

NUMERICAL SIMULATIONS OF STEADY AND UNSTEADY OBLIQUE
DETONATION PHENOMENA WITH APPLICATION TO PROPULSION

A Dissertation

Submitted to the Graduate School
of the University of Notre Dame
In Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

by

Matthew John Grismer, B.S., M.S.

Joseph M. Powers, Director

Department of Aerospace and Mechanical Engineering

Notre Dame, Indiana

November, 1994

NUMERICAL SIMULATIONS OF STEADY AND UNSTEADY OBLIQUE DETONATION PHENOMENA WITH APPLICATION TO PROPULSION

Abstract

by

Matthew John Grismer

Oblique detonations and their propulsive applications were studied by numerically integrating the two-dimensional, reactive Euler equations in a generalized, curvilinear coordinate system. The integration was accomplished using the Roe scheme combined with fractional stepping; nonlinear flux limiting was used to prevent unphysical solution oscillations near discontinuities. The method was verified on one- and two-dimensional flows with exact solutions, and its ability to correctly predict one-dimensional detonation instability was demonstrated.

Unsteady phenomena were considered in a study of straight oblique detonations attached to curved walls. Using the exact, steady oblique detonation solution as an initial condition, the numerical simulation predicted both steady and unsteady oblique detonation solutions when a detonation parameter known as the normal overdrive was varied. The normal overdrive value at which the oblique detonation transitioned from steady to unsteady behavior was slightly higher than the corresponding value for a one-dimensional detonation.

An oblique detonation application was considered to determine the steady propagation speed of an idealized ram accelerator. Propagation speeds were found which gave rise to shocks of such strength as to induce a reaction zone to be in a region which allowed the combustion induced thrust to balance the wave drag. For fixed heat

release greater than a critical value, two steady propagation speeds were predicted. The solution at the higher Mach number was stable to static perturbations while the solution at the lower Mach number was unstable. The Chapman-Jouguet velocity in the direction of projectile propagation was found to be relevant only as an overly conservative lower bound for possible flight speeds. In the far-field the detonation wave angle was found to be that of a Chapman-Jouguet oblique detonation.

Dedication

This dissertation is dedicated to my parents, who instilled in me the desire to learn, and to my wife, with whom I have shared this graduate experience.

Contents

LIST OF FIGURES	v
LIST OF SYMBOLS	xi
ACKNOWLEDGEMENTS	xviii
1 INTRODUCTION	1
2 LITERATURE REVIEW	7
2.1 One-dimensional Detonations	8
2.2 Oblique Detonations	10
2.3 Ram Accelerators	13
3 MODEL EQUATIONS	16
3.1 Cartesian Reactive Euler Equations	16
3.2 Generalized Curvilinear Reactive Euler Equations	22
4 FUNDAMENTAL DETONATION SOLUTIONS	31
4.1 One-dimensional Steady Solution	31
4.2 Two-dimensional Steady Solutions	39
4.2.1 Straight Shock, Curved Wall	39
4.2.2 Curved Shock, Straight Wall	46
5 NUMERICAL METHOD	53
5.1 Godunov Schemes	53
5.2 The Roe Scheme	56
5.3 Extension of the Roe Scheme to Second-Order Accuracy	69
5.4 The Roe Scheme for the Reactive Euler Equations in a Two-Dimensional, Curvilinear Coordinate System	76
6 VERIFICATION OF NUMERICAL ALGORITHM	84
6.1 Riemann Problem	84
6.2 One-dimensional Detonations	91
6.2.1 Unsteady Solutions	91
6.2.2 Steady Solutions	96
6.3 Inert Supersonic Flow over a Wedge	99
6.4 Straight Wall, Curved Oblique Detonation Problem	128

7	UNSTEADY OBLIQUE DETONATIONS	132
7.1	Normal Overdrive $f_n = 2.0$	134
7.2	Normal Overdrive $f_n = 1.6$	151
7.3	Normal Overdrive $f_n = 1.7$	161
7.4	Normal Overdrive $f_n = 1.76$	170
7.5	Normal Overdrive $f_n = 1.78$	176
7.6	Normal Overdrive $f_n = 1.8$	185
8	RAM ACCELERATOR CALCULATIONS	196
8.1	Methodology	196
8.2	Model Problem	197
8.3	Results	199
9	CONCLUSIONS	214
9.1	Unsteady Oblique Detonations	214
9.2	Steady Propagation of a Ram Accelerator	217
A	PROGRAM LISTING	220
A.1	roe2d.f	220
A.2	center.f	230
A.3	ghost.f	231
A.4	metrcc.f	235
A.5	metrci.f	236
A.6	metrcj.f	236
A.7	timest.f	237
A.8	flxxi.f	238
A.9	boundx.f	242
A.10	decode.f	243
A.11	flxeta.f	244
A.12	boundn.f	248
	LIST OF REFERENCES	252

List of Figures

1.1	Schematic of oblique detonation.	1
1.2	Simplistic concept for oblique detonation wave engine.	3
1.3	Schematic of ram accelerator flow field. [37]	4
2.1	Illustration of overdrive.	8
2.2	Unsteady detonations resulting from hypervelocity projectiles. [49]	11
2.3	Schematic of ram accelerator test facility.	13
3.1	Illustration of the transformation from Cartesian space to the generalized, curvilinear space.	23
4.1	Illustration of the steady detonation problem and the Galilean coordinate transformation.	31
4.2	Diagram of detonation in the \check{x} coordinate system.	33
4.3	Rayleigh lines and Hugoniot curves for ZND detonation.	35
4.4	Overdriven ZND detonation solution ($f = 1.8, q = 0.719, \Theta = 0.719, \gamma = 1.2$).	38
4.5	Chapman-Jouguet detonation solution ($f = 1.0, q = 0.719, \Theta = 0.719, \gamma = 1.2$).	39
4.6	Schematic of straight shock, curved wall detonation flowfield.	40
4.7	Variation of oblique detonation wave angle with wedge angle and heat release ($\gamma = 1.2, \mathcal{M}_0 = 5$).	43
4.8	Weak, overdriven straight shock, curved wall detonation solutions for pressure and product mass fraction ($f_n = 1.6, \mathcal{M}_0 = 9, q = 0.719, \Theta = 0.719, \gamma = 1.2$).	44
4.9	Strong straight shock, curved wall detonation solutions for pressure and product mass fraction ($f_n = 1.6, \mathcal{M}_0 = 8, q = 0.719, \Theta = 0.719, \gamma = 1.2$).	45
4.10	Oblique detonation vorticity contours from asymptotic solution ($\mathcal{M}_0 = 20, \bar{q} = 10, \gamma = 1.4, \theta = 20^\circ$).	50
4.11	Oblique detonation product mass fraction contours from asymptotic solution ($\mathcal{M}_0 = 20, \bar{q} = 10, \gamma = 1.4, \theta = 20^\circ$).	50
4.12	Oblique detonation pressure contours from asymptotic solution ($\mathcal{M}_0 = 20, \bar{q} = 10, \gamma = 1.4, \theta = 20^\circ$).	51
4.13	Oblique detonation pressure contours from numerical solution ($\mathcal{M}_0 = 20, \bar{q} = 10, \gamma = 1.4, \theta = 20^\circ$).	51

4.14	Comparison of exact, asymptotic and numerical solutions using average pressure difference e_1	52
5.1	$x-t$ diagram of the Riemann problem.	54
5.2	Exact solution to Riemann problem at $\tilde{t} = 0.0061$ s (diaphragm initially at $\tilde{x} = 0$, initial pressure ratio = 10, initial density ratio = 8, initial velocity = 0, $\gamma = 1.4$, total domain length = 10 m).	55
5.3	Illustration of Godunov's method.	56
5.4	Diagram of linear Riemann solution.	61
5.5	Linear Riemann solution by quadrant.	62
5.6	Illustration of one-dimensional travelling shock.	64
5.7	Linear extrapolation for $s_{i+1/2}$	70
5.8	Types of conditions required for the computational domain boundaries.	78
5.9	Schematic of wall boundary computational cell.	82
6.1	Comparison of Roe scheme solutions with the exact density solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).	85
6.2	Comparison of Roe scheme solutions with the exact velocity solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).	86
6.3	Comparison of Roe scheme solutions with the exact pressure solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).	87
6.4	Comparison of Roe scheme solutions with the exact density solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 100$, $\tilde{\rho}_l/\tilde{\rho}_r = 100$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0039$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).	88
6.5	Comparison of 2nd order Roe scheme solutions using different limiters with the exact density solution for the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).	89
6.6	Enlarged view of the contact and shock discontinuities in the comparison of the 2nd order Roe scheme solutions. ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m)	90
6.7	Measured order of accuracy for the first- and second-order Roe schemes using the exact Riemann solution as a basis.	90
6.8	Unsteady detonation peak pressures ($f = 1.6$, CFL = 0.4, 14 points $\tilde{L}_{1/2}$).	92
6.9	Unsteady detonation peak pressures ($f = 1.6$, CFL = 0.4, 27 points $\tilde{L}_{1/2}$).	93
6.10	Unsteady detonation peak pressures ($f = 1.6$, CFL = 0.35, 27 points $\tilde{L}_{1/2}$).	94
6.11	Unsteady detonation peak pressures ($f = 1.72$, CFL = 0.4, 5 points $\tilde{L}_{1/2}$).	95
6.12	Unsteady detonation peak pressures ($f = 1.72$, CFL = 0.4, 10 points $\tilde{L}_{1/2}$).	95
6.13	Steady detonation peak pressures ($f = 1.8$, CFL = 0.4).	96

6.14	Steady detonation peak pressures ($f = 1.74$, CFL = 0.4).	97
6.15	Measured order of accuracy for the second-order Roe scheme using the exact ZND solution as a basis ($f = 1.8$).	98
6.16	Computational grid of cell centers used for 20° wedge.	99
6.17	Pressure contours and velocity vectors for $\mathcal{M}_0 = 10$	102
6.18	Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 10$	103
6.19	Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 10$	104
6.20	Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 10$	105
6.21	Residual histories of density and u component of velocity for $\mathcal{M}_0 = 10$	106
6.22	Pressure contours and velocity vectors for $\mathcal{M}_0 = 2$ and $K = 0.0$	109
6.23	Pressure contours and velocity vectors for $\mathcal{M}_0 = 2$ and $K = 0.025$	110
6.24	Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$	111
6.25	Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$	112
6.26	Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$	113
6.27	Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$	114
6.28	Pressure contours and velocity vectors for $\mathcal{M}_0 = 2$ and $K = 0.035$	116
6.29	Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$	117
6.30	Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$	118
6.31	Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$	119
6.32	Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$	120
6.33	Pressure contours and velocity vectors for $\mathcal{M}_0 = 4$ and $K = 0.025$	122
6.34	Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$	123
6.35	Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$	124
6.36	Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$	125
6.37	Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$	126
6.38	Measured order of accuracy for the second-order Roe scheme in two dimensions using the exact oblique shock solution as a basis ($\mathcal{M}_0 = 4$ and $K = 0.025$).	127
6.39	Oblique detonation pressure contours from Roe scheme numerical solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).	130

6.40	Oblique detonation pressure contours from asymptotic solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).	130
6.41	Comparison of surface pressures for numerical and asymptotic oblique detonation solutions, and numerical and exact oblique shock solutions ($\mathcal{M}_0 = 20$, $\gamma = 1.4$, $\theta = 20^\circ$).	131
6.42	Comparison of shock locations for numerical and asymptotic oblique detonation solutions ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).	131
7.1	Typical grid of cell centers used in oblique detonation studies.	133
7.2	Shaded density contours as a function of time ($f_n = 2.0$, 99 by 99 grid, $t = 409.8, 657.5, 1111.6, 2066.9, 2605.0, 3687.0$).	138
7.3	Shaded pressure contours as a function of time ($f_n = 2.0$, 99 by 99 grid, $t = 409.8, 657.5, 1111.6, 2066.9, 2605.0, 3687.0$).	139
7.4	Shaded product mass fraction contours as a function of time ($f_n = 2.0$, 99 by 99 grid, $t = 409.8, 657.5, 1111.6, 2066.9, 2605.0, 3687.0$).	140
7.5	Time histories of density and u component of velocity residuals ($f_n = 2.0$, 149 by 149 grid).	141
7.6	Shaded density contours as a function of time ($f_n = 2.0$, 149 by 149 grid, $t = 246.8, 645.2, 845.0, 969.9, 1094.8, 1869.6$).	142
7.7	Shaded pressure contours as a function of time ($f_n = 2.0$, 149 by 149 grid, $t = 246.8, 645.2, 845.0, 969.9, 1094.8, 1869.6$).	143
7.8	Shaded product mass fraction contours as a function of time ($f_n = 2.0$, 149 by 149 grid, $t = 246.8, 645.2, 845.0, 969.9, 1094.8, 1869.6$).	144
7.9	Velocity vectors for final, steady solution ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).	145
7.10	Comparison of numerical and exact density contours ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$, dashed = numerical, line = exact).	145
7.11	Comparison of numerical and exact density solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).	146
7.12	Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).	147
7.13	Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).	148
7.14	Comparison of numerical and exact pressure solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).	149
7.15	Comparison of numerical and exact product mass fraction solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).	150
7.16	Shaded density and pressures contours for steady numerical solution ($f_n = 1.6$, 99 by 99 grid, $t = 3128.2$).	155
7.17	Shaded density contours as a function of time ($f_n = 1.6$, 139 by 184 grid, $t = 1176.3, 1929.1, 2776.0, 3481.7, 4234.5, 5175.5$).	156
7.18	Shaded pressure contours as a function of time ($f_n = 1.6$, 139 by 184 grid, $t = 1176.3, 1929.1, 2776.0, 3481.7, 4234.5, 5175.5$).	157
7.19	Shaded product mass fraction contours as a function of time ($f_n = 1.6$, 139 by 184 grid, $t = 1176.3, 1929.1, 2776.0, 3481.7, 4234.5, 5175.5$).	158
7.20	Illustration of three-shock structure [22].	159

7.21	Comparison of progression of density solutions on different grids (left - 99 by 124 grid, right - 139 by 184 grid).	160
7.22	Shaded density contours as a function of time ($f_n = 1.7$, 159 by 184 grid, $t = 960.5, 1870.6, 2831.1, 3791.7, 7077.8, 9251.7$).	163
7.23	Shaded pressure contours as a function of time ($f_n = 1.7$, 159 by 184 grid, $t = 960.5, 1870.6, 2831.1, 3791.7, 7077.8, 9251.7$).	164
7.24	Shaded product mass fraction contours as a function of time ($f_n = 1.7$, 159 by 184 grid, $t = 960.5, 1870.6, 2831.1, 3791.7, 7077.8, 9251.7$).	165
7.25	Shaded density contours as a function of time for bursting events ($f_n = 1.7$, 159 by 184 grid, $t = 2527.8, 2831.1, 3336.7, 3741.1, 3892.8, 4145.6$).	166
7.26	Shaded pressure contours as a function of time for bursting events ($f_n = 1.7$, 159 by 184 grid, $t = 2527.8, 2831.1, 3336.7, 3741.1, 3892.8, 4145.6$).	167
7.27	Shaded product mass fraction contours as a function of time for bursting events ($f_n = 1.7$, 159 by 184 grid, $t = 2527.8, 2831.1, 3336.7, 3741.1, 3892.8, 4145.6$).	168
7.28	Shaded density contours and velocity vectors within the bursting regions ($f_n = 1.7$, 159 by 184 grid, $t = 2831.1$ and 3892.8).	169
7.29	Shaded density contours as a function of time ($f_n = 1.76$, CFL = 0.35, 184 by 209 grid, $t = 754.5, 1349.9, 2660.1, 4168.8, 7146.5, 8655.2$).	172
7.30	Shaded pressure contours as a function of time ($f_n = 1.76$, CFL = 0.35, 184 by 209 grid, $t = 754.5, 1349.9, 2660.1, 4168.8, 7146.5, 8655.2$).	173
7.31	Shaded product mass fraction contours as a function of time ($f_n = 1.76$, CFL = 0.35, 184 by 209 grid, $t = 754.5, 1349.9, 2660.1, 4168.8, 7146.5, 8655.2$).	174
7.32	Comparison of progression of density solutions on identical grids for different time step criteria (left - CFL = 0.4, right - CFL = 0.35).	175
7.33	Shaded density contours as a function of time ($f_n = 1.78$, 184 by 209 grid, $t = 290.9, 756.3, 1105.4, 2181.7, 2647.2, 3839.9$).	178
7.34	Shaded pressure contours as a function of time ($f_n = 1.78$, 184 by 209 grid, $t = 290.9, 756.3, 1105.4, 2181.7, 2647.2, 3839.9$).	179
7.35	Shaded product mass fraction contours as a function of time ($f_n = 1.78$, 184 by 209 grid, $t = 290.9, 756.3, 1105.4, 2181.7, 2647.2, 3839.9$).	180
7.36	Comparison of numerical and exact density contours ($f_n = 1.78$, 184 by 209 grid, $t = 3839.9$, dashed = numerical, line = exact).	181
7.37	Comparison of numerical and exact density solutions along $j =$ constant grid lines ($f_n = 1.78$, 184 by 209 grid, $t = 3839.9$).	182
7.38	Comparison of numerical and exact pressure solutions along $j =$ constant grid lines ($f_n = 1.78$, 184 by 209 grid, $t = 3839.9$).	183
7.39	Comparison of numerical and exact product mass fraction solutions along $j =$ constant grid lines ($f_n = 1.78$, 184 by 209 grid, $t = 3839.9$).	184
7.40	Shaded density contours as a function of time ($f_n = 1.8$, CFL = 0.35, 184 by 209 grid, $t = 253.1, 810.1, 1316.6, 1924.2, 2658.5, 4557.5$).	187
7.41	Shaded pressure contours as a function of time ($f_n = 1.8$, CFL = 0.35, 184 by 209 grid, $t = 253.1, 810.1, 1316.6, 1924.2, 2658.5, 4557.5$).	188

