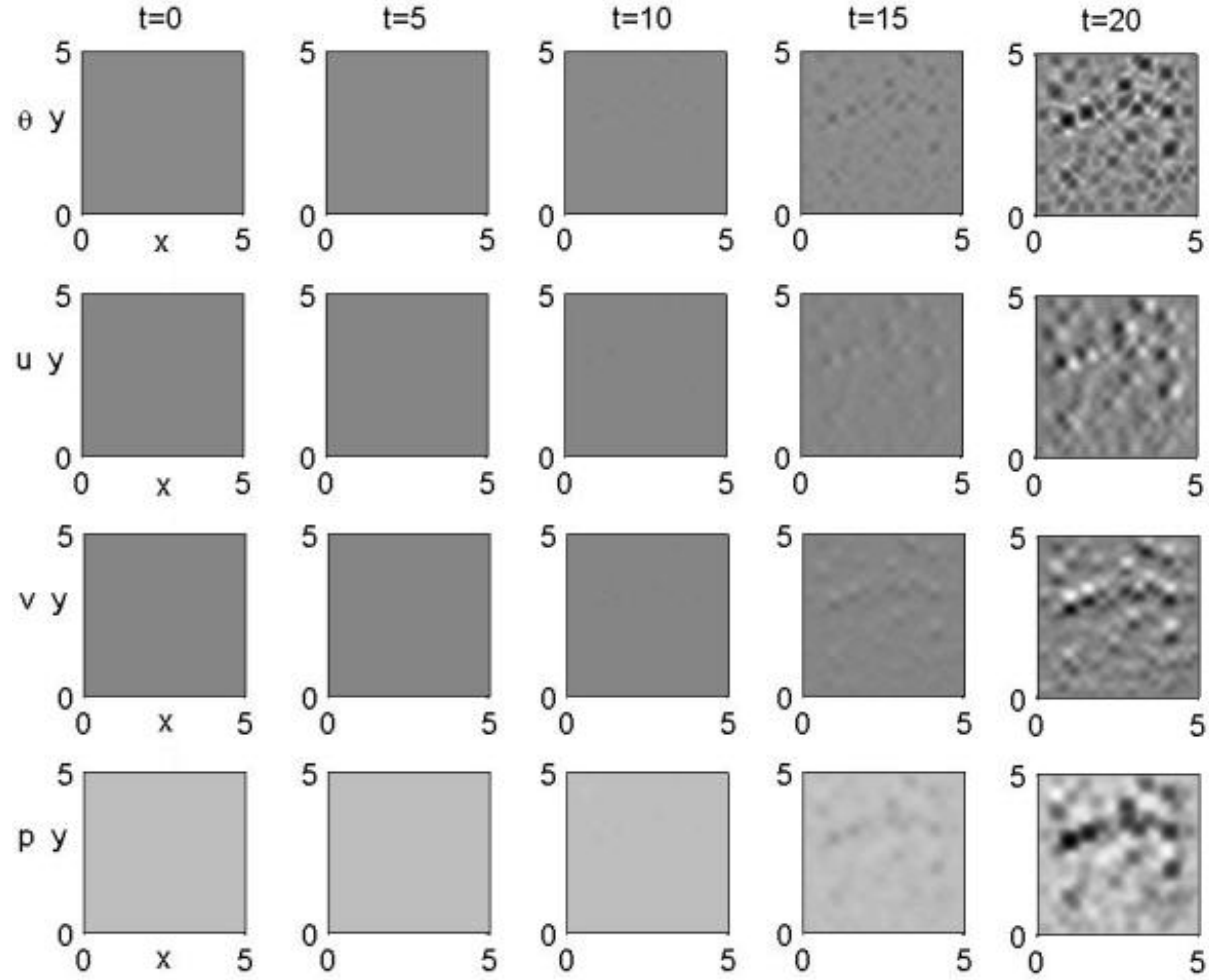


Figure 5.9: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, on a 5×5 domain, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 82×82).