7.42	Shaded product mass fraction contours as a function of time ($f_n = 1.8$, CFL = 0.35, 184 by 209 grid, $t = 253.1, 810.1, 1316.6, 1924.2, 2658.5, 4557.5$).	189
7.43	Comparison of numerical and exact density contours ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$, dashed = numerical, line = exact).	190
7.44	Comparison of numerical and exact density solutions along $j =$ constant grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).	191
7.45	Comparison of numerical and exact u component of velocity solutions along $j =$ constant grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).	192
7.46	Comparison of numerical and exact v component of velocity solutions along $j =$ constant grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).	193
7.47	Comparison of numerical and exact pressure solutions along $j =$ constant grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).	194
7.48	Comparison of numerical and exact product mass fraction solutions along $j =$ constant grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).	195
8.1	Schematic of generic configuration	198
8.2	Detailed schematic for $\tilde{H}_c \rightarrow \infty$	198
8.3	Net thrust force versus Mach number for varying heat release.	201
8.4	Bifurcation diagram for steady state speed versus heat of reaction.	202
8.5	Pressure contours for statically stable steady configuration.	203
8.6	Product mass fraction contours for statically stable configuration.	204
8.7	Pressure contours for statically unstable steady configuration.	205
8.8	Product mass fraction contours for statically unstable steady configuration.	206
8.9	Pressure traces on wedge surface.	207
8.10	Sixth-order polynomial curve fit of thrust curve for $\tilde{q} = 1.014 MJ/kg$	208
8.11	Projectile acceleration as a function of time ($\tilde{q} = 1.014 MJ/kg$, $\tilde{m} = 1.2108 kg/m$).	209
8.12	Projectile acceleration as a function of distance ($\tilde{q} = 1.014 MJ/kg$, $\tilde{m} = 1.2108 kg/m$).	210
8.13	Projectile velocity as a function of time ($\tilde{q} = 1.014 MJ/kg$, $\tilde{m} = 1.2108 kg/m$).	210
8.14	Projectile velocity as a function of position ($\tilde{q} = 1.014 MJ/kg$, $\tilde{m} = 1.2108 kg/m$).	211
8.15	Projectile position as a function of time ($\tilde{q} = 1.014 MJ/kg$, $\tilde{m} = 1.2108 kg/m$).	211
8.16	Comparison of exact and predicted detonation wave angles for steady speeds of propagation.	212
8.17	Exact and predicted detonation wave angles as a function of heat release ($\mathcal{M}_0 = 8.4$).	212
8.18	Comparison of exact CJ detonation pressure solution with far-field numerical pressure solution along a line normal to the oblique detonation wave ($\mathcal{M}_0 = 8.4$, $\tilde{q} = 0.985 MJ/kg$).	213

LIST OF SYMBOLS

$\tilde{\bullet}$	Accent denoting dimensional quantities
$\hat{\bullet}$	Accent denoting quantities in generalized coordinate system
$\check{\bullet}$	Accent denoting quantities in detonation frame of reference
$\bar{\bullet}$	Accent denoting Roe averaged quantities
\bullet_0	Subscript denoting freestream quantities
\bullet_m	Subscript denoting m^{th} component of vector or m^{th} column of matrix
$\bullet_{min}, \bullet_{max}$	Subscript denoting minimum and maximum values of quantities
\bullet_l, \bullet_r	Subscript denoting quantities in the left and right states
a_p	Projectile acceleration
\mathbf{b}	Source vector for primitive variable formulation of model equations
c	Isentropic frozen speed of sound
$c_{p,i}$	Specific heat at constant pressure for i^{th} species
c_p	Specific heat at constant pressure of species 1 and 2
c_v	Specific heat at constant volume of species 1 and 2
$d\hat{\mathbf{f}}_m$	Jump in $\hat{\mathbf{f}}$ across the m^{th} characteristic
$d\hat{\mathbf{f}}_{i-1/2}^{m+}$	Jump in $\hat{\mathbf{f}}$ across the m^{th} positive characteristic at $\xi_{i-1/2}$
$d\hat{\mathbf{f}}_{i+3/2}^{m-}$	Jump in $\hat{\mathbf{f}}$ across the m^{th} negative characteristic at $\xi_{i+3/2}$
$d\check{\mathbf{f}}_{i-1/2}^{m+}$	Limited form of $d\hat{\mathbf{f}}_{i-1/2}^{m+}$
$d\check{\mathbf{f}}_{i+3/2}^{m-}$	Limited form of $d\hat{\mathbf{f}}_{i+3/2}^{m-}$
ds	Element of arc length

e	Specific internal energy
e_1	Fractional error
f	Overdrive factor
f_n	Normal overdrive factor
\mathbf{f}, \mathbf{g}	Cartesian conserved flux vectors
$\hat{\mathbf{f}}_{i+1/2}^n$ or $\hat{\mathbf{F}}(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n)$	Average flux at $\xi_{i+1/2}$ over temporal interval τ^n to τ^{n+1}
$\hat{\mathbf{f}}_i^-, \hat{\mathbf{f}}_i^+$	Portions of $\hat{\mathbf{f}}$ due to waves on negative/positive characteristics
$\hat{\mathbf{f}}_{i+1/2}^{+b}$	Linearly extrapolated fluxes at $\xi_{i+1/2}$ due to $\hat{\mathbf{f}}_i^+$ and $\hat{\mathbf{f}}_{i-1}^+$
$\hat{\mathbf{f}}_{i+1/2}^{-f}$	Linearly extrapolated fluxes at $\xi_{i+1/2}$ due to $\hat{\mathbf{f}}_{i+1}^-$ and $\hat{\mathbf{f}}_{i+2}^-$
$\bar{\bar{\mathbf{f}}}_{i+1/2}^n$	First-order Roe flux at $\xi_{i+1/2}$ and τ^n
$\bar{\bar{\mathbf{f}}}_{i+1/2}^{(1)}$	First-order numerical flux at $\xi_{i+1/2}$ and τ^n
$\bar{\bar{\mathbf{f}}}_{i+1/2}^{(2)n}$	Second-order Roe flux at $\xi_{i+1/2}$ and τ^n
$\bar{\bar{\mathbf{f}}}_{i+1/2}^{(2)\dagger}$	$\bar{\bar{\mathbf{f}}}_{i+1/2}^{(2)n+1/2}$
h_i	Specific enthalpy of i^{th} species
h_i^0	Standard enthalpy of formation per unit mass for i^{th} species at T^0
i, j	Grid point indices
$k_{i,j}$	Maximum absolute value of eigenvalue λ_m at cell i, j
$l_{i,j}$	Maximum absolute value of eigenvalue μ_m at cell i, j
$\ell(\xi, \tau)$	Maximum value of m such that $\xi - \bar{\bar{\lambda}}_m \tau \geq 0$
m	Projectile mass per depth
m_s	Slope of a line
$m_{i,j}$	$\sqrt{k_{i,j}^2 + l_{i,j}^2}$
n_i	Unit normal
p	Pressure
p_m, \mathbf{p}	Primitive variables and variable vector
p_∞	Far-field pressure

p_{a_i}	Asymptotic pressure solution at a point
p_{n_i}	Numerical pressure solution at a point
q	Heat release parameter
\bar{q}	Alternative nondimensional form of heat release parameter
q_m, \mathbf{q}	Cartesian conserved variables and variable vector
$\hat{\mathbf{q}}_0(\xi)$	$\hat{\mathbf{q}}(\xi, 0)$
$\hat{\mathbf{q}}_i^n, \hat{\mathbf{Q}}_i^n$	Average $\hat{\mathbf{q}}$ at time τ^n over spatial interval $\xi_{i-1/2}$ to $\xi_{i+1/2}$
$\hat{\mathbf{q}}_i^\dagger$	$\hat{\mathbf{q}}_i^{n+1/2}$
$\hat{\mathbf{q}}^*(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n)$	Value of $\hat{\mathbf{q}}$ at $\xi_{i+1/2}$ due to Riemann solution between cells i and $i + 1$
\mathbf{r}_m, \mathbf{R}	Right eigenvectors and right eigenvector matrix of $\hat{\mathbf{A}}$
s	Shock speed
s_i	Dummy variable at ξ_i
t	Time
$t_{1/2}$	Time required for a fluid particle to cross $L_{1/2}$
u, v	Cartesian velocities
u_s	Shocked velocity
u_{CJ}	End-state velocity of Chapman-Jouguet detonation
u_p	Projectile velocity
v_m, \mathbf{v}	Characteristic variable and variable vector
\mathbf{w}	Cartesian conserved source vector
x, y	Cartesian coordinates
y_w	Wall function
z_m, \mathbf{z}	Variables and variable vector for Roe averaging
\bar{z}_m	Components of arithmetic mean of \mathbf{z}_l and \mathbf{z}_r
\mathbf{A}, \mathbf{B}	Cartesian flux Jacobian matrices
B	Arrhenius prefactor

CFL	Courant-Friedrichs-Lewy number
C_1, C_2	Constants in ZND detonation solution
$\bar{\mathbf{C}}, \bar{\mathbf{D}}$	Average matrices in Roe averaging
CJ	Chapman-Jouguet
\mathbf{C}	Chapman-Jouguet state for ZND detonation solution
D	Detonation velocity
D_c	Contact discontinuity
D_{CJ}	Chapman-Jouguet detonation velocity
E	Total energy per unit mass
\hat{E}	Activation energy
\tilde{E}_a, Θ	Mass based activation energy (dimensional and nondimensional)
\bar{E}_a	Alternative nondimensional activation energy
F_{net}	Net force per depth
\mathbf{G}	$\partial \hat{\mathbf{f}} / \partial \mathbf{p}$
H	Total enthalpy per unit mass
H_c	Cowl surface separation distance
\mathbf{H}, \mathbf{K}	Flux Jacobian matrices for primitive variable model equations
\mathcal{H}	Hugoniot curve
\mathbf{I}	Identity matrix
J	Grid Jacobian
K	Constant in subsonic boundary condition estimation
L_p	Length of symmetric double wedge
$L[a, b]$	Nonlinear limiting function
$L_{1/2}$	Half reaction zone length
L_{sh}	Length of shock within computational domain
L_{ratio}	Ratio of L_{sh} to $L_{1/2}$
$\mathcal{L}_\xi^{\Delta\tau}, \mathcal{L}_\eta^{\Delta\tau}, \mathcal{L}_Y^{\Delta\tau}$	Solution operators from τ^n to τ^{n+1} for split conservation equations

M	Molecular weight of species 1 and 2
M_i	Molecular weight of the i^{th} species
\mathbf{M}	$\partial\hat{\mathbf{q}}/\partial\mathbf{p}$
\mathcal{M}	Mach number
\mathcal{M}_{0_n}	Component of \mathcal{M}_0 normal to detonation
\mathcal{M}_2	Mach number following detonation
\mathcal{M}_{2n}	Component of \mathcal{M}_2 normal to detonation
$\mathcal{M}_{0_{CJ}}$	Chapman-Jouguet Mach number
N	Total number of computational points
N_{sp}	Total number of species
\mathbf{N}	Neumann point (shocked state) in ZND detonation solution
NASP	National Aerospace Plane
\mathbf{O}	Initial unshocked state in ZND detonation solution
\mathbf{OD}	Overdriven
ODWE	Oblique detonation wave engine
\mathcal{P}	$\zeta\mathbf{Y}^{-1}\frac{\partial\mathbf{p}}{\partial\xi}$
R	Gas constant
\mathcal{R}	Rayleigh line
\mathfrak{R}	Universal gas constant
S_1, S_2	Intersecting shock fronts
S'	Reflected shock
\mathbf{S}	Strong solution
T	Temperature
\mathbf{T}	Left eigenvector matrix of $\hat{\mathbf{A}}$
T^0	Standard temperature
$\text{TV}(q)$	Total variation of q
\mathbf{U}	Right eigenvector matrix of $\hat{\mathbf{B}}$

UD	Underdriven
U^c, V^c	Contravariant velocities
V'_1	Velocity following reflected shock S'
V_2	Velocity following shock S_2
W	Weak solution
X	Right eigenvector matrix of K
Y	Right eigenvector matrix of H
Y_i	Mass fraction of i^{th} species
Y_1	Reactant mass fraction
Y_2	Product mass fraction
Z	$\partial \hat{\mathbf{g}} / \partial \mathbf{p}$
ZND	Zeldovich-von Neumann-Doering
α	Temperature exponent in Arrhenius kinetics
$\alpha_m, \boldsymbol{\alpha}$	Constants and constant matrix in solution of linear Riemann problem
β	Detonation wave angle
β_{CJ}	Chapman-Jouguet detonation wave angle
γ	Ratio of specific heats
ϵ	\mathcal{M}_0^{-2}
ε	Parameter in entropy correction to Roe scheme
$\zeta_m, \boldsymbol{\zeta}$	Eigenvalues and eigenvalue matrix of H
θ	Sharp-edged straight wedge angle
$\boldsymbol{\kappa}$	Eigenvalue matrix of K
$\lambda_m, \boldsymbol{\lambda}$	Eigenvalues and eigenvalue matrix of $\hat{\mathbf{A}}$
$ \bar{\lambda}_m _{\text{mod}}$	Entropy corrected absolute eigenvalue for Roe scheme
$\mu_m, \boldsymbol{\mu}$	Eigenvalues and eigenvalue matrix of $\hat{\mathbf{B}}$
ξ, η	Generalized, curvilinear coordinates

ξ_i	Spatial coordinate $i(\Delta\xi)$
$\xi_x, \xi_y, \xi_t, \eta_x,$ $\eta_y, \eta_t, x_\xi, x_\eta,$ $x_\tau, y_\xi, y_\eta, y_\tau$	Grid metrics
ρ	Density
$\sigma_{i+1/2}^{m\pm}$	$\alpha_m \bar{\lambda}_{m\pm}$ at $\xi_{i+1/2}$
$\check{\sigma}_{i-1/2}^{m+}$	$L[\sigma_{i-1/2}^{m+}, \sigma_{i+1/2}^{m+}]$
$\check{\sigma}_{i+3/2}^{m-}$	$L[\sigma_{i+3/2}^{m-}, \sigma_{i+1/2}^{m-}]$
τ	Generalized time
τ^n	Time at the n^{th} time step
φ	Specific volume
ψ	Parameter in Superbee limiter
ω_z	z component of vorticity
Γ	Large value in derivation of Rankine-Hugoniot jump relations
Υ	Generic variable

ACKNOWLEDGEMENTS

I would like to acknowledge the NASA Lewis Research Center, the Center for Applied Mathematics at the University of Notre Dame, and the Department of Aerospace and Mechanical Engineering at the University of Notre Dame for providing support for my research in the form of summer internships, fellowships, and assistantships, respectively. I would also like to thank the Computational Fluid Dynamics Branch in the Flight Dynamics Directorate of Wright Laboratory, Wright-Patterson Air Force Base, for their support while I completed this dissertation.

I would like to acknowledge my advisor, Dr. Joe Powers, for his guidance and instruction throughout my graduate career. He was always available to offer advice and prod my investigations in the right direction, but was never heavy-handed in his approach. He is also a friend, and made a good roommate for a memorable summer in Cleveland. I would like to thank him for making my graduate experience pleasant and rewarding. I would also like to acknowledge Drs. Paolucci, Sen, and Yang for their insightful suggestions, as well as taking the time to be on my doctoral committee.

I would like to acknowledge Mr. John R. Roof IV for his assistance with some of the numerical calculations. I would like to thank my officemates Tom Copps, Andy Smith, and Markie Nuñez, for their wit, wisdom, and friendship; the Hessert grad students for their good lunch time company; George Ross for his general joviality; Ron Hugo and Mark Rennie for their camping companionship; and Keith Gonthier for the never ending, always humorous tales of his hound Alamo. I would also like to thank Mr. Kevin Peters for the time he always found to solve my problems, even

though there was always a long list of other problems to tackle.

Lastly, I would like thank my wife, Debbie, for all her love and support these last eight years. She is always there when I need her, and was the reason I came to Notre Dame. My memories of Notre Dame will always be fond ones because of the time Debbie and I shared here together.

Chapter 1

INTRODUCTION

This dissertation will discuss oblique detonations, their numerical simulation, their applications to propulsion, and their steady and unsteady behavior. In this chapter a description of a generic oblique detonation will be given along with a discussion of applications. Next, the specific questions this research addresses will be given, followed by a brief description of the general format of the remainder of the dissertation.