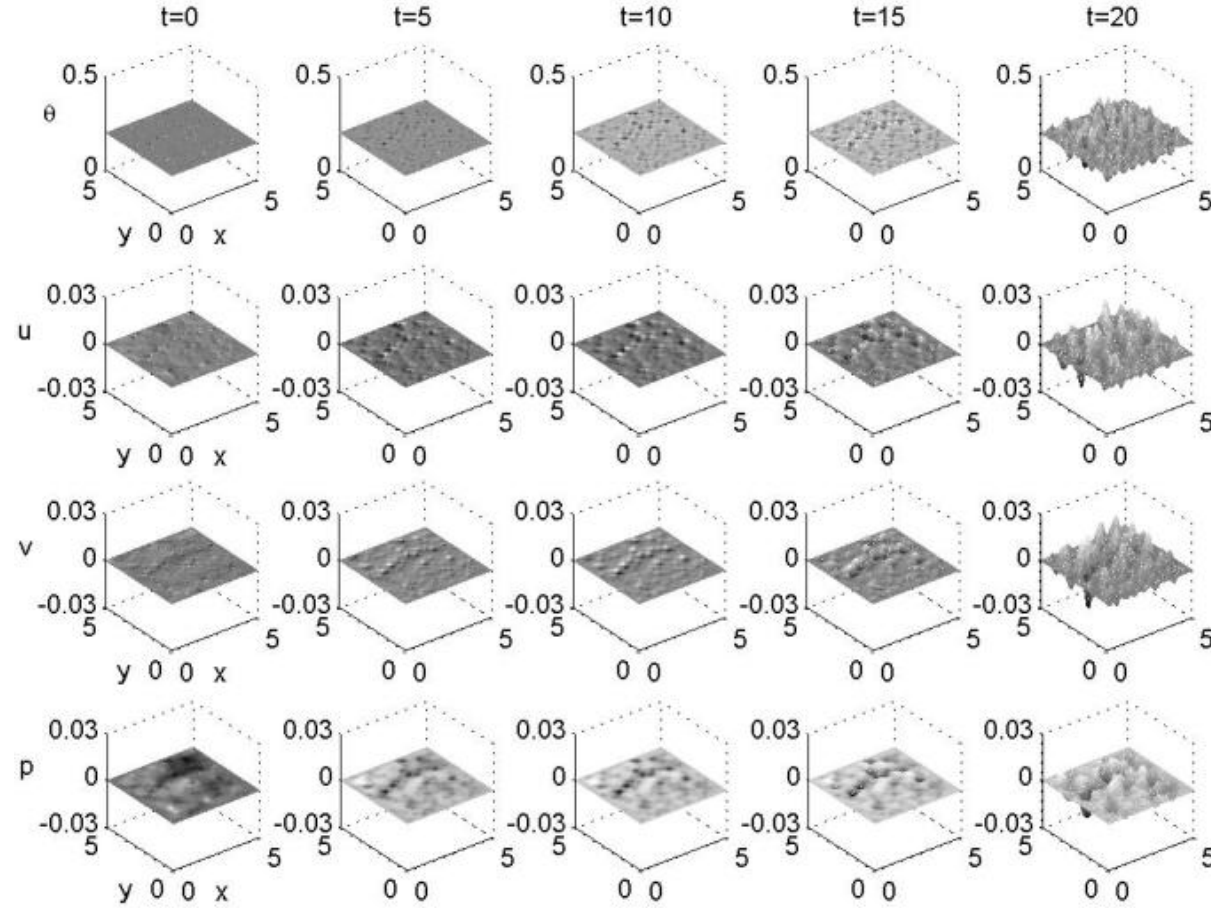


Figure 5.10: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, on a 5×5 domain, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 82×82).

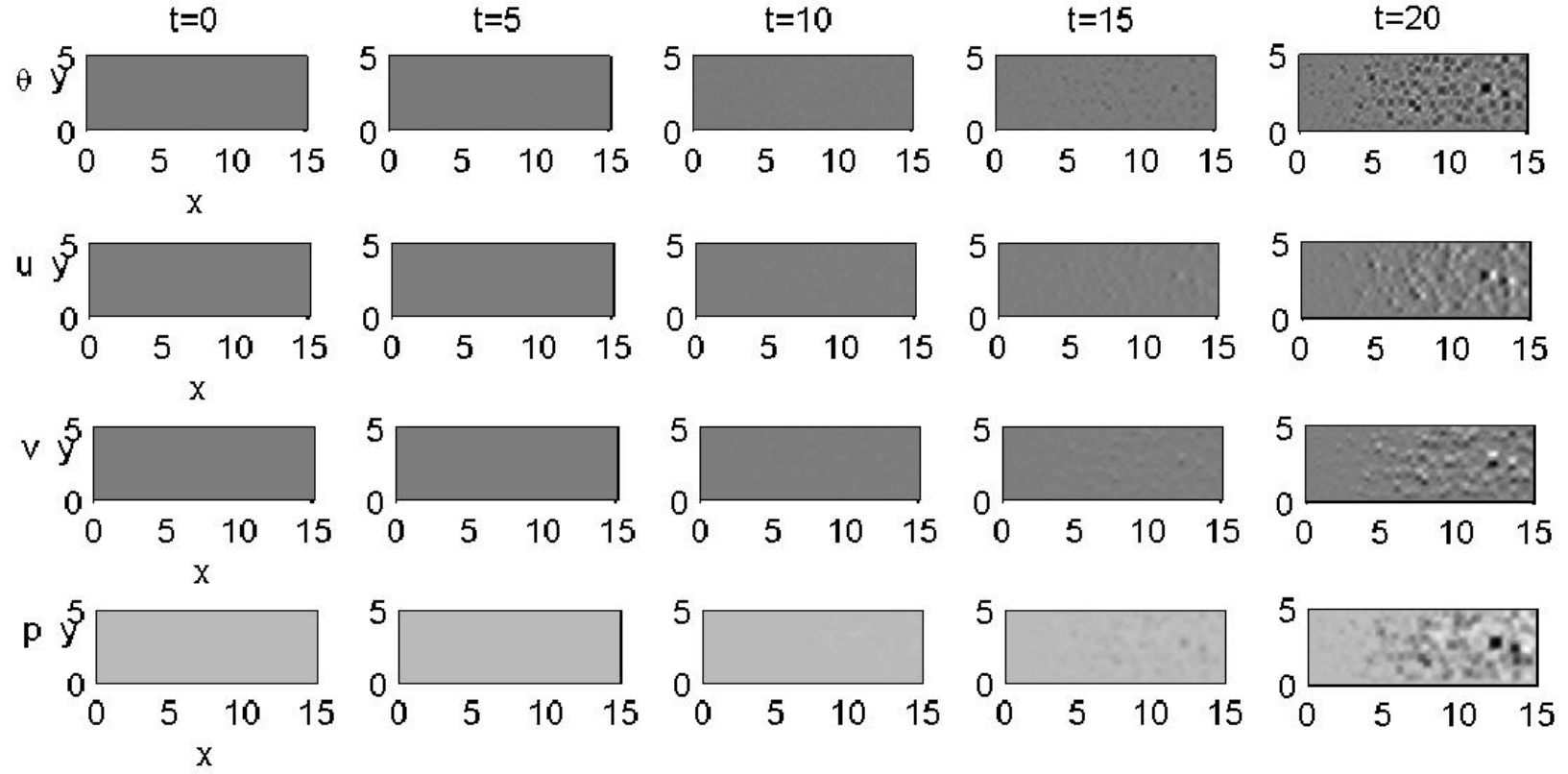


Figure 5.11: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, and dimensional parameters $h(x) = 0.1 + \frac{0.4}{15}x$, on a 15×5 domain, at $t = 0, 5, 10, 15$ and 20 (pcolor plot: 194×66).