An oblique detonation is defined here as an oblique shock which induces exothermic chemical reaction in a flow. For the special case in which the flow is two-dimensional and modeled by reactive Euler equations, Fig. 1.1 indicates general fea-

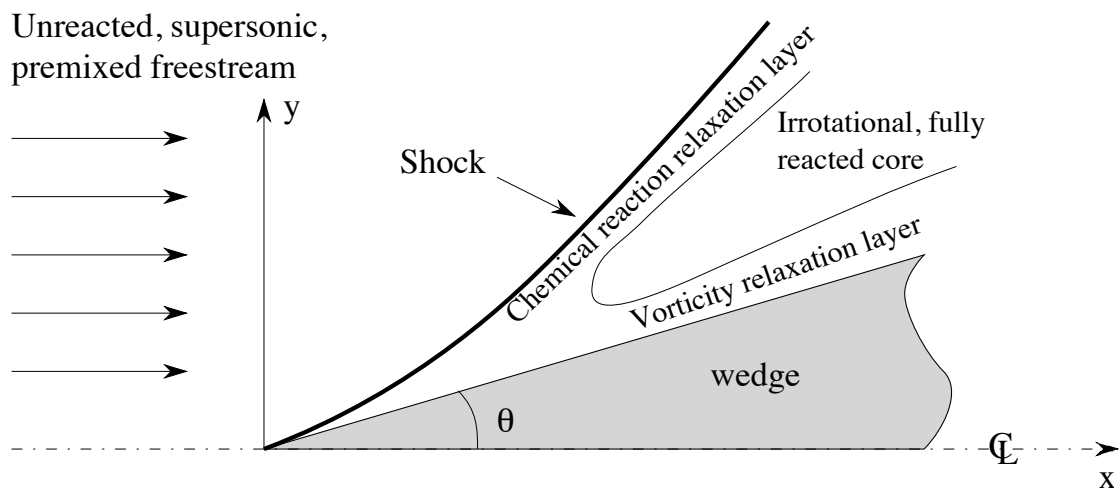


Figure 1.1: Schematic of oblique detonation.

tures of an oblique detonation over a sharp-edged, straight, semi-infinite wedge. An oblique shock is formed off the wedge tip due to the impinging supersonic freestream,

provided the freestream Mach number is high enough to prevent shock detachment. Assuming that the upstream reaction rate is negligibly small and the temperature increase due to the shock is large enough, chemical reaction is initiated in the post-shock flow. Solution of the Rankine-Hugoniot reactive jump equations [69, 34] shows that the oblique detonation wave angle is greater than that of an inert oblique shock wave. When finite reaction zone lengths are considered, the shock does not remain straight [63]. At the apex of the wedge the flow has not had time to react, and thus the shock angle is that of an inert shock. Downstream of the apex the chemical reaction releases increasing amounts of heat into the flow, leading to pressure disturbances which propagate along characteristics that intersect the shock. The shock is then strengthened by the disturbances, the effect of which is the shock being pushed farther from the wall and developing curvature. Far from the wall the shock relaxes to zero curvature. As the only mechanism for generating vorticity in the inviscid flow is shock curvature, vorticity is generated at the shock near the wall, and then convected downstream along particle paths roughly parallel to the wedge surface. Thus, as indicated, there is a layer of chemical reaction following the shock, and a layer of vorticity near the wedge surface. Both layers eventually relax to an irrotational, completely reacted core region downstream of the shock and away from the wedge surface.

Such high-speed nonequilibrium flows have been the subject of many studies over the past decades, and have recently been receiving more attention due to renewed interest in hypersonic air- and spacecraft such as the proposed National Aerospace Plane (NASP). One of the propulsion systems considered for this vehicle is the oblique detonation wave engine (ODWE) [25, 56, 15], which, as the name suggests, relies on oblique detonations to provide thrust. Figure 1.2 illustrates the general concept of the ODWE as proposed by Dunlap, *et al.* [25]. Supersonic air enters the engine and is mixed with fuel. The mixture is then ignited by an oblique shock induced by a

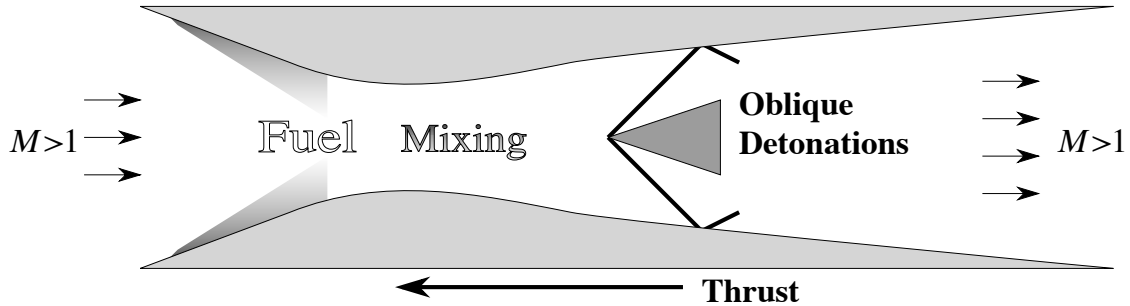


Figure 1.2: Simplistic concept for oblique detonation wave engine.

wedge, greatly increasing the pressure inside the engine. This high pressure forces the combustion products to expand out the back of the engine at high speed, creating thrust.

The ODWE is perceived to have the same advantage over a ramjet as the supersonic combusting ramjet (SCramjet): improved efficiencies at high Mach numbers due to the reduced total pressure losses possible with supersonic combustion [83]. In addition, it is a competitive alternative to the SCramjet because it would use a smaller combustion chamber and not require an active ignition device [25]. On the other hand, a number of issues remain to be addressed regarding the ODWE: Can the supersonic fuel and air be sufficiently mixed? Will the mixture have enough time to react fully before leaving the engine? Will the standing oblique detonations be stable over a broad range of Mach numbers?

Another rapidly emerging application is the ram accelerator, first proposed and studied by Hertzberg, *et al.* [38, 39]. This device uses oblique detonations to drive projectiles to high speed. While currently undergoing basic feasibility studies, envisioned applications include weapons and inexpensive launch systems [7]. Figure 1.3 illustrates one of the theorized modes of propulsion for this device. A projectile is injected at high speed (approximately 1,500 m/s), via a conventional powder or light gas gun, into a tube filled with fuel and oxidizer (methane and oxygen at 20 atm , for example). A conical shock forms over the sharp tip of the projectile, and

is reflected off the interior tube wall. This reflected shock induces rapid combustion, which causes a large increase in pressure behind the projectile (approximately 600 *atm*). The pressure differential between the front and the back of the projectile then accelerates the projectile forward. The projectile would continue to accelerate until the end of the tube was reached, or until the lead shock became strong enough to

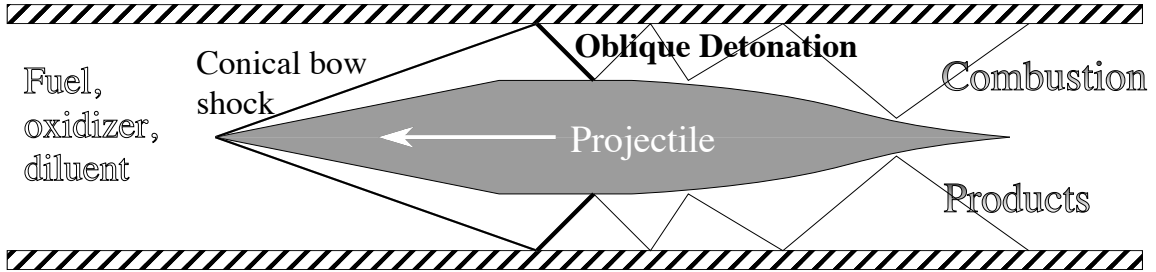


Figure 1.3: Schematic of ram accelerator flow field. [37]

induce combustion and thus increase the pressure on the front of the projectile to such an extent as to balance the aft force. Hertzberg, *et al.* [39] have observed velocities as high as 2,500 *m/s* at the end of the tube.

In order to optimize performance of these devices, it is important to gain a fundamental understanding of oblique detonation behavior. One factor which may degrade performance is combustion instability. Experimental [49] and numerical [28, 9] studies of fundamentally one-dimensional detonations have shown that under certain conditions detonations have an unsteady, oscillating character, while under other conditions they are stable. In the unsteady regime, peak flow quantities (such as pressure, density, or velocity) following the detonation can vary greatly from their average values. Similar multi-dimensional phenomena have been observed experimentally (see chapter 7 of [27]). Thus, for the proper design of devices utilizing oblique detonations, it is important to obtain answers to the following questions:

- What is the relation between one-dimensional detonation instability and oblique detonation instability?

- What is the behavior of oblique detonations in the unstable regime?

Finally, as tools were developed to answer these questions, it became apparent that these same tools could be easily applied to the ram accelerator in order to answer the following questions:

- What is the maximum propagation speed for a ram accelerator?
- How long should it take a ram accelerator to reach steady state?

This study has attempted to answer these questions by numerically solving simplified equations which model oblique detonations. Real combustion phenomena are extremely complex, involving many chemical reactions between many different species. Even in the combustion of simple systems, such as hydrogen and oxygen, it is difficult to determine all intermediate products and the elementary reactions that give rise to those products. The situation is further complicated by the many disparate time scales associated with the reactions, which can differ by as much as ten orders of magnitude [55]. In a detonation, this complex chemistry interacts with a complex flowfield. This is typically modeled with partial differential equations (PDE's) governing the evolution of each chemical species, as well as the standard PDE's describing conservation of mass, momenta, and energy. The numerical solution of this complex system is usually non-trivial due to the difficulty associated with sufficiently resolving all length and time scales of the reacting flow, as well as the question of how to correctly model turbulence.

As an alternative approach, the focus of this study is the solution of the simplest system of equations that still captures the main features of an oblique detonation: the Euler equations for a calorically perfect ideal gas combined with one equation for one irreversible, one-step exothermic reaction with Arrhenius kinetics. These equations are a subset of equation sets that model more complex physics; they are the most basic system to be studied before more complex systems are considered. Consideration of

such systems most efficiently exposes any inadequacies in our understanding. Though the approach has the disadvantage of not modeling certain physical realities (such as real gas effects, viscous effects, *etc.*) it has a long history as a useful tool for understanding [28, 27, 9, 8].

The format of this dissertation is as follows: a general review of the detonation literature will be presented, followed by a presentation of the model equations. Then a number of analytical solutions to the model equations will be presented and discussed. Next, the numerical method used to solve the model equations will be presented, followed by numerical solutions to a number of one- and two-dimensional problems to assess its accuracy. The numerical method will then be applied to oblique detonation problems to determine their steady and unsteady behavior. Finally, conclusions will be drawn from the aforementioned numerical solutions.

Chapter 2

LITERATURE REVIEW

This review of the relevant literature will be divided into two sections: the first will review detonations, while the second will focus on ram accelerators. The detonation literature will be organized in two parts: the first will discuss one-dimensional detonations, covering analytical linear stability results, and then numerical studies of unstable detonations, while the second will discuss oblique detonation results, covering experimental evidence for steady and unsteady oblique detonations, analytical and numerical solutions for nonequilibrium flows with oblique shocks/detonations, and finally, analytical and numerical studies focused on oblique detonation stability. Finally, the ram accelerator literature will be divided into representative experimental and numerical studies.

A number of detonation-related terms in this chapter are first defined. Detonations are typically classified by the overdrive factor $f \equiv (D/D_{CJ})^2$, where D is the wave speed measured with respect to a laboratory frame of a piston-supported detonation; D_{CJ} is the speed of an unsupported, freely travelling detonation; and CJ refers to what is known as the Chapman-Jouguet condition (see Fig. 2.1). The detonation becomes unsupported when the piston velocity drops below a certain velocity denoted u_{CJ} . The freely travelling detonation is commonly referred to as a CJ detonation. When $f > 1$ the detonation is referred to as overdriven, while for CJ waves $f = 1$.

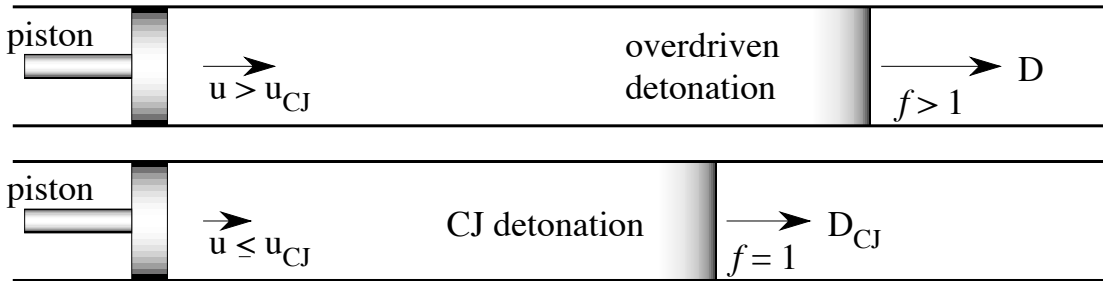


Figure 2.1: Illustration of overdrive.

2.1 One-dimensional Detonations

In the early 1940's Zeldovich [90], von Neumann [79], and Doering [24] independently modeled detonations with what is now known as the ZND model [27]. The ZND detonation is one-dimensional and modeled by reactive Euler equations; it is composed of a travelling shock wave followed by a zone of chemical reaction. The reaction is initiated in the fluid by the shock, which is treated as a discontinuity over which no reaction occurs. The particular studies discussed here focused on the case of a ZND detonation with one irreversible, exothermic chemical reaction.

The stability of ZND detonations has been studied both analytically, using linearized equations, and numerically, using the complete, nonlinear equations. Fickett and Davis [27] present a summary of the limited linear analytical stability results available in 1979. The parameters varied in the studies were the activation energy (a parameter in the chemical reaction model), the amount of heat released in the chemical reaction, the wave number (which is inversely proportional to wavelength) of the transverse disturbance, and the amount of overdrive for the detonation. The results' general trends were as follows: for a transverse wave number of zero, increasing the activation energy lowered the transition threshold to instability for finite heat release and low values of overdrive. Sufficiently high overdrive resulted in stability. For an activation energy of zero there was complete stability for all values of heat release and overdrive. When the transverse wave number approached infinity, increasing

the activation energy again resulted in instability. In this case, however, stability was achieved for low and high levels of heat release and overdrive, but not for intermediate values of the two. Once again, zero activation energy resulted in complete stability. Finally, for intermediate values of transverse wave number, increasing the activation energy resulted in the transition to instability occurring at ever lower values of heat release for all values of overdrive. Even at zero activation energy, higher levels of heat release yielded unstable results. Lee and Stewart [47] recently completed a linear analytical study of the stability of ZND detonations to longitudinal disturbances. Using modern techniques they were able to both support the results of the previous studies, and then fill in and extend the range of parameters (activation energy, heat release, longitudinal wave number) which determine stability boundaries.

Time-accurate numerical studies of ZND detonations have been performed to verify the linear stability analyses and to determine the nonlinear behavior of the detonation. Fickett and Wood [28] used the method of characteristics to find pulsating detonations that were consistent with the linearized stability results. The unstable detonations oscillated about the steady solution, with peak pressures approximately 50% higher than the steady pressure. Later, Bourlioux, Majda, and Roytburd [9] studied unstable detonations and made detailed comparisons to the results of Lee and Stewart [47]. Using an asymptotic analysis, they were able to show that their numerical method was predicting the correct unstable behavior. Then Bourlioux and Majda [8] extended the study to two spatial dimensions, finding complex two-dimensional cellular structures arising from the original one-dimensional, steady detonation. They showed that for certain cases in which the ZND detonation profile was unstable to long transverse wavelength disturbances, the cell spacing predicted by simple theories agreed well with the numerical simulation. When the ZND profile had a complex stability diagram, such as instability to short wavelength transverse perturbations, neither the simple theories nor more complex theories were able to predict the com-

puted cell spacings accurately. Finally, their numerical solutions appeared to show transition to something akin to turbulence in the wake of the unstable detonation front as the activation energy and heat release were increased.

2.2 Oblique Detonations

Experimental results for oblique detonations fall into two categories: wind tunnel data and hypervelocity projectile data. The observations suggest that oblique detonations have regimes of stability and instability. Gross and Chinitz [35] produced apparently steady oblique detonations by injecting hydrogen upstream of a wedge placed in a supersonic wind tunnel. Their main method of measurement was schlieren photographs. Rubins and Rhodes [67] used a similar apparatus and also produced apparently steady oblique detonations. Ruegg and Dorsey [68] obtained early evidence of detonation instability by firing projectiles into combustible mixtures. Later, Behrens, Struth, and Weckens [6] fired spherical projectiles into stoichiometric mixtures of hydrogen and air at low pressures (0.24-0.72 *atm*), and using high speed shadow and schlieren photographs deduced shock pressures, stagnation temperatures, and frequency of shock oscillation. They found that the period of shock oscillation was the same as the induction time for self-ignition of the mixture. Lehr [49] additionally considered projectiles with both spherically and conically shaped noses. Both studies found apparently steady and unsteady combustion phenomena following the bow shock in front of the projectile. Many of these photographs are considered to record examples of fundamentally one-dimensional instabilities (see left side of Fig. 2.2). However, Lehr obtained one photograph (see right side of Fig. 2.2) of what he described as “an oblique Chapman-Jouguet detonation” away from the centerline of the projectile. The photograph shows a straight oblique detonation followed by a complicated, unsteady flowfield, and could be interpreted as an example of oblique detonation instability.

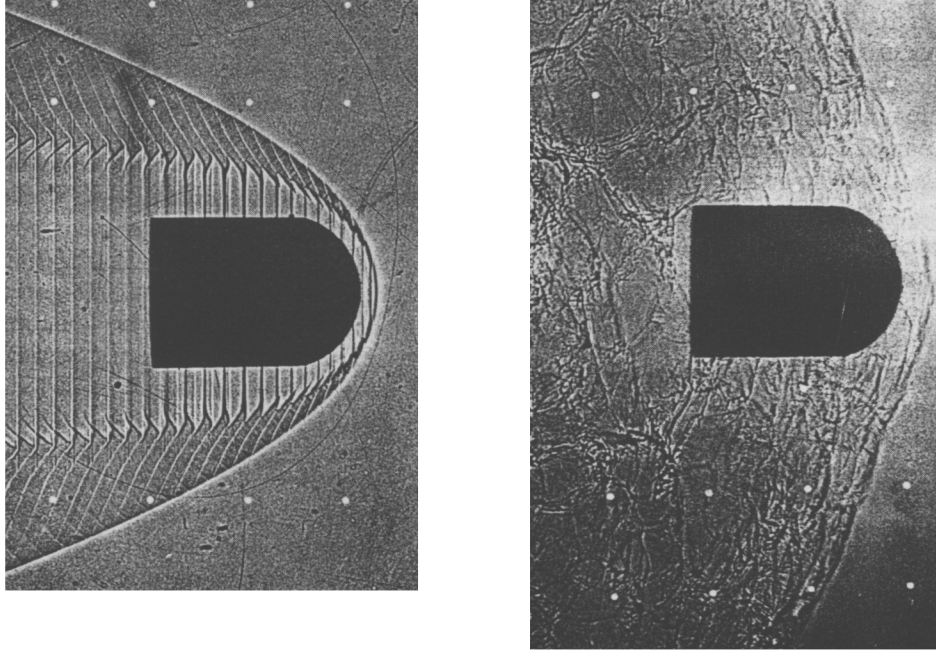


Figure 2.2: Unsteady detonations resulting from hypervelocity projectiles. [49]

There are a number of early representative analytical studies of the type of flow depicted in Fig. 1.1. Vincenti [78] considered supersonic, nonequilibrium flow over a thin wedge, and obtained asymptotic solutions by perturbing about a uniform, equilibrium freestream flow. Capioux and Washington [16] considered the hypersonic flow of a diatomic, dissociating gas over a finite-angle wedge, and obtained flow solutions using the method of characteristics. Lee [48] considered supersonic flow over a wedge in which the fluid was in vibrational nonequilibrium. By perturbing about the frozen shocked-state, he was able to obtain asymptotic solutions for the nonequilibrium flow downstream.

Oblique detonations were originally studied with discontinuity analyses, in which the detonation is considered to be a discontinuous wave with heat release. Siestrunck, *et al.* [71] and Rutkowski and Nicholls [69] presented and discussed oblique detonation polars based upon this analysis. Larisch [44], and later Gross [34], obtained analytical jump equations for the flow properties across the detonation, with heat release as a

parameter. Chernyi [18] discussed the formation of detonations over a number of different bodies, including wedges and cones. Others studies of this sort include Oppenheim, *et al.* [57], Buckmaster and Lee [12], and Pratt, *et al.* [64]. A more complete review of this type of analysis is given by Pratt *et al.*, who discuss and synthesize much of the previous work.

More recently, many numerical solutions of the model partial differential equations for reactive flows have been obtained. Examples include the work of Brackett and Bogdanoff [10], Fujiwara, *et al.* [30, 29], Wang, *et al.* [81], Cambier, *et al.* [14, 15], Yungster, *et al.* [87, 89, 88], Ahuja and Tiwari [2, 1], Li, *et al.* [51, 52, 53], and Tivanov and Rom [77]. Recently, Powers and Stewart [63] obtained asymptotic solutions for a steady oblique detonation which are valid in the limit of high Mach number. Grismer and Powers [32] then calculated numerical solutions to the same equations and compared their results to the asymptotic solutions. These calculations, to be described in Ch. 4, show that at very high Mach numbers (~ 20), the differences between the two solutions were attributable almost entirely to numerical artifacts and the truncation error of the numerical method. On the other hand, at lower Mach numbers (~ 10) the majority of the difference was due to the inaccuracy of the asymptotic solution. The studies of Grismer and Powers were not time accurate, however, and therefore cannot be used to ascertain the stability characteristics of oblique detonations.

Also related to this work are recent studies involving the stability of shocks with detonation. Jackson, Kapila, and Hussaini [42] and Lasseigne, Jackson, and Hussaini [45] examined the passage of a weak vorticity disturbance through a reactive shock wave (detonation) by linearizing about the steady shocked state, and by numerically simulating the model partial differential equations. In particular, they found that heat release due to chemical reaction acts to amplify the vorticity disturbance downstream of the detonation. By considering the asymptotic limit of high activation

energy, they also determined that the vorticity disturbance affects the detonation by causing the fire zone (a thin region of intense chemical reaction) position to oscillate. Buckmaster [11] examined small perturbations to an oblique detonation wave in the high activation energy limit. He determined that these perturbations grow while moving downstream along the wave, and thus concluded that the oblique detonation is structurally unstable in the limit of infinite activation energy.

2.3 Ram Accelerators

The ram accelerator was first proposed and investigated by Hertzberg, Bruckner, and Bogdanoff [38]. In order to test the concept, they built an experimental facility consisting of a light gas gun to launch the projectile at high speed; three connected steel tubes filled with varying amounts of fuel, oxidizer, and diluent; an evacuated dump tank into which the projectile flew after exiting the tubes; and lastly, a large bore decelerator tube filled with steel lathe turnings. Figure 2.3 illustrates this device. Using this facility, projectile velocities and accelerations as high as 1500 m/s

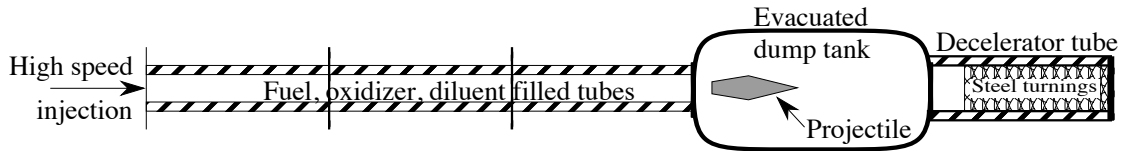


Figure 2.3: Schematic of ram accelerator test facility.

and 16,000 g , respectively, were achieved. The authors suggested possible classes of flowfields termed “propulsive modes” for the ram accelerator, and labeled them subdetonative and superdetonative. Subdetonative modes are limited to speeds below the CJ speed for the gas mixture, while superdetonative modes exist above the CJ speed. Hertzberg, Bruckner, and Knowlen [39] then studied these modes in more detail. They found that by filling each of the three tubes with different mixtures of fuel, oxidizer, and diluent, and thus altering the CJ speed, subdetonative propulsion could be maintained and still accelerate the projectile. The same technique could

also be used to force a transition from subdetonative propulsion in one tube to superdetonative propulsion in the next. Finally, it was found that the projectile could transition from subdetonative to superdetonative propulsion when the gas mixture was the same throughout the tubes. Using these techniques, the maximum velocity achieved by the projectile was increased to 2,500 m/s .

A number of numerical studies of ram accelerators have been performed. Brackett and Bogdanoff [10] used a Godunov scheme to solve the Euler equations combined with one global Arrhenius rate expression for all the chemical reactions. They found that an oblique detonation could occur where the initial conical shock attached to the projectile nose reflected off the tube wall. An oblique detonation could also be induced by placing a small ramp at the midpoint of the projectile body. In either case, they found positive thrust on the projectile. Yungster, Eberhardt, and Bruckner [89] developed a code to solve the Euler equations with detailed multispecies, multireaction chemistry and real gas effects. The code was verified using experimental results for hypersonic, exothermic blunt body flows, and then applied to a ram accelerator configuration at two flight speeds. In both cases there was a positive thrust on the projectile. Yungster and Bruckner [88] then performed a detailed study using the code, investigating the performance characteristics of various projectile configurations in the range of 5-10 km/s . They examined the effects of varying projectile geometry, tube cross sectional area, and gas mixture on the net thrust developed on the projectile. Positive thrusts were found in all cases. Yungster [87] studied the shock-wave/boundary-layer interaction on a ram accelerator configuration using a code developed to solve the Reynolds-averaged Navier-Stokes equations with detailed multispecies, multireaction chemistry and a Baldwin-Lomax algebraic turbulence model. The analysis centered on an oblique detonation propulsion mode, and indicated that a reflected shock wave initiated significant combustion in the boundary layer on the projectile. If the projectile speed was increased, significant combustion

began in the boundary layer spontaneously. For one flight speed the thrust on the projectile was determined, and it was found to be approximately ten percent lower than the corresponding inviscid case.

Other representative experimental and numerical studies of ram accelerators include Kruczynski and Nusca [43], Srulijes, *et al.* [73], and Sinha, *et al.* [72]. These are just a few of the studies on ram accelerators. Much of the most recent work in this area is summarized in the proceedings of a recent international workshop [41].

None of the studies mentioned has addressed the nonlinear stability of oblique detonations attached to wedges and modeled by the reactive Euler equations. The early analytical studies did not consider stability, while the recent numerical studies mentioned considered complicated, multireaction chemistry models. Jackson, *et al.* and Buckmaster use a linearized analysis, while Lasseigne, *et al.* do not consider a resolved reaction zone. The two-dimensional simulations of Bourlioux and Majda consider the reactive Euler equations with a resolved reaction zone, but use periodic boundary conditions; thus, they do not consider the effect of wall boundaries such as a wedge. In this research oblique detonations attached to wedges and having resolved reaction zones were studied in order to relate oblique detonation instability to one-dimensional detonation instability, and to determine the behavior of unstable oblique detonations.

Grismer and Powers [33] sought to determine the maximum steady propagation speed of an idealized ram accelerator projectile. None of the studies referenced here considered this aspect of the ram accelerator. In the experimental studies the projectile continuously accelerated through the end of the tube, and thus no steady speeds were obtained. In the numerical studies only individual points of positive and negative thrust on the projectile were studied; there was little emphasis placed on determining steady speeds. A methodology will be presented here for determining the steady speed, along with the results that were obtained for a model problem.

Chapter 3

MODEL EQUATIONS

This chapter introduces the model equations employed in this research. The first section presents the equations and their underlying assumptions. The equations are then further simplified algebraically and nondimensionalized. The second section presents the transformation of the equations to a generalized, curvilinear coordinate system. This coordinate system allows for the solution of the equations within geometrically arbitrary domains.

3.1 Cartesian Reactive Euler Equations

In a wide variety of scenarios, the model equations for a reactive, viscous, Newtonian fluid can be taken to be the Navier-Stokes equations, with additional terms to account for the chemical reactions [85, 13]. However, as the purpose of this research is to ascertain the characteristics of oblique detonations by studying the simplest system of equations having oblique detonation solutions, the following assumptions have been made:

1. Molecular transport processes are negligible, i.e. there is no momentum, energy, or species diffusion.
2. Body forces are negligible.
3. There are no internal heat generation or radiation effects.

4. There is one irreversible reaction ($A \rightarrow B$).
5. A and B are calorically perfect ideal gases.
6. The specific heats and molecular weights of A and B are identical.
7. Three-dimensional effects are negligible.

These simplifying assumptions remove many physical features that are contained in the solution of the complete reactive Navier-Stokes equations. The physical flow features that are being neglected can be qualitatively described by considering each of the assumptions in turn. Neglecting transport processes eliminates the effects of viscosity and thermal conduction on the flow. Therefore, there will be no viscous or thermal boundary layers near solid bodies, and flow separation is not a possibility. There is no mechanism for the solid surface to impart heat to the flow; it acts essentially as an adiabatic body. The lack of viscosity and restriction to two dimensions means momentum diffusion generated turbulence is also no longer possible, so numerical resolution of the very small scales associated with this type of turbulence is not necessary. Removing the viscous terms from the equations results in a hyperbolic system of equations in time, instead of a mixed hyperbolic-elliptic system. The hyperbolic system has real characteristics, which will be taken advantage of later in the numerical method. Also, since the dissipation mechanisms have been neglected, sharp gradients contained in the solution of complete equations may become discontinuous in the solution of the reduced equations.

Neglecting body forces eliminates the possibility of gravity driven flows; there will be no gravity-induced density gradients. Disregarding internal heat generation removes the possibility of using heat sources (or sinks) to affect the flow or to influence the chemical reactions. This is not to say that the chemical reaction itself does not release (or absorb) heat; this is accounted for in the internal energy of the flow. Radiation is another transport process. It would tend to smooth sharp temperature

gradients and equalize the temperature everywhere by transporting heat away from local hot spots in the flow. Neglecting radiation, like the other transport effects, thus permits for local high temperature regions and discontinuous solutions.

Finally, the remaining assumptions limit the complexity of the chemical reactions and the fluid properties. Considering one irreversible reaction limits the equations to only the very simplest possible reacting system. Real reacting flows have much more complex chemistry involving multiple reversible reactions among multiple species. The reactions can be both exo- and endothermic, and occur over widely varying time scales. Thus, the one reaction model is only in the very broadest sense applicable to real chemically reacting flows. Considering the species to be calorically perfect ideal gases with identical properties further limits the applicability to real systems. This is a very broad simplification, since real gas properties vary with the thermodynamic state. This is particularly true for reacting systems, which often have large changes in temperature and pressure. Additionally, the properties of the products and reactants themselves are also in general different. Lastly, the ideal gas assumption does not account for situations in which the gas reaches non-ideal conditions, such as very high pressure and low temperature. It also does not account for gases that do not behave ideally.

Though many of these assumptions are made for tractability, there also exist some physical justifications. For the high speed flows considered here, transport processes generally are not significant. The boundary layer is very thin, and thus most of the detonation occurs in the essentially inviscid region outside of it. Heat and species diffusion have very little time to occur as the flow is convected through the detonation and downstream. Anderson [4] discusses many of the useful inviscid solutions that have been obtained for both inert and reactive flows.

With the simplifying assumptions the equations of [85] reduce to the following:

$$\frac{\partial \tilde{\rho}}{\partial t} + \frac{\partial}{\partial \tilde{x}}(\tilde{\rho}\tilde{u}) + \frac{\partial}{\partial \tilde{y}}(\tilde{\rho}\tilde{v}) = 0, \quad (3.1)$$

$$\frac{\partial}{\partial \tilde{t}}(\tilde{\rho}\tilde{u}) + \frac{\partial}{\partial \tilde{x}}(\tilde{\rho}\tilde{u}^2 + \tilde{p}) + \frac{\partial}{\partial \tilde{y}}(\tilde{\rho}\tilde{u}\tilde{v}) = 0, \quad (3.2)$$

$$\frac{\partial}{\partial \tilde{t}}(\tilde{\rho}\tilde{v}) + \frac{\partial}{\partial \tilde{x}}(\tilde{\rho}\tilde{u}\tilde{v}) + \frac{\partial}{\partial \tilde{y}}(\tilde{\rho}\tilde{v}^2 + \tilde{p}) = 0, \quad (3.3)$$

$$\frac{\partial}{\partial \tilde{t}}(\tilde{\rho}\tilde{E}) + \frac{\partial}{\partial \tilde{x}}(\tilde{\rho}\tilde{u}\tilde{H}) + \frac{\partial}{\partial \tilde{y}}(\tilde{\rho}\tilde{v}\tilde{H}) = 0, \quad (3.4)$$

$$\frac{\partial}{\partial \tilde{t}}(\tilde{\rho}Y_2) + \frac{\partial}{\partial \tilde{x}}(\tilde{\rho}\tilde{u}Y_2) + \frac{\partial}{\partial \tilde{y}}(\tilde{\rho}\tilde{v}Y_2) = \tilde{\rho}\tilde{B}\tilde{T}^\alpha e^{\hat{E}/\tilde{\mathfrak{R}}\tilde{T}}(1 - Y_2), \quad (3.5)$$

where “ \sim ” indicates a dimensional quantity. Equations (3.1-3.4) represent conservation of mass, momenta, and energy, respectively, and Eq. (3.5) represents evolution of the product mass fraction. Dependent variables in the equations are the density $\tilde{\rho}$, the Cartesian velocities \tilde{u} and \tilde{v} , the pressure \tilde{p} , the total energy (internal and kinetic) per unit mass \tilde{E}^\dagger , the total enthalpy per unit mass \tilde{H}^\dagger , the product mass fraction Y_2 , and the temperature \tilde{T} . Parameters in the equations are the Arrhenius prefactor \tilde{B} , an exponent determining the temperature dependency of the reaction α , the activation energy for the reaction \hat{E} , and the universal gas constant $\tilde{\mathfrak{R}}$. The total energy is related to the specific internal energy (\tilde{e}) by $\tilde{E} = \tilde{e} + (\tilde{u}^2 + \tilde{v}^2)/2$, and to the total enthalpy by $\tilde{H} = \tilde{E} + \tilde{p}/\tilde{\rho}$. Finally, since the species mass fractions must add to one, the reactant mass fraction can be determined by

$$Y_1 = 1 - Y_2. \quad (3.6)$$

For a multispecies mixture, the ideal gas law and the caloric equation of state are

$$\tilde{p} = \tilde{\rho}\tilde{\mathfrak{R}}\tilde{T} \sum_{i=1}^{N_{sp}} \frac{Y_i}{\tilde{M}_i}, \quad (3.7)$$

$$\tilde{h}_i = \tilde{h}_i^0 + \int_{\tilde{T}^0}^{\tilde{T}} \tilde{c}_{p,i} d\tilde{T}, \quad (3.8)$$

respectively, where \tilde{h}_i is the specific enthalpy of the i^{th} species, \tilde{h}_i^0 is the standard enthalpy of formation per unit mass for species i at the temperature \tilde{T}^0 , $\tilde{c}_{p,i}$ is the specific heat at constant pressure for the i^{th} species, \tilde{M}_i is the molecular weight of