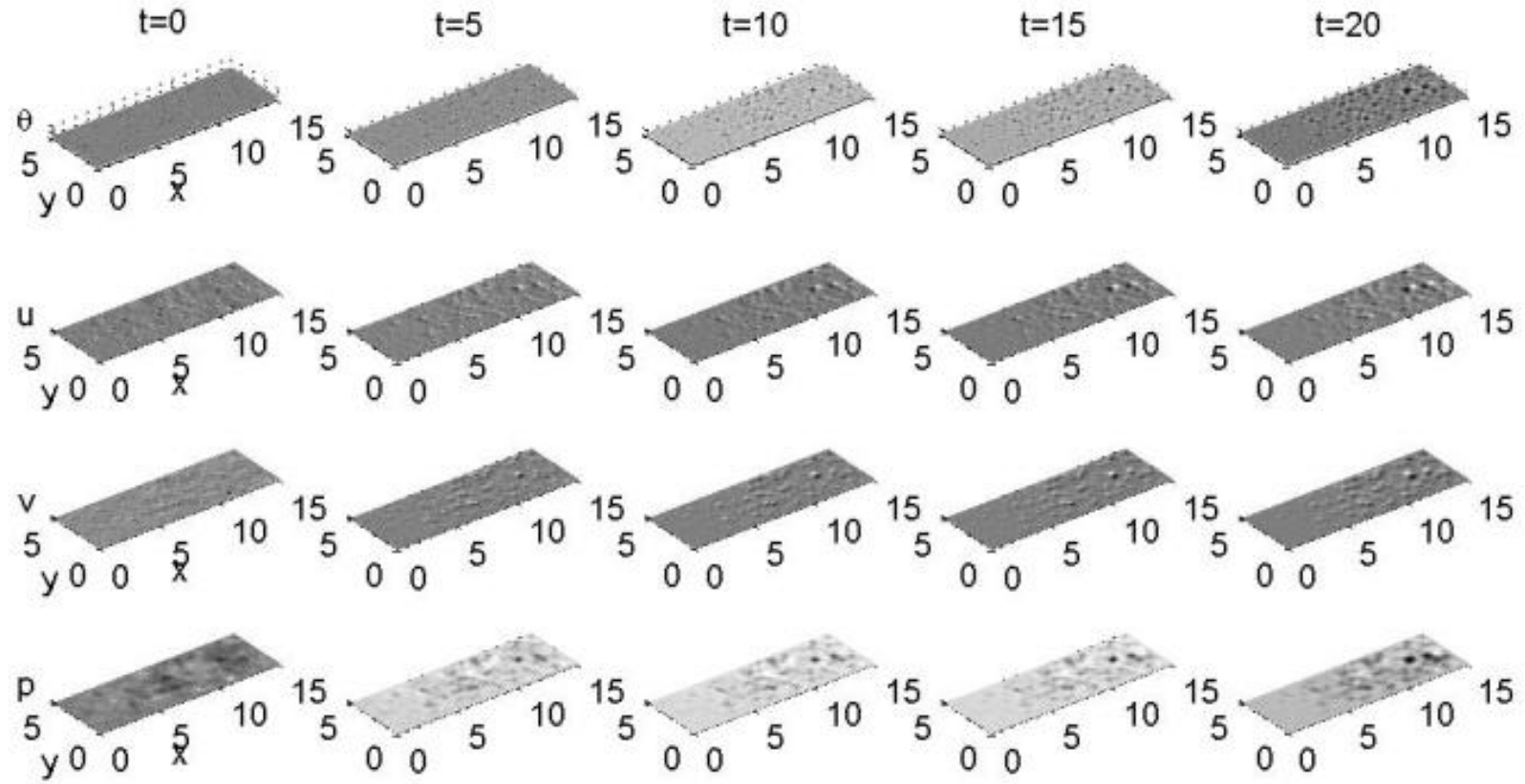


Figure 5.12: Solution of θ , u , v and p to nondimensional system (5.41) - (5.43) with initial condition described in (5.10), in which $\theta_0 = 0.2$ and $n_{qf} = 7$, and with nondimensional parameters $\alpha = 2$, $\beta = 50/9$, and $\psi = 0.5 \sin(1.2\pi\theta)$, and dimensional parameters $h(x) = 0.1 + \frac{0.4}{15}x$, on a 15×5 domain, at $t = 0, 5, 10, 15$ and 20 (mesh plot: 194×66).

Chapter 6

Conclusions and Future Work

6.1 Current Work

Theoretical and numerical analysis for a fluid mixture model of tissue deformation have been performed in this dissertation.

The immersed interface method has been used to solve other type of problems, such as Poisson problems, heat equations, wave equations with fixed interface and homogeneous jump conditions, etc. This is the first known extension of this idea to a system combining hyperbolic and elliptic equations with moving interface and nonhomogeneous jump conditions, resulting in a new high resolution sharp interface method for solving a fluid mixture model of tissue deformations.

The problems analyzed in this dissertation are systems of hyperbolic and elliptic equations with four physical parameters which affect the behavior of the system. In Chapter 2, the one dimensional systems with four types of parameters have been analyzed theoretically and numerically. We found the relationship between those parameters and the pattern of tissue deformations via linear stability analysis. Several numerical experiments support our theoretical analysis. The discontinuities and non-smoothness in the solution introduce numerical difficulties. To maintain accuracy for the velocity v near the discontinuities, we applied the immersed interface method to the elliptic equation in Chapter 2. In order to use the immersed interface method, we need to derive the jump condition for the elliptic equation. According to our analysis, the solution of v is non-smooth, piece-wise linear. So the jump conditions of v across the interfaces are homogeneous but the jump conditions of first derivative of v across the interfaces are nonhomogeneous. By enforcing the jump conditions

across the interfaces at the irregular grid points, the immersed interface method, one of the sharp interface methods, helps eliminate non-physical oscillations near the interfaces.

Comparison of two algorithms has been made for the one dimensional case. One algorithm uses Runge-Kutta with WENO for the hyperbolic equation and central finite difference method for the elliptic equation without tracking the interfaces as time evolves. Another one is using the immersed interface method for the elliptic equation with interfaces tracked via Runge-Kutta method as time evolves. Oscillations are observed near the interfaces when the interfaces are not tracked. The immersed interface method avoids non-physical oscillations, and gives a high resolution solution.

In Chapter 3, we have applied the immersed interface method to a one dimensional linear advection equation with nonhomogeneous jump conditions across a moving interface. Three types of discontinuous initial conditions have been tested: piecewise constant, piecewise C^∞ function (constant + sin function) with the same frequencies on two pieces, and piecewise C^∞ function (constant + sin) with different frequencies on two pieces. For the linear advection equation with a piecewise constant initial condition, we obtained the exact solution. No errors are observed at all. For the linear advection equation with piecewise C^∞ functions (constant + sin), solutions of second order (in L_∞ , L_1 and L_2 norms) were obtained. Our results showed that the immersed interface method is a robust approach for a linear advection equation with discontinuous initial condition and prior knowledge of the jump condition across the interface.

In Chapter 4, the immersed interface method has been applied to a one dimensional fluid mixture model of tissue deformations with all constant physical parameters based on the three-level leapfrog method and two-step MacCormack method. Deriving extra jump conditions for both hyperbolic and elliptic equations is necessary and important. For some types of physical parameters, deriving the extra jump conditions can be very difficult, even impossible for some cases. Solutions of high resolution near the interface have been obtained for a simple physical situation.

In Chapter 5, we derived the model system for a thin two dimensional case from the three dimensional model so that the thickness of the gel is taken into account. The two dimensional systems with constant parameters have been analyzed theoretically and numerically. We found the relationship between those parameters and the pattern of tissue deformations via linear stability analysis. Numerical examples have been completed and the numerical solutions are qualitatively consistent with experimental observation.

6.2 Future Work

In this dissertation, the immersed interface method has been applied to solve a fluid mixture model of tissue deformations which is a system combined with hyperbolic and elliptic equations in one dimension. This method eliminates non-physical oscillations near the interfaces. This work can be extended to multi-dimensional cases with interfaces.

6.2.1 Problems with Variable Coefficients

The physical parameters in the system can vary with the volume fraction of cell fiber phase. Currently, the immersed interface method is used to solve the problem with constant parameters. We can solve similar problems with discontinuous variable parameters equally effectively. The conventional finite difference methods for problems with variable parameters can be used in the smooth region, i.e., away from the discontinuities. The main issue is to find all necessary jump conditions across the interfaces. However, this is not a trivial work to get enough jump conditions to allow the finite difference methods to achieve second order accurate solutions.

6.2.2 Theoretical Analysis for the Immersed Interface Method

For the problems discussed in this dissertation, the local truncation errors near the interface are one order lower than at the rest of the regular grid points. It is commonly believed that the convergence rate of the global error will not be affected. This claim has been proved to be true for some problems [38]. However, it is a challenge to find out whether the above claim is true for all different interface problems (fixed interface or moving interface problems) or whether there are certain conditions under which the claim above is true. More rigorous analysis for the convergence of the immersed interface method is needed.

The application of the immersed interface method to our fluid mixture model for tissue deformations with constant physical parameters and piecewise constant initial condition does not support the claim above. Given an initial distribution of volume fraction of cell and fiber phase θ , the errors of the velocity v solved by the immersed interface method are very small (around $e - 16$). But the error of v is randomly distributed. For the moving interface problem, such distribution can cause an aliasing instability. We are not sure whether this will affect the order of accuracy of the θ or not for our problem. The immersed

interface method did eliminate the non-physical oscillations near the interface and achieve high resolution solution compared with other conventional finite difference methods.

6.2.3 Two Dimensional Problem with Discontinuous Initial Conditions

In our current work, the immersed interface method is implemented for the one dimensional system with constant parameters and discontinuities in initial conditions. This can be extended to the similar two dimensional problem with discontinuous initial conditions, such as the example in Figs 1.1 (from [1]) and 5.2. Working out the jump conditions across the interfaces up to second order derivatives may be challenging. The level set method is needed to implicitly represent the interface in 3D. Keeping track of the irregular grid points near the interface of discontinuities will be another component of the implementation.

6.2.4 Two Dimensional Problem with Non-Discontinuous Initial Conditions

We used the upwind scheme and central finite difference method to solve the fluid mixture model of tissue deformations with continuous initial conditions or randomly distributed initial conditions. We observed that the derivatives of θ grew with the time almost exponentially (see Fig. 6.1), which makes the numerical algorithm fail at late time steps. One proposed solution is using $\eta = \ln \theta$ instead of θ to solve the problem.

Let $\eta = \ln \theta$. So $\theta = e^\eta$. Replacing θ in model (5.35) - (5.37) with e^η , we can rewrite the model as

$$\begin{aligned} \frac{\partial \eta}{\partial t} + \nabla \cdot (\eta \underline{V}) + (1 - \eta) \nabla \cdot \underline{V} &= 0, \quad x_0 < x < x_f, y_0 < y < y_f, \\ \nabla \cdot \left(\underline{V} - \frac{(1 - e^\eta)^3}{\varphi e^\eta} \nabla p \right) &= 0, \\ \nabla \cdot (\eta (\lambda (\nabla \cdot \underline{V}) \underline{I} + 2\mu \underline{E})) + (1 - \eta) \nabla \cdot (\lambda (\nabla \cdot \underline{V}) \underline{I} + 2\mu \underline{E}) \\ - e^{-\eta} \nabla p &= -\nabla(\eta \psi) - (1 - \eta) \nabla \psi + \frac{2\mu}{h^2} \underline{V}. \end{aligned}$$

Note that this may or may not work since $\eta_x = (\ln \theta)_x = \frac{\theta_x}{\theta} > \theta_x$ due to $0 < \theta < 1$. We did not pursue this line of work. It is desirable to find an efficient finite difference method to handle the problems with spikes in the domain.