[†]This notation is common in computational fluid mechanics literature and should not be confused with that of classical thermodynamics, which reserves capital letters for extensive properties

the i^{th} species, and N_{sp} is the total number of species. For $N_{sp} = 2$, $\tilde{M}_1 = \tilde{M}_2 = \tilde{M}$, and $\tilde{c}_{p,1} = \tilde{c}_{p,2} = \tilde{c}_p$, these reduce to

$$\tilde{p} = \tilde{\rho}\tilde{R}\tilde{T}, \quad (3.9)$$

$$\tilde{h}_1 = \tilde{h}_1^0 + \tilde{c}_p(\tilde{T} - \tilde{T}_0), \quad (3.10)$$

$$\tilde{h}_2 = \tilde{h}_2^0 + \tilde{c}_p(\tilde{T} - \tilde{T}_0), \quad (3.11)$$

where $\tilde{R} = \tilde{\mathfrak{R}}/\tilde{M}$ is the gas constant. Finally, these can be related to the specific internal energy using the identity

$$\tilde{e} = \tilde{h}_1 Y_1 + \tilde{h}_2 Y_2 - \frac{\tilde{p}}{\tilde{\rho}}, \quad (3.12)$$

which in this case reduces to

$$\tilde{e} = \frac{\tilde{p}}{\tilde{\rho}(\gamma - 1)} - Y_2 \tilde{q}. \quad (3.13)$$

In (3.13) a heat release parameter, $\tilde{q} = \tilde{h}_1^0 - \tilde{h}_2^0$, has been defined, the heat of formation of species one has been taken to be $\tilde{h}_1^0 = \tilde{c}_p \tilde{T}^0$, $\tilde{c}_v = \tilde{c}_p - \tilde{R}$ is the specific heat at constant volume, and $\gamma = \tilde{c}_p/\tilde{c}_v$ is the ratio of specific heats. Finally, for a calorically perfect ideal gas, the isentropic frozen speed of sound \tilde{c} is

$$\tilde{c} = \sqrt{\gamma \tilde{p}/\tilde{\rho}}. \quad (3.14)$$

The Mach number \mathcal{M} can then be defined as

$$\mathcal{M} = \frac{\sqrt{\tilde{u}^2 + \tilde{v}^2}}{\tilde{c}}. \quad (3.15)$$

By defining the activation energy on a mass basis, $\tilde{E}_a = \tilde{E}/\tilde{M}$, removing the temperature dependency from the Arrhenius term, $\alpha \equiv 0$, and using the ideal gas law to replace temperature in favor of pressure and density, the species equation (Eq. 3.5) can be simplified to

$$\frac{\partial}{\partial \tilde{t}}(\tilde{\rho} Y_2) + \frac{\partial}{\partial \tilde{x}}(\tilde{\rho} \tilde{u} Y_2) + \frac{\partial}{\partial \tilde{y}}(\tilde{\rho} \tilde{v} Y_2) = \tilde{\rho} \tilde{B} (1 - Y_2) e^{-\tilde{E}_a \tilde{\rho}/\tilde{p}}. \quad (3.16)$$

Equations (3.1, 3.2, 3.3, 3.4, and 3.16) can be nondimensionalized by the following scheme:

$$\begin{aligned} x &= \frac{\tilde{B}\tilde{x}}{\mathcal{M}_0\sqrt{\tilde{p}_0/\tilde{\rho}_0}}, & y &= \frac{\tilde{B}\tilde{y}}{\mathcal{M}_0\sqrt{\tilde{p}_0/\tilde{\rho}_0}}, & u &= \frac{\tilde{u}}{\mathcal{M}_0\sqrt{\tilde{p}_0/\tilde{\rho}_0}}, & v &= \frac{\tilde{v}}{\mathcal{M}_0\sqrt{\tilde{p}_0/\tilde{\rho}_0}}, \\ \rho &= \frac{\tilde{\rho}}{\tilde{\rho}_0}, & p &= \frac{\tilde{p}}{\mathcal{M}_0^2\tilde{p}_0}, & t &= \tilde{t}\tilde{B}, \end{aligned} \quad (3.17)$$

where \mathcal{M}_0 , \tilde{p}_0 , and $\tilde{\rho}_0$ are the freestream Mach number, pressure, and density, respectively. Using this scheme, continuity (Eq. 3.1), momenta (Eqs. 3.2 and 3.3), and energy (Eq. 3.4) do not change form:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) = 0, \quad (3.18)$$

$$\frac{\partial}{\partial t}(\rho u) + \frac{\partial}{\partial x}(\rho u^2 + p) + \frac{\partial}{\partial y}(\rho uv) = 0, \quad (3.19)$$

$$\frac{\partial}{\partial t}(\rho v) + \frac{\partial}{\partial x}(\rho uv) + \frac{\partial}{\partial y}(\rho v^2 + p) = 0, \quad (3.20)$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x}(\rho u H) + \frac{\partial}{\partial y}(\rho v H) = 0. \quad (3.21)$$

The species equation (Eq. 3.16) becomes

$$\frac{\partial}{\partial t}(\rho Y_2) + \frac{\partial}{\partial x}(\rho u Y_2) + \frac{\partial}{\partial y}(\rho v Y_2) = \rho(1 - Y_2)e^{-\Theta\rho/p}. \quad (3.22)$$

The supplementary algebraic relations for secondary variables become

$$e = \frac{p}{\rho(\gamma - 1)} - Y_2 q, \quad (3.23)$$

$$T = \frac{p}{\rho}, \quad (3.24)$$

$$c = \sqrt{\gamma p/\rho}, \quad (3.25)$$

$$E = e + \frac{1}{2}(u^2 + v^2), \quad (3.26)$$

$$H = E + \frac{p}{\rho}, \quad (3.27)$$

respectively, where $\Theta = \tilde{E}_a\tilde{\rho}_0/\mathcal{M}_0^2\tilde{p}_0$, $e = \tilde{e}\tilde{\rho}_0/\mathcal{M}_0^2\tilde{p}_0$, $q = \tilde{q}\tilde{\rho}_0/\mathcal{M}_0^2\tilde{p}_0$, $T = \tilde{T}/\mathcal{M}_0^2\tilde{T}_0$, $c = \tilde{c}/\mathcal{M}_0\sqrt{\tilde{p}_0/\tilde{\rho}_0}$, $E = \tilde{E}\tilde{\rho}_0/\mathcal{M}_0^2\tilde{p}_0$, and $H = \tilde{H}\tilde{\rho}_0/\mathcal{M}_0^2\tilde{p}_0$. Powers and Stewart [63] show that in the limit as $M_0 \rightarrow \infty$, the reaction zone length is $O(\tilde{u}_0\tilde{B}^{-1} =$

$\mathcal{M}_0\sqrt{\gamma\tilde{p}_0/\tilde{\rho}_0\tilde{B}^{-1}}$). The time to cross the reaction zone is thus $O(\tilde{B}^{-1})$. Noting that γ is $O(1)$ and comparing these quantities to the nondimensionalization scheme, it is apparent that the variables have essentially been nondimensionalized by the characteristic length and time scales of the flow. Thus, the variables have been nondimensionalized based upon characteristic freestream quantities, and, in the limit of high Mach number, the characteristic length of the reaction zone and time for the reaction to take place.

3.2 Generalized Curvilinear Reactive Euler Equations

For the numerical portion of this research, Eqs. (3.18-3.22) are not the preferred form; it is more useful to have the equations written in the conservative form for an arbitrary curvilinear coordinate system. The curvilinear coordinate system allows the equations to be solved over arbitrary, complex shapes, unlike the Cartesian equivalents. As is commonly done with the two-dimensional, Cartesian Euler equations [40], Eqs. (3.18-3.22) can be written in a vector conservative form

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = \mathbf{w}, \quad (3.28)$$

where

$$\mathbf{q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \\ \rho Y_2 \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uH \\ \rho uY_2 \end{bmatrix}, \quad \mathbf{g} = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vH \\ \rho vY_2 \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \rho(1 - Y_2)e^{-\Theta\rho/p} \end{bmatrix}.$$

The vectors \mathbf{f} and \mathbf{g} are commonly called the flux vectors. These equations can be transformed to the curvilinear space (ξ, η, τ) by taking

$$\xi = \xi(x, y, t), \quad \eta = \eta(x, y, t), \quad \tau = t, \quad (3.29)$$

and using the chain rule

$$\frac{\partial}{\partial x} = \frac{\partial \xi}{\partial x} \frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial x} \frac{\partial}{\partial \eta},$$

$$\begin{aligned}\frac{\partial}{\partial y} &= \frac{\partial \xi}{\partial y} \frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial y} \frac{\partial}{\partial \eta}, \\ \frac{\partial}{\partial t} &= \frac{\partial \xi}{\partial t} \frac{\partial}{\partial \xi} + \frac{\partial \eta}{\partial t} \frac{\partial}{\partial \eta} + \frac{\partial}{\partial \tau}.\end{aligned}\tag{3.30}$$

The terms ξ_x , ξ_y , ξ_t , η_x , η_y , and η_t (where subscripts denote partial derivatives with respect to the subscripted variables) are referred to as the grid metrics. Since the transformation defined in Eqs. (3.29) is general, the particular form chosen for $\xi(x, y, t)$ and $\eta(x, y, t)$ is arbitrary. Given a computational grid of points in Cartesian space, Fig. 3.1 illustrates the chosen transformation to curvilinear space: each grid point

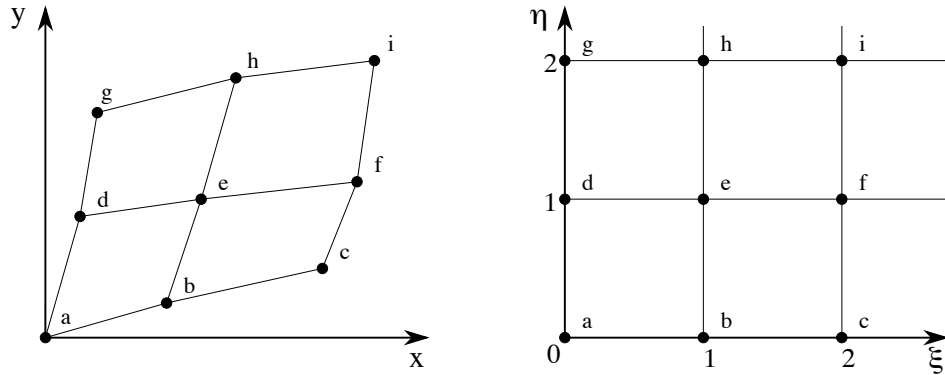


Figure 3.1: Illustration of the transformation from Cartesian space to the generalized, curvilinear space.

is defined to lie at integer locations of ξ and η . In this way $\Delta\xi = \Delta\eta = 1$, which simplifies the computations necessary to determine the grid metrics. The metrics can be determined from known derivatives in Cartesian space once a computational grid has been defined. This is accomplished as follows [3]. The changes in ξ , η , and τ can be written

$$\begin{aligned}d\xi &= \xi_x dx + \xi_y dy + \xi_t dt, \\ d\eta &= \eta_x dx + \eta_y dy + \eta_t dt, \\ d\tau &= dt,\end{aligned}\tag{3.31}$$

or in matrix form

$$\begin{bmatrix} d\xi \\ d\eta \\ d\tau \end{bmatrix} = \begin{bmatrix} \xi_x & \xi_y & \xi_t \\ \eta_x & \eta_y & \eta_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dx \\ dy \\ dt \end{bmatrix}.\tag{3.32}$$

Similarly, assuming Eqs. (3.29) can be inverted such that $x = x(\xi, \eta, \tau)$, $y = y(\xi, \eta, \tau)$, and $t = \tau$, the changes in x , y , and t can be written

$$\begin{bmatrix} dx \\ dy \\ dt \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta & x_\tau \\ y_\xi & y_\eta & y_\tau \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} d\xi \\ d\eta \\ d\tau \end{bmatrix}. \quad (3.33)$$

Solving Eq. (3.33) for $d\xi$, $d\eta$, and $d\tau$,

$$\begin{bmatrix} d\xi \\ d\eta \\ d\tau \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta & x_\tau \\ y_\xi & y_\eta & y_\tau \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} dx \\ dy \\ dt \end{bmatrix}. \quad (3.34)$$

and comparing to Eq. (3.32), it is apparent that

$$\begin{bmatrix} \xi_x & \xi_y & \xi_t \\ \eta_x & \eta_y & \eta_t \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} x_\xi & x_\eta & x_\tau \\ y_\xi & y_\eta & y_\tau \\ 0 & 0 & 1 \end{bmatrix}^{-1}. \quad (3.35)$$

The inverse matrix can be written explicitly as

$$\begin{bmatrix} x_\xi & x_\eta & x_\tau \\ y_\xi & y_\eta & y_\tau \\ 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} y_\eta J & -x_\eta J & (x_\eta y_\tau - x_\tau y_\eta) J \\ -y_\xi J & x_\xi J & (x_\tau y_\xi - x_\xi y_\tau) J \\ 0 & 0 & 1 \end{bmatrix}, \quad (3.36)$$

where $J = 1/(x_\xi y_\eta - y_\xi x_\eta)$. Thus a term by term comparison shows that

$$\begin{aligned} \xi_x &= y_\eta J, & \xi_y &= -x_\eta J, & \xi_t &= (x_\eta y_\tau - x_\tau y_\eta) J = -x_\tau \xi_x - y_\tau \xi_y, \\ \eta_x &= -y_\xi J, & \eta_y &= x_\xi J, & \eta_t &= (x_\tau y_\xi - x_\xi y_\tau) J = -x_\tau \eta_x - y_\tau \eta_y. \end{aligned} \quad (3.37)$$

All of the terms on the right hand sides of Eqs. (3.37) can be determined through finite differencing using the grid point locations in Cartesian space and the transformation to curvilinear space defined previously.

Applying Eqs. (3.30) to Eqs. (3.28) results in

$$\frac{\partial \mathbf{q}}{\partial \tau} + \xi_t \frac{\partial \mathbf{q}}{\partial \xi} + \eta_t \frac{\partial \mathbf{q}}{\partial \eta} + \xi_x \frac{\partial \mathbf{f}}{\partial \xi} + \eta_x \frac{\partial \mathbf{f}}{\partial \eta} + \xi_y \frac{\partial \mathbf{g}}{\partial \xi} + \eta_y \frac{\partial \mathbf{g}}{\partial \eta} = \mathbf{w}. \quad (3.38)$$

Writing out the metric quantities using Eqs. (3.37) and dividing through by J leads to

$$\frac{1}{J} \frac{\partial \mathbf{q}}{\partial \tau} + (y_\tau x_\eta - x_\tau y_\eta) \frac{\partial \mathbf{q}}{\partial \xi} + y_\eta \frac{\partial \mathbf{f}}{\partial \xi} - x_\eta \frac{\partial \mathbf{g}}{\partial \xi} + (x_\tau y_\xi - y_\tau x_\xi) \frac{\partial \mathbf{q}}{\partial \eta} - y_\xi \frac{\partial \mathbf{f}}{\partial \eta} + x_\xi \frac{\partial \mathbf{g}}{\partial \eta} = \frac{\mathbf{w}}{J}. \quad (3.39)$$

Since the conservative form of the equations is necessary to capture shock speeds accurately [50], it is necessary to have the curvilinear equations in the form of Eqs. (3.28). To this end the following groups are useful:

$$\begin{aligned} \frac{\partial}{\partial \xi} [(y_\tau x_\eta - x_\tau y_\eta) \mathbf{q} + y_\eta \mathbf{f} - x_\eta \mathbf{g}] &= (y_\tau x_\eta - x_\tau y_\eta) \frac{\partial \mathbf{q}}{\partial \xi} + y_\eta \frac{\partial \mathbf{f}}{\partial \xi} - x_\eta \frac{\partial \mathbf{g}}{\partial \xi} - \mathbf{g} x_{\xi\eta} + \\ &\quad \mathbf{q} [y_{\tau\xi} x_\eta + y_\tau x_{\xi\eta} - x_{\tau\xi} y_\eta - x_\tau y_{\eta\xi}] + \mathbf{f} y_{\xi\eta} \end{aligned} \quad (3.40)$$

$$\begin{aligned} \frac{\partial}{\partial \eta} [(x_\tau y_\xi - y_\tau x_\xi) \mathbf{q} - y_\xi \mathbf{f} + x_\xi \mathbf{g}] &= (x_\tau y_\xi - y_\tau x_\xi) \frac{\partial \mathbf{q}}{\partial \eta} - y_\xi \frac{\partial \mathbf{f}}{\partial \eta} + x_\xi \frac{\partial \mathbf{g}}{\partial \eta} + \mathbf{g} x_{\eta\xi} + \\ &\quad \mathbf{q} [x_{\tau\eta} y_\xi + x_\tau y_{\eta\xi} - y_{\tau\eta} x_\xi - y_\tau x_{\xi\eta}] - \mathbf{f} y_{\xi\eta} \end{aligned} \quad (3.41)$$

Comparing the last three sets of equations, it is apparent that the first three terms on the right sides of Eqs. (3.40) and (3.41) match the last six terms on the left side of Eqs. (3.39). Substituting and canceling terms results in

$$\begin{aligned} \frac{1}{J} \frac{\partial \mathbf{q}}{\partial \tau} + \frac{\partial}{\partial \xi} [(y_\tau x_\eta - x_\tau y_\eta) \mathbf{q} + y_\eta \mathbf{f} - x_\eta \mathbf{g}] + \frac{\partial}{\partial \eta} [(x_\tau y_\xi - y_\tau x_\xi) \mathbf{q} - \\ y_\xi \mathbf{f} + x_\xi \mathbf{g}] - \mathbf{q} [y_{\tau\xi} x_\eta + x_{\tau\eta} y_\xi - x_{\tau\xi} y_\eta - y_{\tau\eta} x_\xi] = \frac{\mathbf{w}}{J}. \end{aligned} \quad (3.42)$$