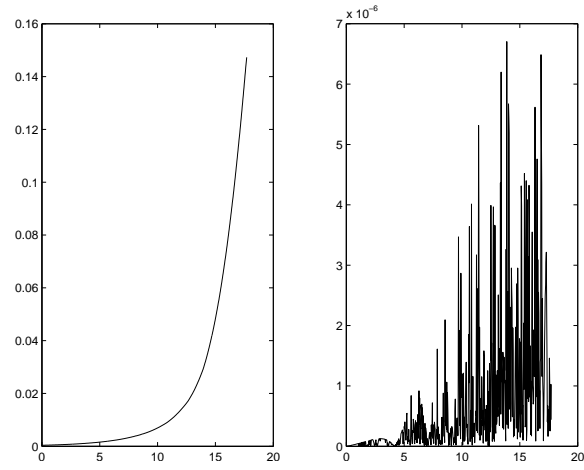


Figure 6.1: Maximum of minimum values of $\Delta\theta$ along with time 82×82 . The simulation crashes at $t = 17.7$. The plot on the left is a plot of $|\Delta\theta|_{max}$ vs time. The plot on the right is a plot of $|\Delta\theta|_{min}$ vs time. Here $\Delta\theta$ is the difference between adjacent grid points.

Many biological problems involving tissues can be modeled by mixtures of Stokes fluids. It is expected that our algorithm will be used for various mathematical biology problems.

Bibliography

- [1] J. Garvin, J. Qi, M. Maloney, and A. Banes. Novel system for engineering bioartificial tendons and application of mechanical load. *Tissue Engineering*, 9(5):967–979, 2003.
- [2] M. Dembo and F. Harlow. Cell motion, contractile networks, and the physics of interpenetrating reactive flow. *Biophysical Journal*, 50:109–121, 1986.
- [3] S. R. Lubkin and T. Jackson. Multiphase mechanics of capsule formation in tumors. *Journal of Biomechanical Engineering*, 124:1–7, 2002.
- [4] D. Manoussaki, S. R. Lubkin, R. B. Vernon, and J. D. Murray. A mechanical model for the formation of vascular networks in vitro. *Acta Biotheoretica*, 44:271–282, 1996.
- [5] D. E. Kenyon. Transient filtration in a porous elastic cylinder. *ASME Journal of Applied Mechanics*, 43:594–598, 1976.
- [6] V. C. Mow, S. C. Kuei, W. M. Lai, and Armstrong C. G. Biphasic creep and stress relaxation of articular cartilage in compression: theory and experiments. *Journal of Biomechanical Engineering*, 102(1):73–84, 1980.
- [7] W. M. Lai, J. S. Hou, and V. C. Mow. A triphasic theory for the swelling and deformation behaviors of articular cartilage. *Journal of Biomechanical Engineering*, 113(3):245–258, 1991.
- [8] Janssen J. D. Huyghe, J. M. Quadriphasic mechanics of swelling incompressible porous media. *International Journal of Engineering Science*, 35(8):793–802, 1997.
- [9] S. M. Klisch, S. S. Chen, R. L. Sah, and Hoger A. A growth mixture theory for cartilage with application to growth-related experiments on cartilage explants. *Journal of Biomechanical Engineering*, 125(2):169–179, 2003.

- [10] G. Lemon, J. R. King, H. M. Byrne, O. E. Jensen, and K. M. Shakesheff. Mathematical modelling of engineered tissue growth using a multiphase porous flow mixture theory. *Journal of Mathematical Biology*, 52(5):571–594, 2006.
- [11] M. Stastna. A moving boundary value problem in soft tissue mechanics. *Journal of Canadian Applied Mathematics Quarterly*, 13(2):183–198, 2005.
- [12] X. He and M. Dembo. Numerical simulation of oil-droplet cleavage by surfactant. *J. Biomechanical Engineering*, 118:201–209, 1996.
- [13] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.
- [14] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser-Verlag, 1990.
- [15] A. Harten. High resolution schemes for hyperbolic conservation laws. *J. Comput. Phys.*, 49:357–393, 1983.
- [16] B. van Leer. Towards the ultimate conservative difference scheme I. The quest of monotonicity. *Springer Lecture Notes in Physics*, 18:163–168, 1973.
- [17] B. van Leer. Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second order scheme. *J. Comput. Phys.*, 14:361–370, 1974.
- [18] B. van Leer. Towards the ultimate conservative difference scheme III. Upstream-centered finite-difference schemes for ideal compressible flow. *J. Comput. Phys.*, 23:263–275, 1977.
- [19] B. van Leer. Towards the ultimate conservative difference scheme IV. A new approach to numerical convection. *J. Comput. Phys.*, 23:276–299, 1977.
- [20] B. van Leer. Towards the ultimate conservative difference scheme V. A second order sequel to Godunov’s method. *J. Comput. Phys.*, 32:101–136, 1979.
- [21] T. Chan. Stability analysis of finite difference schemes for the advection-diffusion equation. *SIAM Journal of Numerical Analysis*, 21(2):272–284, 1984.
- [22] X.-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

- [23] X.-D. Liu and S. Osher. Convex ENO high order multi-dimensional schemes without field-by-field decomposition or staggered grids. *Journal of Computational Physics*, 142(2):304–330, 1998.
- [24] A. Harten. ENO schemes with subcell resolution. *J. Comput. Phys.*, 83:148–184, 1987.
- [25] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 71:231–303, 1987.
- [26] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Some results on uniformly high order accurate essentially non-oscillatory schemes. *Journal of Applied Numerical Mathematics*, 2:347–377, 1986.
- [27] G. Jiang and C. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202–228, 1996.
- [28] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes. *Journal of Computational Physics*, 77:439–471, 1988.
- [29] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes, II. *Journal of Computational Physics*, 83:32–78, 1989.
- [30] B. Cockburn, C. Johnson, C.-W. Shu, and E. Tadmor. *Lecture Notes in Mathematics - Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*. Springer, 1998.
- [31] B. Costa, W. S. Don, D. Gittlieb, and R. Sendersky. Two-dimensional multi-domain hybrid spectral-WENO methods for conservation laws. *Communications in Computational Physics*, 1(3):548–574, 2006.
- [32] G. Jiang and D. Peng. Weighted ENO schemes for Hamilton–Jacobi equations. *SIAM J. Sci. Comput.*, 21:2126–2143, 2000.
- [33] Y. Xing and C.-W. Shu. A new approach of high order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Communications in Computational Physics*, 1(1):100–134, 2006.