Noting that

$$\frac{\partial}{\partial \tau} \left(\frac{\mathbf{q}}{J} \right) = \frac{\partial}{\partial \tau} [\mathbf{q} (x_\xi y_\eta - y_\xi x_\eta)] = \frac{1}{J} \frac{\partial \mathbf{q}}{\partial \tau} + \mathbf{q} (x_{\xi\tau} y_\eta + x_\xi y_{\eta\tau} - y_{\xi\tau} x_\eta - y_\xi x_{\eta\tau}), \quad (3.43)$$

Eqs. (3.42) becomes

$$\frac{\partial}{\partial \tau} \left(\frac{\mathbf{q}}{J} \right) + \frac{\partial}{\partial \xi} [(y_\tau x_\eta - x_\tau y_\eta) \mathbf{q} + y_\eta \mathbf{f} - x_\eta \mathbf{g}] + \frac{\partial}{\partial \eta} [(x_\tau y_\xi - y_\tau x_\xi) \mathbf{q} - y_\xi \mathbf{f} + x_\xi \mathbf{g}] = \frac{\mathbf{w}}{J}. \quad (3.44)$$

Finally, resubstituting the metric quantities from Eqs. (3.37) leads to

$$\frac{\partial}{\partial \tau} \left(\frac{\mathbf{q}}{J} \right) + \frac{\partial}{\partial \xi} \left[\frac{\xi_t \mathbf{q} + \xi_x \mathbf{f} + \xi_y \mathbf{g}}{J} \right] + \frac{\partial}{\partial \eta} \left[\frac{\eta_t \mathbf{q} + \eta_x \mathbf{f} + \eta_y \mathbf{g}}{J} \right] = \frac{\mathbf{w}}{J}. \quad (3.45)$$

Thus, the conservative, curvilinear form of the governing equations is

$$\frac{\partial \hat{\mathbf{q}}}{\partial \tau} + \frac{\partial \hat{\mathbf{f}}}{\partial \xi} + \frac{\partial \hat{\mathbf{g}}}{\partial \eta} = \hat{\mathbf{w}}, \quad (3.46)$$

where

$$\begin{aligned}
\hat{\mathbf{q}} &= \frac{\mathbf{q}}{J} = J^{-1} \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \\ \rho Y_2 \end{bmatrix}, \\
\hat{\mathbf{f}} &= \frac{\xi_t \mathbf{q} + \xi_x \mathbf{f} + \xi_y \mathbf{g}}{J} = J^{-1} \begin{bmatrix} \rho U^c \\ \rho u U^c + \xi_x p \\ \rho v U^c + \xi_y p \\ \rho H U^c - \xi_t p \\ \rho Y_2 U^c \end{bmatrix}, \\
\hat{\mathbf{g}} &= \frac{\eta_t \mathbf{q} + \eta_x \mathbf{f} + \eta_y \mathbf{g}}{J} = J^{-1} \begin{bmatrix} \rho V^c \\ \rho u V^c + \eta_x p \\ \rho v V^c + \eta_y p \\ \rho H V^c - \eta_t p \\ \rho Y_2 V^c \end{bmatrix}, \\
\hat{\mathbf{w}} &= \frac{\mathbf{w}}{J} = J^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \rho(1 - Y_2)e^{-\Theta\rho/p} \end{bmatrix}.
\end{aligned}$$

The contravariant velocities, $U^c = \xi_t + \xi_x u + \xi_y v$ and $V^c = \eta_t + \eta_x u + \eta_y v$, represent the velocities along the coordinate directions ξ and η , respectively.

Because the numerical scheme to be used in this research takes advantage of the characteristic formulation of the Euler equations, it is necessary to modify the equations further. In particular, it necessary to obtain the eigenvalues, right eigenvectors, and left eigenvectors of the flux Jacobian matrices (defined below) in the curvilinear coordinate system. This is done as follows: Eqs. (3.28) can be written in the following form

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{q}}{\partial x} + \mathbf{B} \frac{\partial \mathbf{q}}{\partial y} = \mathbf{w}, \quad (3.47)$$

where \mathbf{A} and \mathbf{B} are the flux Jacobians

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{q}} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \frac{\gamma-1}{2}(u^2 + v^2) - u^2 & (3 - \gamma)u & (1 - \gamma)v & (\gamma - 1) & (\gamma - 1)q \\ -uv & v & u & 0 & 0 \\ \frac{\gamma-1}{2}u(u^2 + v^2) - uH & H + (1 - \gamma)u^2 & (1 - \gamma)uv & \gamma u & (\gamma - 1)uq \\ -uY_2 & Y_2 & 0 & 0 & u \end{bmatrix},$$

$$\mathbf{B} = \frac{\partial \mathbf{g}}{\partial \mathbf{q}} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ -uv & v & u & 0 & 0 \\ \frac{\gamma-1}{2}(u^2+v^2) - v^2 & (1-\gamma)u & (3-\gamma)v & (\gamma-1) & (\gamma-1)q \\ \frac{\gamma-1}{2}v(u^2+v^2) - vH & (1-\gamma)uv & H + (1-\gamma)v^2 & \gamma v & (\gamma-1)vq \\ -vY_2 & 0 & Y_2 & 0 & v \end{bmatrix}.$$

As before, the curvilinear version of these equations is desired. Thus, Eqs. (3.46) can be written

$$\frac{\partial \hat{\mathbf{q}}}{\partial \tau} + \hat{\mathbf{A}} \frac{\partial \hat{\mathbf{q}}}{\partial \xi} + \hat{\mathbf{B}} \frac{\partial \hat{\mathbf{q}}}{\partial \eta} = \hat{\mathbf{w}}. \quad (3.48)$$

Knowing the definitions of the flux terms $\hat{\mathbf{f}}$ and $\hat{\mathbf{g}}$ in Eq. (3.46), the chain rule, and the Cartesian flux Jacobians, the curvilinear flux Jacobians can be determined:

$$\begin{aligned} \hat{\mathbf{f}} &= \frac{\xi_t \mathbf{q} + \xi_x \mathbf{f} + \xi_y \mathbf{g}}{J} \\ &= f[\mathbf{q}(\hat{\mathbf{q}}), \mathbf{f}(\mathbf{q}(\hat{\mathbf{q}})), \mathbf{g}(\mathbf{q}(\hat{\mathbf{q}}))], \end{aligned} \quad (3.49)$$

$$\begin{aligned} \hat{\mathbf{A}} = \frac{\partial \hat{\mathbf{f}}}{\partial \hat{\mathbf{q}}} &= \frac{\partial f}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \hat{\mathbf{q}}} + \frac{\partial f}{\partial \mathbf{f}} \frac{\partial \mathbf{f}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \hat{\mathbf{q}}} + \frac{\partial f}{\partial \mathbf{g}} \frac{\partial \mathbf{g}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \hat{\mathbf{q}}} \\ &= \frac{\xi_t}{J} J \mathbf{I} + \frac{\xi_x}{J} \frac{\partial \mathbf{f}}{\partial \mathbf{q}} J + \frac{\xi_y}{J} \frac{\partial \mathbf{g}}{\partial \mathbf{q}} J \\ &= \xi_t \mathbf{I} + \xi_x \frac{\partial \mathbf{f}}{\partial \mathbf{q}} + \xi_y \frac{\partial \mathbf{g}}{\partial \mathbf{q}}, \end{aligned} \quad (3.50)$$

and similarly

$$\begin{aligned} \hat{\mathbf{g}} &= \frac{\eta_t \mathbf{q} + \eta_x \mathbf{f} + \eta_y \mathbf{g}}{J} \\ &= g[\mathbf{q}(\hat{\mathbf{q}}), \mathbf{f}(\mathbf{q}(\hat{\mathbf{q}})), \mathbf{g}(\mathbf{q}(\hat{\mathbf{q}}))], \end{aligned} \quad (3.51)$$

$$\begin{aligned} \hat{\mathbf{B}} = \frac{\partial \hat{\mathbf{g}}}{\partial \hat{\mathbf{q}}} &= \frac{\partial g}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \hat{\mathbf{q}}} + \frac{\partial g}{\partial \mathbf{f}} \frac{\partial \mathbf{f}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \hat{\mathbf{q}}} + \frac{\partial g}{\partial \mathbf{g}} \frac{\partial \mathbf{g}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \hat{\mathbf{q}}} \\ &= \frac{\eta_t}{J} J \mathbf{I} + \frac{\eta_x}{J} \frac{\partial \mathbf{f}}{\partial \mathbf{q}} J + \frac{\eta_y}{J} \frac{\partial \mathbf{g}}{\partial \mathbf{q}} J \\ &= \eta_t \mathbf{I} + \eta_x \frac{\partial \mathbf{f}}{\partial \mathbf{q}} + \eta_y \frac{\partial \mathbf{g}}{\partial \mathbf{q}}, \end{aligned} \quad (3.52)$$

where \mathbf{I} is the identity matrix. Carrying out the operations in Eqs. (3.50) and (3.52), and substituting the definitions of U^c and V^c yields

$$\hat{\mathbf{A}} = \begin{bmatrix} \xi_t & \xi_x \\ \xi_x \frac{\gamma-1}{2}(u^2+v^2) - u(U^c - \xi_t) & (2-\gamma)\xi_x u + U^c \\ \xi_y \frac{\gamma-1}{2}(u^2+v^2) - v(U^c - \xi_t) & \xi_x v + (1-\gamma)\xi_y u \\ (U^c - \xi_t)[\frac{1}{2}(\gamma-1)(u^2+v^2) - H] & \xi_x H + (1-\gamma)u(U^c - \xi_t) \\ -(U^c - \xi_t)Y_2 & \xi_x Y_2 \end{bmatrix}$$

$$\hat{\mathbf{B}} = \begin{bmatrix} \xi_y & 0 & 0 \\ \xi_y u + (1-\gamma)\xi_x v & \xi_x(\gamma-1) & \xi_x(\gamma-1)q \\ (2-\gamma)\xi_y v + U^c & \xi_y(\gamma-1) & \xi_y(\gamma-1)q \\ \xi_y H + (1-\gamma)v(U^c - \xi_t) & \xi_t + \gamma(U^c - \xi_t) & (U^c - \xi_t)(\gamma-1)q \\ \xi_y Y_2 & 0 & U^c \end{bmatrix},$$

$$\hat{\mathbf{B}} = \begin{bmatrix} \eta_t & \eta_x \\ \eta_x \frac{\gamma-1}{2}(u^2 + v^2) - u(V^c - \eta_t) & (2-\gamma)\eta_x u + V^c \\ \eta_y \frac{\gamma-1}{2}(u^2 + v^2) - v(V^c - \eta_t) & \eta_x v + (1-\gamma)\eta_y u \\ (V^c - \eta_t)[\frac{1}{2}(\gamma-1)(u^2 + v^2) - H] & \eta_x H + (1-\gamma)u(V^c - \eta_t) \\ -(V^c - \eta_t)Y_2 & \eta_x Y_2 \end{bmatrix}.$$

$$\hat{\mathbf{B}} = \begin{bmatrix} \eta_y & 0 & 0 \\ \eta_y u + (1-\gamma)\eta_x v & \eta_x(\gamma-1) & \eta_x(\gamma-1)q \\ (2-\gamma)\eta_y v + V^c & \eta_y(\gamma-1) & \eta_y(\gamma-1)q \\ \eta_y H + (1-\gamma)v(V^c - \eta_t) & \eta_t + \gamma(V^c - \eta_t) & (V^c - \eta_t)(\gamma-1)q \\ \eta_y Y_2 & 0 & V^c \end{bmatrix}.$$

The right eigenvectors and eigenvalues of $\hat{\mathbf{A}}$ are obtained from the eigenvalue problem:

$$\hat{\mathbf{A}}\mathbf{R} = \mathbf{R}\boldsymbol{\lambda}, \quad (3.53)$$

where $\boldsymbol{\lambda}$ is the diagonal matrix of eigenvalues, and $\mathbf{R} = [\mathbf{r}_1|\mathbf{r}_2|\mathbf{r}_3|\mathbf{r}_4|\mathbf{r}_5]$ is the matrix of right eigenvectors ($\mathbf{r}_1, \mathbf{r}_2 \dots$ are columns containing the right eigenvectors of $\hat{\mathbf{A}}$). Since only fixed computational grids were used in the numerical studies, $\xi_t = \eta_t = 0$. Using standard methods the eigenvalues of $\hat{\mathbf{A}}$ were determined to be

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 & 0 \\ 0 & 0 & 0 & \lambda_4 & 0 \\ 0 & 0 & 0 & 0 & \lambda_5 \end{bmatrix} = \begin{bmatrix} U^c & 0 & 0 & 0 & 0 \\ 0 & U^c & 0 & 0 & 0 \\ 0 & 0 & U^c - c\sqrt{\xi_x^2 + \xi_y^2} & 0 & 0 \\ 0 & 0 & 0 & U^c + c\sqrt{\xi_x^2 + \xi_y^2} & 0 \\ 0 & 0 & 0 & 0 & U^c \end{bmatrix} \quad (3.54)$$

and the corresponding matrix of right eigenvectors is

$$\mathbf{R} = \begin{bmatrix} \xi_x & 0 & 1 & 1 & 0 \\ U^c & \xi_y & u - \frac{\xi_x}{\sqrt{\xi_x^2 + \xi_y^2}}c & u + \frac{\xi_x}{\sqrt{\xi_x^2 + \xi_y^2}}c & 0 \\ 0 & -\xi_x & v - \frac{\xi_y}{\sqrt{\xi_x^2 + \xi_y^2}}c & v + \frac{\xi_y}{\sqrt{\xi_x^2 + \xi_y^2}}c & 0 \\ (u^2 - v^2)\frac{\xi_x}{2} + \xi_y uv & \xi_y u - \xi_x v & H - \frac{c}{\sqrt{\xi_x^2 + \xi_y^2}}U^c & H + \frac{c}{\sqrt{\xi_x^2 + \xi_y^2}}U^c & -q \\ 0 & 0 & Y_2 & Y_2 & 1 \end{bmatrix}. \quad (3.55)$$

The left eigenvector problem for $\hat{\mathbf{A}}$ is

$$\mathbf{T}\hat{\mathbf{A}} = \lambda\mathbf{T}, \quad (3.56)$$

where \mathbf{T} is the matrix of left eigenvectors (the rows of \mathbf{T} are the left eigenvectors of $\hat{\mathbf{A}}$). Solving Eqs. (3.53) and (3.56) for $\hat{\mathbf{A}}$

$$\hat{\mathbf{A}} = \mathbf{R}\lambda\mathbf{R}^{-1}, \quad (3.57)$$

$$\hat{\mathbf{A}} = \mathbf{T}^{-1}\lambda\mathbf{T}, \quad (3.58)$$

it is apparent that $\mathbf{T} = \mathbf{R}^{-1}$. Thus, the left eigenvector matrix is simply:

$$\mathbf{R}^{-1} = \begin{bmatrix} \frac{1}{\xi_x} + \frac{1-\gamma}{2\xi_x c^2}(u^2 + v^2) & \frac{\gamma-1}{\xi_x c^2}u \\ \frac{\gamma-1}{2\xi_x c^2}v(u^2 + v^2) - \frac{\xi_y}{\xi_x(\xi_x^2 + \xi_y^2)}U^c & \frac{1-\gamma}{\xi_x c^2}uv + \frac{\xi_y}{\xi_x^2 + \xi_y^2} \\ \frac{\gamma-1}{4c^2}(u^2 + v^2) + \frac{U^c}{2c\sqrt{\xi_x^2 + \xi_y^2}} & \frac{1-\gamma}{2c^2}u - \frac{\xi_x}{2c\sqrt{\xi_x^2 + \xi_y^2}} \\ \frac{\gamma-1}{4c^2}(u^2 + v^2) - \frac{U^c}{2c\sqrt{\xi_x^2 + \xi_y^2}} & \frac{1-\gamma}{2c^2}u + \frac{\xi_x}{2c\sqrt{\xi_x^2 + \xi_y^2}} \\ \frac{1-\gamma}{2c^2}(u^2 + v^2)Y_2 & \frac{\gamma-1}{c^2}uY_2 \\ \frac{\gamma-1}{\xi_x c^2}v & \frac{1-\gamma}{\xi_x c^2} & \frac{1-\gamma}{\xi_x c^2}q \\ \frac{1-\gamma}{\xi_x c^2}v^2 - \frac{\xi_x}{\xi_x^2 + \xi_y^2} & \frac{\gamma-1}{\xi_x c^2}v & \frac{\gamma-1}{\xi_x c^2}vq \\ \frac{1-\gamma}{2c^2}v - \frac{\xi_y}{2c\sqrt{\xi_x^2 + \xi_y^2}} & \frac{\gamma-1}{2c^2} & \frac{\gamma-1}{2c^2}q \\ \frac{1-\gamma}{2c^2}v + \frac{\xi_y}{2c\sqrt{\xi_x^2 + \xi_y^2}} & \frac{\gamma-1}{2c^2} & \frac{\gamma-1}{2c^2}q \\ \frac{\gamma-1}{c^2}vY_2 & \frac{1-\gamma}{c^2}Y_2 & 1 + \frac{1-\gamma}{c^2}qY_2 \end{bmatrix} \quad (3.59)$$