- [34] A. K. Henrick, T. D. Aslam, and J. M. Powers. Mapped weighted essentially non-oscillatory schemes: achieving optimal order near critical points. *Journal of Computational Physics*, 207(2):542–567, 2005.
- [35] S. Zhang and Shu C.-W. A new smoothness indicator for the WENO schemes and its effect on the convergence to steady state solutions. *Journal of Scientific Computing*, 31(112):273–305, 2007.
- [36] Z. Xu and C.-W. Shu. Anti-diffusive flux corrections for high order finite difference WENO schemes. *Journal of Computational Physics*, 205(2):458–485, 2005.
- [37] T. D. Aslam. A level-set algorithm for tracking discontinuities in hyperbolic conservation laws I: scalar equations. *Journal of Computational Physics*, 167(2):413–438, 2001.
- [38] Z. Li. *The Immersed Interface Method — A Numerical Approach for Partial Differential Equations with Interfaces*. PhD thesis, University of Washington, 1994.
- [39] Z. Li. Immersed interface method for moving interface problems. *Numerical Algorithms*, 14:269–293, 1997.
- [40] R. J. LeVeque and Z. Li. Immersed interface method for Stokes flow with elastic boundaries or surface tension. *SIAM J. Sci. Comput.*, 18:709–735, 1997.
- [41] Z. Li. An overview of the immersed interface method and its applications. *Taiwanese J. Mathematics*, 7:1–49, 2003.
- [42] K. Ito, Y. Kyei, and Z. Li. Higher-order, Cartesian grid based finite difference schemes for elliptic equations in irregular domains. *SIAM Journal on Scientific Computing*, 27:346–367, 2005.
- [43] L. M. Adams and Z. Li. The immersed interface/multigrid methods for interface problems. *SIAM J. Sci. Comput.*, 24:463–479, 2002.
- [44] A. Wiegmann, Z. Li, and R. LeVeque. Crack jump conditions for elliptic problems. *Applied Math. Letters*, 12:81–88, 1999.
- [45] S. Deng, K. Ito, and Z. Li. Three dimensional elliptic solvers for interface problems and applications. *J. Comput. Phys.*, 184:215–243, 2003.

- [46] R. J. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31:1019–1044, 1994.
- [47] Z. Li. A note on immersed interface methods for three dimensional elliptic equations. *Computers Math. Appl.*, 31:9–17, 1996.
- [48] Z. Li and K. Ito. *The Immersed Interface Method - Numerical Solutions of PDEs Involving Interfaces and Irregular Domains*. SIAM Frontier Series in Applied Mathematics, FR33, 2006.
- [49] J. B. Bell, C. N. Dawson, and G. R. Shubin. An unsplit, higher order Godunov method for scalar conservation laws in multiple dimensions. *J. Comput. Phys.*, 74:1–24, 1988.
- [50] G. R. Shubin and J. B. Bell. An analysis of the grid orientation effect in numerical simulation of miscible displacement. *Comp. Meth. Appl. Mech. Eng.*, 47:47–71, 1984.
- [51] A. N. Tikhonov and A. A. Samarskii. Homogeneous difference schemes. *USSR Comput. Math. and Math. Phys.*, 1:5–67, 1962.
- [52] C. S. Peskin. Numerical analysis of blood flow in the heart. *J. Comput. Phys.*, 25:220–252, 1977.
- [53] C. S. Peskin. Lectures on mathematical aspects of physiology. *Lectures in Appl. Math.*, 19:69–107, 1981.
- [54] C. S. Peskin and D. M. McQueen. A three-dimensional computational method for blood flow in the heart: (I) immersed elastic fibers in a viscous incompressible fluid. *J. Comput. Phys.*, 81:372–405, 1989.
- [55] C. S. Peskin and D. M. McQueen. A three-dimensional computational method for blood flow in the heart: (II) contractile fibers. *J. Comput. Phys.*, 82:289–297, 1989.
- [56] M-C. Lai. *Simulations of the flow past an array of circular cylinders as a test of the immersed boundary method*. PhD thesis, New York University, 1998.
- [57] R. P. Beyer. *A computational model of the cochlea using the immersed boundary method*. PhD thesis, University of Washington, 1989.

- [58] R. H. Dillon, L. J. Fauci, and A. L. Fogelson. Modeling biofilm processes using the immersed boundary method. *J. Comput. Phys.*, 129:57–73, 1996.
- [59] L. Zhu and C. Peskin. Simulation of a flapping flexible filament in a flowing soap film by the immersed boundary method. *J. Comput. Phys.*, 179:452–468, 2002.
- [60] D. Bottino. Modeling viscoelastic networks and cell deformation in the context of the immersed boundary method. *J. Comput. Phys.*, 147:86–113, 1998.
- [61] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the Ghost Fluid Method). *Journal of Computational Physics*, 152:457–492, 1999.
- [62] R. P. Fedkiw. Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the Ghost Fluid Method. *Journal of Computational Physics*, 175:200–224, 2002.
- [63] T. G. Liu, B. C. Khoo, and K. S. Yeo. Ghost fluid method for strong shock impacting on material interface. *Journal of Computational Physics*, 190:651–681, 2003.
- [64] T. G. Liu, B. C. Khoo, and C. W. Wang. Ghost fluid method for compressible gas-water simulation. *Journal of Computational Physics*, 204:193–221, 2005.
- [65] J. Glimm, X.-L. Li, Y.-J. Liu, Z. Xu, and N. Zhao. Conservative front tracking with improved accuracy. *SIAM Journal of Numerical Analysis*, 41(5):1926–1947, 2003.
- [66] J. Glimm, M. J. Graham, J. Grove, X.L. Lin, T. M. Smith, D. Tan, F. Tangerman, and Q. Zhang. Front tracking in two and three dimensions,. *J. Comp. Math.*, 7:1–12, 1998.
- [67] J. Glimm, J. W. Grove, and X. Li. Three-dimensional front tracking. *SIAM J. Sci. Comput.*, 19:703–727, 1999.
- [68] T. D. Aslam. A level set algorithm for tracking discontinuities in hyperbolic conservation laws II: system of equations. *Journal of Scientific Computing*, 19(1).
- [69] Z. Li and A. Mayo. ADI methods for heat equations with discontinuities along an arbitrary interface. In *Proc. Symp. Appl. Math.* W. Gautschi, editor, volume 48, pages 311–315. AMS, 1993.

- [70] A. Mayo. On the rapid evaluation of heat potentials on general regions. IBM Technical report 14305, 1991.
- [71] Z. Li and S. R. Lubkin. Numerical analysis of interfacial two-dimensional Stokes flow with discontinuous viscosity and variable surface tension. *International Journal of Numerical Methods in Fluids*, 37(5):525–540, 2001.
- [72] Z. Li, S. R. Lubkin, and X. Wan. An augmented IIM-level set method for Stokes equations with discontinuous viscosity. *Electronic Journal of Differential Equations*, (15):193–210, 2007.
- [73] Z. Li, S. R. Lubkin, and X. Wan. An augmented approach for pressure boundary condition in a stationary Stokes flow. *Communications in Computational Physics*, 1(5):874–885, 2006.
- [74] C. Zhang. *Immersed Interface Method for hyperbolic System of PDEs with Discontinuous Coefficients*. PhD thesis, University of Washington, 1996.
- [75] L. Lee and R. J. LeVeque. An immersed interface method for incompressible Navier Stokes equations. *SIAM J. Sci. Comput.*, 25:832–856, 2003.
- [76] L. Lee. *Immersed Interface Methods for Incompressible Flow with Moving Interfaces*. PhD thesis, University of Washington, 2002.
- [77] G. Chen. *Immersed interface method for biharmonic equations defined on irregular domains and its application to the Stokes flow*. PhD thesis, 2003.
- [78] X. Yang. *Immersed Interface Method for Elasticity Problems with Interfaces*. PhD thesis, North Carolina State University, 2004.
- [79] Y. Gong. *Immersed-Interface Finite-Element Methods for Elliptic and Elasticity Interface Problems*. PhD thesis, North Carolina State University, 2007.
- [80] D. Gottlieb, S. A. Orszag, and E. Turkel. Stability of pseudospectral and finite-difference methods for variable coefficient problems. *Journal of Mathematics of Computation*, 37(156):293–305, 1981.

- [81] P. W. White. Finite-difference methods in numerical weather prediction. In *Proceedings of Royal Society of London. Series A, Mathematical and Physical Sciences*, volume 323, pages 285–292. The Royal Society, 1971.
- [82] C.-W. Shu. Total-variation-diminishing time discretizations. *SIAM J. Sci. Stat. Comput.*, 9:1073–1084, 1988.
- [83] S. Gottlieb and C.-W. Shu. Total variation diminishing Runge-Kutta schemes. *Mathematics of Computation*, 67(221):73–85, 1998.
- [84] S. Gottlieb. On high order strong stability preserving Runge-Kutta and multi step time discretizations. *Journal of Scientific Computing*, 25(1).
- [85] E. F. Toro, R. C. Millington, and L. A. M. Nejad. Towards very high order Godunov schemes. In E. F. Toro, editor, *Godunov Methods Theory and Applications*. Kluwer Academic/Plenum Publishers, New York, New York, New York, 2001.
- [86] R. W. MacCormack. The effects of viscosity in hypervelocity impact cratering. AIAA Paper 69-354, 1969.
- [87] F.M. White. *Viscous Fluid Flow*. McGraw-Hill, Inc., 1991.
- [88] V.N. Constantinescu. *Laminar Viscous Flow*. Springer, 1995.
- [89] J.R. Ockendon H. Ockendon. *Viscous Flow*. Cambridge University Press, 1995.
- [90] R. J. LeVeque. Clawpack and Amrclaw – Software for high-resolution Godunov methods. 4-th Intl. Conf. on Wave Propagation, Golden, Colorado, 1998.
- [91] L. M. Adams and T. P. Chartier. New geometric immersed interface multigrid solvers. *SIAM J. Sci. Comput.*, 25:1516–1533, 2004.
- [92] L. M. Adams and T. P. Chartier. A comparison of algebraic multigrid and geometric immersed interface multigrid methods for interface problems. *SIAM J. Sci. Comput.*, 26:762–784, 2005.

Appendix

Appendix A

Accuracy of MacCormack's Method for a Fluid Mixture Model of Tissue Deformations

MacCormack's method is known as a two-step approach with second order accuracy. The problem with MacCormack's method is that it typically produces spurious oscillations. For a linear system, MacCormack's method is equivalent to the Lax-Wendroff method.