The eigenvalues, μ , right eigenvector matrix, \mathbf{U} , and left eigenvector matrix, \mathbf{U}^{-1} , of $\hat{\mathbf{B}}$ are

$$\mu = \begin{bmatrix} V^c & 0 & 0 & 0 & 0 \\ 0 & V^c & 0 & 0 & 0 \\ 0 & 0 & V^c - c\sqrt{\xi_x^2 + \xi_y^2} & 0 & 0 \\ 0 & 0 & 0 & V^c + c\sqrt{\xi_x^2 + \xi_y^2} & 0 \\ 0 & 0 & 0 & 0 & V^c \end{bmatrix} \quad (3.60)$$

$$\mathbf{U} = \begin{bmatrix} \eta_y & 0 & 1 & 1 & 0 \\ 0 & \eta_y & u - \frac{\eta_x}{\sqrt{\eta_x^2 + \eta_y^2}}c & u + \frac{\eta_x}{\sqrt{\eta_x^2 + \eta_y^2}}c & 0 \\ V^c & -\eta_x & v - \frac{\eta_y}{\sqrt{\eta_x^2 + \eta_y^2}}c & v + \frac{\eta_y}{\sqrt{\eta_x^2 + \eta_y^2}}c & 0 \\ (v^2 - u^2)\frac{\eta_y}{2} + \eta_x uv & \eta_y u - \eta_x v & H - \frac{c}{\sqrt{\eta_x^2 + \eta_y^2}}V^c & H + \frac{c}{\sqrt{\eta_x^2 + \eta_y^2}}V^c & -q \\ 0 & 0 & Y_2 & Y_2 & 1 \end{bmatrix}, \quad (3.61)$$

$$\mathbf{U}^{-1} = \begin{bmatrix} \frac{1}{\eta_y} + \frac{1-\gamma}{2\eta_y c^2}(u^2 + v^2) & & \frac{\gamma-1}{\eta_y c^2}u \\ \frac{1-\gamma}{2\eta_y c^2}u(u^2 + v^2) + \frac{\eta_x}{\eta_y(\eta_x^2 + \eta_y^2)}V^c & \frac{\gamma-1}{\eta_y c^2}u^2 + \frac{\eta_y}{\eta_x^2 + \eta_y^2} & \\ \frac{\gamma-1}{4c^2}(u^2 + v^2) + \frac{V^c}{2c\sqrt{\eta_x^2 + \eta_y^2}} & \frac{1-\gamma}{2c^2}u - \frac{\eta_x}{2c\sqrt{\eta_x^2 + \eta_y^2}} & \\ \frac{\gamma-1}{4c^2}(u^2 + v^2) - \frac{V^c}{2c\sqrt{\eta_x^2 + \eta_y^2}} & \frac{1-\gamma}{2c^2}u + \frac{\eta_x}{2c\sqrt{\eta_x^2 + \eta_y^2}} & \\ \frac{1-\gamma}{2c^2}(u^2 + v^2)Y_2 & & \frac{\gamma-1}{c^2}uY_2 \\ \frac{\gamma-1}{\eta_y c^2}v & \frac{1-\gamma}{\eta_y c^2} & \frac{1-\gamma}{\eta_y c^2}q \\ \frac{\gamma-1}{\eta_y c^2}uv - \frac{\eta_x}{\eta_x^2 + \eta_y^2} & \frac{1-\gamma}{\eta_y c^2}u & \frac{1-\gamma}{\eta_y c^2}uq \\ \frac{1-\gamma}{2c^2}v - \frac{\eta_y}{2c\sqrt{\eta_x^2 + \eta_y^2}} & \frac{\gamma-1}{2c^2} & \frac{\gamma-1}{2c^2}q \\ \frac{1-\gamma}{2c^2}v + \frac{\eta_y}{2c\sqrt{\eta_x^2 + \eta_y^2}} & \frac{\gamma-1}{2c^2} & \frac{\gamma-1}{2c^2}q \\ \frac{\gamma-1}{c^2}vY_2 & \frac{1-\gamma}{c^2}Y_2 & 1 + \frac{1-\gamma}{c^2}qY_2 \end{bmatrix}. \quad (3.62)$$

Even though the flux Jacobian arrays $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$ really only differ in terms of their metric quantities, slightly different forms for the eigenvector arrays \mathbf{R} and \mathbf{U} were chosen. Had the same form been used for both eigenvector arrays, there would have been singularities in one of the inverse eigenvector arrays. In particular, one of the arrays would have had either ξ_y or η_x in the denominator of a term, leading to a singularity when one of those quantities became zero. For example, ξ_y and η_x would become zero on a Cartesian mesh. On the other hand, ξ_x and η_y will not normally become zero, and so \mathbf{R}^{-1} and \mathbf{U}^{-1} have no mesh related singularities.

Chapter 4

FUNDAMENTAL DETONATION SOLUTIONS

This chapter will present three steady analytical solutions to the model equations: the one-dimensional ZND solution, the two-dimensional straight shock detonation solution for supersonic flow past a curved wall (the two-dimensional analog of the ZND solution), and the two-dimensional detonation solution for supersonic flow past a straight wall. These solutions were used to verify the numerical methods used in the research, as well as to provide initial conditions for some of the numerical studies.

4.1 One-dimensional Steady Solution

Figure 4.1 illustrates the steady ZND detonation problem. At $t = 0$ a detonation wave is initiated in a tube of quiescent premixed fuel and oxidizer by striking the

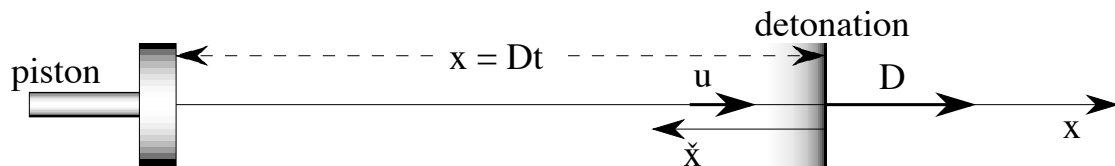


Figure 4.1: Illustration of the steady detonation problem and the Galilean coordinate transformation.

mixture with a sharp blow from a piston (for example). After an initial transient, the detonation proceeds into the tube at a steady velocity D , and induces the fluid

following it to move with velocity u . The flow is modeled by the Euler equations and is one-dimensional, with shocks treated as discontinuities over which no chemical reaction occurs. The reaction is assumed to be initiated by the passage of the shock, and then to proceed to completion at a finite rate. If we consider only one dimension, the model presented in Eqs. (3.18–3.22) is a ZND model:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) = 0, \quad (4.1)$$

$$\frac{\partial}{\partial t}(\rho u) + \frac{\partial}{\partial x}(\rho u^2 + p) = 0, \quad (4.2)$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x}(\rho u H) = 0, \quad (4.3)$$

$$\frac{\partial}{\partial t}(\rho Y_2) + \frac{\partial}{\partial x}(\rho u Y_2) = \rho(1 - Y_2)e^{-\Theta\rho/p}. \quad (4.4)$$

Since the detonation is propagating at the steady velocity D , the following Galilean coordinate transformation will place the equations of motion in the detonation frame of reference (see Fig. 4.1):

$$\check{x} = Dt - x \quad \check{t} = t. \quad (4.5)$$

The relation between the velocity in the new frame and the velocity in the old frame is obtained by taking the derivative of Eq. (4.5)

$$\check{u} = \frac{d\check{x}}{d\check{t}} = D - \frac{dx}{dt} = D - u. \quad (4.6)$$

The equations are transformed in the much the same manner as they were in the previous chapter: derivatives in the old frame of reference are replaced with equivalent derivatives in the new frame of reference using the chain rule. The detonation is assumed steady in the new frame of reference, thus $\partial/\partial\check{t} \equiv 0$.

$$\begin{aligned} \frac{\partial}{\partial t} &= \frac{\partial \check{x}}{\partial t} \frac{\partial}{\partial \check{x}} + \frac{\partial \check{t}}{\partial t} \frac{\partial}{\partial \check{t}} = D \frac{\partial}{\partial \check{x}} + \frac{\partial}{\partial \check{t}} = D \frac{d}{d\check{x}} \\ \frac{\partial}{\partial x} &= \frac{\partial \check{x}}{\partial x} \frac{\partial}{\partial \check{x}} + \frac{\partial \check{t}}{\partial x} \frac{\partial}{\partial \check{t}} = -\frac{d}{d\check{x}} \end{aligned} \quad (4.7)$$

Applying this transformation to Eqs. (4.1–4.4) and using Eq. (4.6) results in (after simplifying)

$$\frac{d}{d\check{x}}(\rho\check{u}) = 0, \quad (4.8)$$

$$\frac{d}{d\check{x}}(\rho\check{u}^2 + p) = 0, \quad (4.9)$$

$$\frac{d}{d\check{x}}(\rho\check{u}\check{H}) = 0, \quad (4.10)$$

$$\frac{d}{d\check{x}}(\rho\check{u}Y_2) = \rho(1 - Y_2)e^{-\Theta\rho/p}, \quad (4.11)$$

where $\check{H} = \check{E} + p/\rho = e + \check{u}^2/2 + p/\rho$.

Figure 4.2 illustrates the problem in the new coordinate system. The detonation sees an incoming flow with known initial conditions ρ_0, p_0, e_0 and velocity D . The

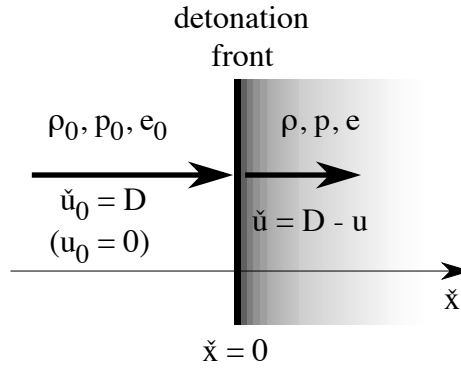


Figure 4.2: Diagram of detonation in the \check{x} coordinate system.

flow variables change through the reaction zone, eventually reaching a fixed final state. Using the initial conditions and the definition for \check{H} , Eqs. (4.8–4.10) can be integrated directly to give

$$\rho\check{u} = \rho_0 D, \quad (4.12)$$

$$\rho\check{u}^2 + p = \rho_0 D^2 + p_0, \quad (4.13)$$

$$\rho\check{u} \left(e + \frac{\check{u}^2}{2} + \frac{p}{\rho} \right) = \rho_0 D \left(e_0 + \frac{D^2}{2} + \frac{p_0}{\rho_0} \right). \quad (4.14)$$

Equation (4.12) can be used in Eq. (4.13) to eliminate \check{u}

$$\mathcal{R} = \rho_0^2 D^2 - \frac{p - p_0}{\varphi_0 - \varphi} = 0, \quad (4.15)$$

where $\varphi = 1/\rho$ is the specific volume. This is the equation for the Rayleigh line \mathcal{R} . Similarly, the equation for a Hugoniot curve \mathcal{H} can be obtained by using Eqs. (4.12) and (4.13) to eliminate \tilde{u} and D from Eq. (4.14)

$$\mathcal{H} = e - e_0 - \frac{(p + p_0)(\varphi_0 - \varphi)}{2} = 0. \quad (4.16)$$

The Hugoniot equation can be written in terms of p , φ , and Y_2 by using the equation of state to eliminate e from Eq. (4.16). The equation of state is Eq. (3.23) written in terms of the specific volume

$$e = \frac{p\varphi}{\gamma - 1} - Y_2q. \quad (4.17)$$

The Hugoniot equation then becomes

$$\mathcal{H} = \frac{p\varphi - p_0\varphi_0}{\gamma - 1} - \frac{1}{2}(p + p_0)(\varphi_0 - \varphi) - Y_2q = 0, \quad (4.18)$$

where $Y_{20} = 0$ (there are no products in the quiescent mixture). This equation may be written in the form of rectangular hyperbola [27]

$$\left(\frac{p}{p_0} + \mu^2\right) \left(\frac{\varphi}{\varphi_0} - \mu^2\right) = 1 - \mu^4 + 2\mu^2 \frac{Y_2q}{p_0v_0}, \quad (4.19)$$

centered on the point $\varphi/\varphi_0 = \mu^2$, $p/p_0 = -\mu^2$, where $\mu^2 = (\gamma - 1)/(\gamma + 1)$.

The intersection of the Rayleigh line with the Hugoniot curve determines the state (p, φ) for a given detonation velocity D . This is illustrated in Fig. 4.3. Three Rayleigh lines are shown for different values of the detonation velocity D ; increasing D results in increasingly negatively sloped lines. Three Hugoniot curves are shown for increasing values of the product mass fraction: $Y_2 = 0$ is no reaction (\mathcal{H}_0), $Y_2 = 0.5$ is a half-complete reaction ($\mathcal{H}_{1/2}$), and $Y_2 = 1$ is complete reaction (\mathcal{H}_1). The initial state is denoted \mathbf{O} where all three Rayleigh lines and the $Y_2 = 0$ Hugoniot curve intersect. The detonation wave first shocks the flow to state \mathbf{N} , the Neumann point, which is the leftmost intersection of the Rayleigh lines with the zero mass fraction Hugoniot curve. As expected, detonations with larger velocities shock the flow to

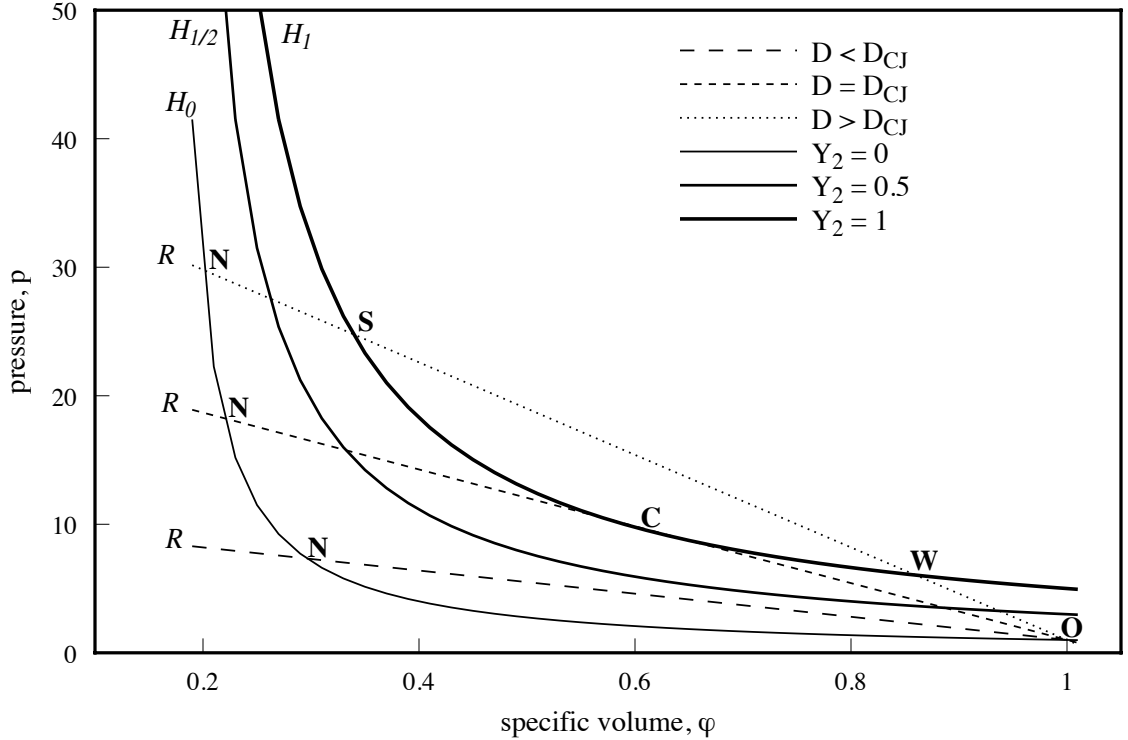


Figure 4.3: Rayleigh lines and Hugoniot curves for ZND detonation.

higher pressures and lower specific volumes. Following the shocked state the flow reacts, proceeding along the Rayleigh lines through Hugoniot curves of increasing Y_2 . The final constant end state **S** is achieved only for detonation velocities above a certain value, denoted the Chapman-Jouguet velocity D_{CJ} , when the Rayleigh lines intersect the Hugoniot curve of complete reaction ($Y_2 = 1$).

Examining the curves more carefully, it is apparent that below a certain velocity there is no solution; the Rayleigh line does not intersect the end state Hugoniot curve H_1 . At D_{CJ} there is one unique intersection at **C**. Above the CJ velocity there are two intersections; the upper intersection is called the strong solution **S**, and the lower intersection **W** is called the weak solution. The weak solution is discarded on the grounds that there is no path from the shocked state to point **W** for this model. Thus, the CJ velocity is the minimum speed at which the detonation with a lead shock can travel.

It can also be argued to be the speed of an unsupported, freely travelling detonation wave. It can be shown that the flow velocity with respect to the front at **S** is subsonic $\check{u} = D - u < c$, at the CJ point is sonic $\check{u} = D - u = c$, and at **W** is supersonic $\check{u} = D - u > c$. Because of the subsonic condition at **S**, the strong solution requires support from the piston depicted in Fig. 4.1. If the piston is slowed below the end state velocity at **S**, rarefactions travelling at $u + c$ will overtake the detonation and weaken it. The detonation velocity will then slow until it reaches $D = u + c = D_{CJ}$, the point at which the detonation speed matches the speed of following disturbances.