For a conservation law, $q_x + f(q)_x = 0$, applying MacCormack's method gives

$$\begin{aligned} Q_i^* &= Q_i^n - \nu(f(Q_{i+1}^n) - f(Q_i^n)), \\ Q_i^{**} &= Q_i^* - \nu(f(Q_i^*) - f(Q_{i-1}^*)), \\ Q_i^{n+1} &= \frac{1}{2}(Q_i^n + Q_i^{**}), \end{aligned} \tag{.1}$$

where $\nu = \frac{k}{h}$, in which k denotes the time step size whereas h the space mesh size. Consider a fluid mixture model of tissue deformations with constant physical parameters

$$\begin{aligned} \theta_t + (v\theta)_x &= 0, \\ v_{xx} + (\psi\theta)_x + (\sigma \ln(1 - \theta))_x &= 0. \end{aligned} \tag{.2}$$

Note that v implicitly depends on time and θ . Applying MacCormack's method to the hyperbolic equation in (.2) gives

$$\begin{aligned} \Theta_i^* &= \Theta_i^n - \nu((\Theta V)_{i+1}^n - (\Theta V)_i^n), \\ \Theta_i^{**} &= \Theta_i^* - \nu((\Theta V)_i^* - (\Theta V)_{i-1}^*), \\ \Theta_i^{n+1} &= \frac{1}{2}(\Theta_i^n + \Theta_i^{**}). \end{aligned} \tag{.3}$$

As a two-step approach applies to the hyperbolic equation in (.2), we need to compute v at time t^* since it is needed in the second step's calculation.

Claim: MacCormack's method is of first order accuracy provided that v is not updated at the first step when this approach is applied to the hyperbolic equation in (.2).

Proof:

At the first step, MacCormack's method gives

$$\begin{aligned}\theta(x_i, t^*) &= \theta(x_i, t^n) - \nu((\theta v)(x_{i+1}, t^n) - (\theta v)(x_i, t^n)), \\ \theta(x_{i-1}, t^*) &= \theta(x_{i-1}, t^n) - \nu((\theta v)(x_i, t^n) - (\theta v)(x_{i-1}, t^n)).\end{aligned}$$

Applying the Taylor series expansion at x_i and t^n gives

$$\begin{aligned}\theta(x_i, t^{n+1}) &= \theta(x_i, t^n) + \theta_t(x_i, t^n)k + \frac{k^2}{2}\theta_{tt}(x_i, t^n) + O(k^3), \\ \theta(x_{i-1}, t^n) &= \theta(x_i, t^n) - \theta_x(x_i, t^n)h + \frac{h^2}{2}\theta_{xx}(x_i, t^n) + O(h^3), \\ v(x_{i-1}, t^n) &= v(x_i, t^n) - v_x(x_i, t^n)h + \frac{h^2}{2}v_{xx}(x_i, t^n) + O(h^3), \\ (\theta v)(x_{i+1}, t^n) &= (\theta v)(x_i, t^n) + (\theta v)_x(x_i, t^n)h + \frac{h^2}{2}(\theta v)_{xx}(x_i, t^n) + O(h^3), \\ (\theta v)(x_{i-1}, t^n) &= (\theta v)(x_i, t^n) - (\theta v)_x(x_i, t^n)h + \frac{h^2}{2}(\theta v)_{xx}(x_i, t^n) + O(h^3).\end{aligned}$$

Suppose that $k = ch$, where $|c| < 1$. Then $O(k^n)$ is equivalent to $O(h^n)$. So

$$\begin{aligned}\theta(x_i, t^*) &= \theta(x_i, t^n) - \nu((\theta v)_x(x_i, t^n)h + \frac{h^2}{2}(\theta v)_{xx}(x_i, t^n) + O(h^3)), \\ \theta(x_{i-1}, t^*) &= \theta(x_i, t^n) - \theta_x(x_i, t^n)h + \frac{h^2}{2}\theta_{xx}(x_i, t^n) + O(h^3) \\ &\quad - \nu((\theta v)_x(x_i, t^n)h - \frac{h^2}{2}(\theta v)_{xx}(x_i, t^n)).\end{aligned}$$

Hence at the second step using the v_i^n gives

$$\begin{aligned}\theta(x_i, t^{**}) &= \theta(x_i, t^n) - \nu((\theta v)_x(x_i, t^n)h + \frac{h^2}{2}(\theta v)_{xx}(x_i, t^n) + O(h^3)) \\ &\quad - \nu((\theta(x_i, t^n) - \nu((\theta v)_x(x_i, t^n)h + \frac{h^2}{2}(\theta v)_{xx}(x_i, t^n)))v(x_i, t^n) \\ &\quad - (\theta(x_i, t^n) - \theta_x(x_i, t^n)h + \frac{h^2}{2}\theta_{xx}(x_i, t^n) \\ &\quad - \nu((\theta v)_x(x_i, t^n)h - \frac{h^2}{2}(\theta v)_{xx}(x_i, t^n)) \\ &\quad (v(x_i, t^n) - v_x(x_i, t^n)h + \frac{h^2}{2}v_{xx}(x_i, t^n))))), \\ &= \theta(x_i, t^n) - 2\nu h(\theta v)_x(x_i, t^n) + \nu k h((\theta v)_x v)_x(x_i, t^n) + O(h^3).\end{aligned}$$

So the local truncation error for MacCormack's method is

$$\begin{aligned}
& \frac{\theta(x_i, t^{n+1}) - \frac{1}{2}(\theta(x_i, t^n) + \theta(x_i, t^{**}))}{k} \\
&= \frac{1}{k}(\theta(x_i, t^n) + \theta_t(x_i, t^n)k + \frac{k^2}{2}\theta_{tt}(x_i, t^n) \\
&\quad - \frac{1}{2}(2\theta(x_i, t^n) - 2\nu h(\theta v)_x(x_i, t^n) + \nu k h((\theta v)_x v)_x(x_i, t^n) + O(h^3))) \quad (.4) \\
&= \frac{k}{2}\theta_{tt}(x_i, t^n) - \frac{\nu h}{2}((\theta v)_x v)_x(x_i, t^n) + O(h^2) \\
&= -\frac{k}{2}(\theta v_t)_x(x_i, t^n) + O(h^2).
\end{aligned}$$

We can see that the dominant local truncation error is of $O(k)$. So MacCormack's method is of first order accuracy when it is applied to the system (.2) without updating the v after the first step and before the second step.