The complete solution ZND solution requires that all of Eqs. (4.8–4.11) be considered. Solution of Eq. (4.11) in general cannot be done analytically, and requires that φ , \check{u} , and p be written only in terms of Y_2 . Equations (4.12–4.14, 4.17) can be manipulated to give

$$\check{u} = \frac{-\gamma C_2 + \sqrt{[\gamma C_2 - (1 + \gamma)C_1\check{u}_s]^2 + 2(1 + \gamma)(1 - \gamma)C_1^2 q Y_2}}{-(1 + \gamma)C_1}, \quad (4.20)$$

$$\varphi = \frac{-\gamma C_2 + \sqrt{[\gamma C_2 - (1 + \gamma)C_1\check{u}_s]^2 + 2(1 + \gamma)(1 - \gamma)C_1^2 q Y_2}}{-(1 + \gamma)C_1^2}, \quad (4.21)$$

$$p = C_2 + \frac{-\gamma C_2 + \sqrt{[\gamma C_2 - (1 + \gamma)C_1\check{u}_s]^2 + 2(1 + \gamma)(1 - \gamma)C_1^2 q Y_2}}{-(1 + \gamma)}, \quad (4.22)$$

where $C_1 = \rho_0 D$ and $C_2 = \rho_0 D^2 + p_0$ are constants. Only the positive roots are shown because the negative roots correspond to an unshocked initial state. Equations (4.12–4.14) can also be manipulated to give the shocked velocity \check{u}_s

$$\frac{\check{u}_s}{\check{u}_0} = \frac{2 + (\gamma - 1)\mathcal{M}_0^2}{(\gamma + 1)\mathcal{M}_0^2}, \quad (4.23)$$

where $\mathcal{M}_0 = D/\sqrt{\gamma p_0 \varphi_0}$ is the Mach number of the travelling detonation. Note that the CJ condition can be determined from Eq. (4.21), for example. At the CJ point ($Y_2 = 1$), φ must be single-valued and real. This occurs when the radical in Eq. (4.21)

is zero:

$$[\gamma C_2 - (1 + \gamma)C_1 \check{u}_s]^2 + 2(1 + \gamma)(1 - \gamma)C_1^2 q = 0. \quad (4.24)$$

The nondimensional constants C_1 and C_2 can be written more simply by noting the preshocked state has been used to nondimensionalize the flow variables:

$$\rho_0 = \frac{\tilde{\rho}_0}{\tilde{\rho}_0} = 1, \quad (4.25)$$

$$D = \frac{\tilde{D}}{\mathcal{M}_0 \sqrt{\tilde{p}_0 / \tilde{\rho}_0}} = \frac{\mathcal{M}_0 \tilde{c}}{\mathcal{M}_0 \sqrt{\tilde{p}_0 / \tilde{\rho}_0}} = \frac{\sqrt{\gamma \tilde{p}_0 / \tilde{\rho}_0}}{\sqrt{\tilde{p}_0 / \tilde{\rho}_0}} = \sqrt{\gamma}, \quad (4.26)$$

$$p_0 = \frac{\tilde{p}_0}{\mathcal{M}_0^2 \tilde{p}_0} = \frac{1}{\mathcal{M}_0^2}. \quad (4.27)$$

This results in $C_1 = \sqrt{\gamma}$ and $C_2 = \gamma + 1/\mathcal{M}_0^2$. Substituting for C_1 , C_2 , and \check{u}_s in Eq. (4.24) results in (note $\check{u}_0 = D = \sqrt{\gamma}$)

$$\left[\frac{\gamma}{\mathcal{M}_0^2} + \gamma^2 - \frac{\gamma[2 + (\gamma - 1)\mathcal{M}_0^2]}{\mathcal{M}_0^2} \right]^2 + 2\gamma(1 - \gamma^2)q = 0. \quad (4.28)$$

The only unknown in this equation is \mathcal{M}_0 , the Mach number of the detonation at the CJ condition; this will be denoted \mathcal{M}_{0CJ} . Before this equation can be solved, however, q must be rewritten so as not to depend on \mathcal{M}_0 . A commonly chosen nondimensional form is $\bar{q} = \tilde{q}\tilde{\rho}_0/\tilde{p}_0 = \mathcal{M}_0^2 q$. Equation (4.28) then becomes

$$\left[\frac{\gamma}{\mathcal{M}_{0CJ}^2} + \gamma^2 - \frac{\gamma[2 + (\gamma - 1)\mathcal{M}_{0CJ}^2]}{\mathcal{M}_{0CJ}^2} \right]^2 + \frac{2\gamma(1 - \gamma^2)\bar{q}}{\mathcal{M}_{0CJ}^2} = 0, \quad (4.29)$$

which can be solved to give

$$\mathcal{M}_{0CJ}^2 = \frac{\gamma^2 + \bar{q}(\gamma^3 - \gamma) + \sqrt{[\gamma^2 + \bar{q}(\gamma^3 - \gamma)]^2 - \gamma^4}}{\gamma^2}. \quad (4.30)$$

Thus, the CJ Mach number depends only upon the fluid properties, initial fluid conditions, and the heat release due to the chemical reaction.

Substituting Eqs. (4.20–4.23) into Eq. (4.11) results in an ordinary differential equation in Y_2 and \check{x} . This can be integrated numerically using standard techniques to obtain $Y_2(\check{x})$, which then determines $\check{u}(\check{x})$, $\varphi(\check{x})$, and $p(\check{x})$. Figure 4.4 shows the

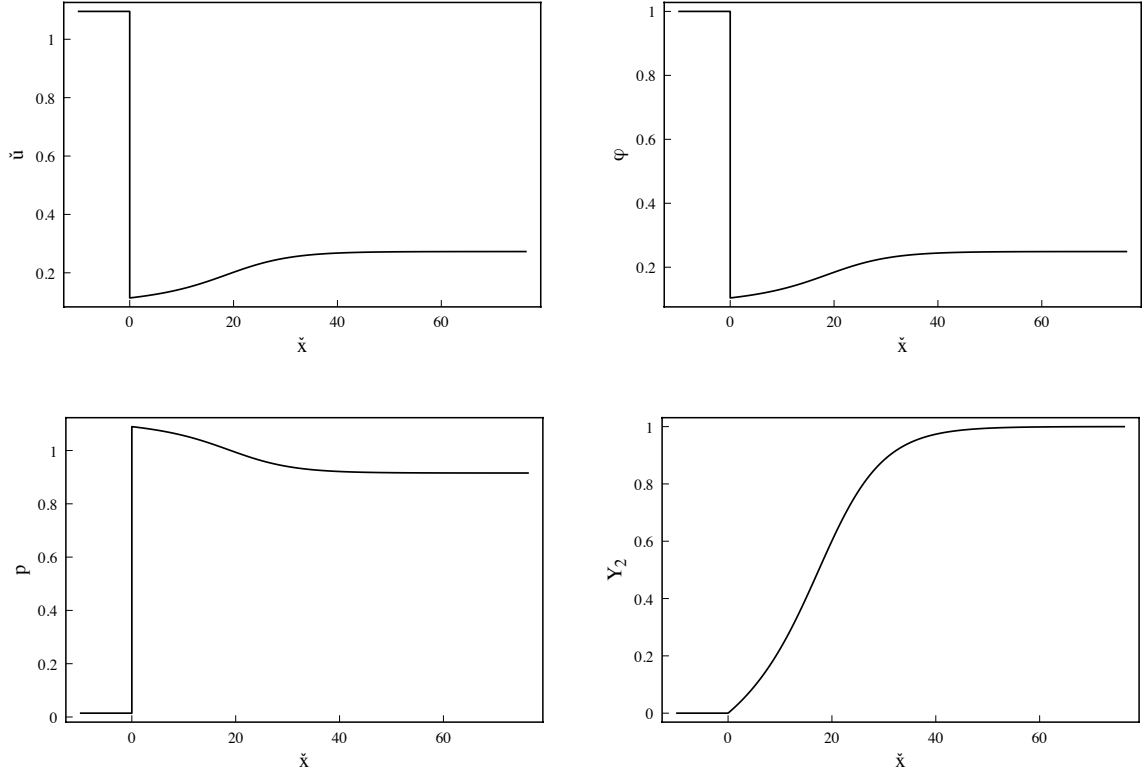


Figure 4.4: Overdriven ZND detonation solution ($f = 1.8$, $q = 0.719$, $\Theta = 0.719$, $\gamma = 1.2$).

solution for a strong ZND detonation when $f = 1.8$ (recall the definition of overdrive from the beginning of Ch. 2), $q = 0.719$, $\Theta = 0.719$, and $\gamma = 1.2$. Examining Fig. 4.4 more closely, it can be seen that \tilde{u} and φ are initially shocked from their freestream values to a minimum value, and then increase through the reaction zone. The pressure acts in the opposite way, peaking at the detonation front and decreasing through the reaction zone. The product mass fraction curve shows there is no induction zone; Y_2 immediately begins increasing following the detonation front. As Y_2 asymptotically approaches 1, \tilde{u} , φ , and p asymptotically approach constant values predicted by the jump analysis.

Figure 4.5 illustrates a CJ detonation for the same q , Θ , and γ . Comparing to Fig. 4.4 it is apparent that the reaction zone is much longer than in the overdriven case; the weaker lead shock of the CJ detonation results in the reaction rate being

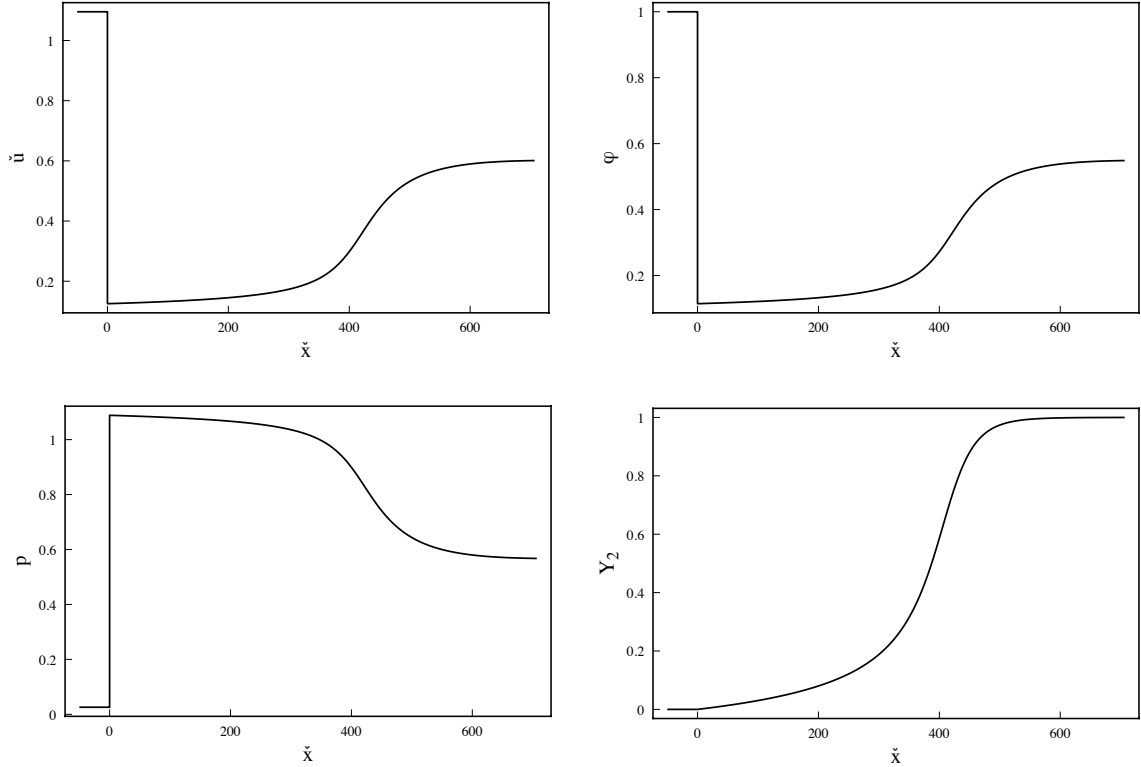


Figure 4.5: Chapman-Jouguet detonation solution ($f = 1.0$, $q = 0.719$, $\Theta = 0.719$, $\gamma = 1.2$).

much lower initially. There is an induction zone and then a narrow region in which the reaction proceeds very quickly to completion. The other flow variables follow Y_2 , remaining close to the shocked state and then undergoing a rapid transition to their final values.

4.2 Two-dimensional Steady Solutions

4.2.1 Straight Shock, Curved Wall

The straight shock, curved wall detonation solution is the two-dimensional analog of the one-dimensional ZND detonation. In this case consider steady flow over a curved wedge. The wall is assumed to be curved in such a fashion that the resulting oblique detonation is straight, as shown in Fig. 4.6. The exact shape of the wedge is initially unknown; it will be determined by the solution of the equations of motion.

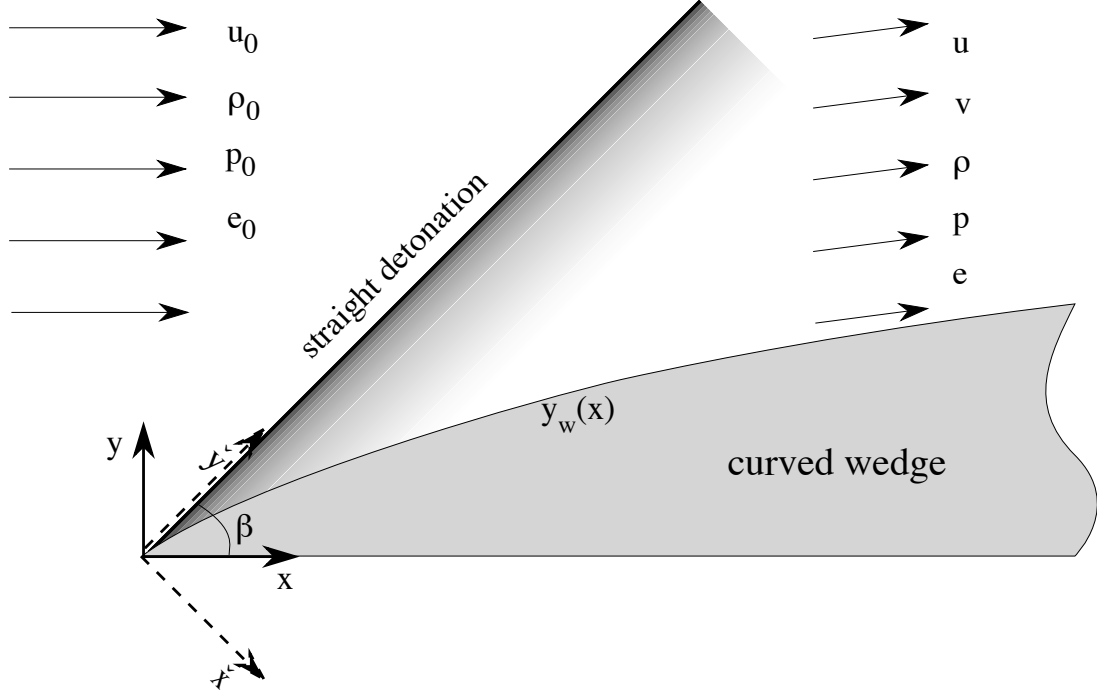


Figure 4.6: Schematic of straight shock, curved wall detonation flowfield.

Once again Eqs. (3.18–3.22) are appropriate, except the time derivatives are zero; the detonation is considered to be steady in the fixed frame attached to the wedge. In this case it is useful to consider the equations in the coordinate system (\check{x}, \check{y}) oriented with the detonation as indicated in Fig. 4.6:

$$\frac{\partial}{\partial \check{x}}(\rho \check{u}) + \frac{\partial}{\partial \check{y}}(\rho \check{v}) = 0, \quad (4.31)$$

$$\frac{\partial}{\partial \check{x}}(\rho \check{u}^2 + p) + \frac{\partial}{\partial \check{y}}(\rho \check{u} \check{v}) = 0, \quad (4.32)$$

$$\frac{\partial}{\partial \check{x}}(\rho \check{u} \check{v}) + \frac{\partial}{\partial \check{y}}(\rho \check{v}^2 + p) = 0, \quad (4.33)$$

$$\frac{\partial}{\partial \check{x}}(\rho \check{u} \check{H}) + \frac{\partial}{\partial \check{y}}(\rho \check{v} \check{H}) = 0, \quad (4.34)$$

$$\frac{\partial}{\partial \check{x}}(\rho \check{u} Y_2) + \frac{\partial}{\partial \check{y}}(\rho \check{v} Y_2) = \rho(1 - Y_2)e^{-\Theta\rho/p}. \quad (4.35)$$

The velocities \check{u} and \check{v} are in the \check{x} and \check{y} directions, respectively, and $\check{H} = \check{E} + p/\rho = e + (\check{u}^2 + \check{v}^2)/2 + p/\rho$. The detonation coordinate system and wedge coordinate system

may be related using the detonation angle β :

$$x = \check{x} \sin \beta + \check{y} \cos \beta, \quad (4.36)$$

$$y = -\check{x} \cos \beta + \check{y} \sin \beta. \quad (4.37)$$

The relation between the velocities in the two coordinate systems is found by taking the derivative with respect to time of the coordinate transformation above. The resulting relations are:

$$u = \check{u} \sin \beta + \check{v} \cos \beta, \quad (4.38)$$

$$v = -\check{u} \cos \beta + \check{v} \sin \beta. \quad (4.39)$$

As in the ZND model, the detonation is initiated by a lead shock. In this case the shock is an oblique shock, and it is assumed that there are no changes in the \check{y} direction; thus $\frac{\partial}{\partial \check{y}} \equiv 0$. Equations (4.31–4.35) then become

$$\frac{d}{d\check{x}}(\rho\check{u}) = 0, \quad (4.40)$$

$$\frac{d}{d\check{x}}[\rho\check{u}^2 + p] = 0, \quad (4.41)$$

$$\frac{d}{d\check{x}}(\rho\check{u}\check{v}) = 0, \quad (4.42)$$

$$\frac{d}{d\check{x}}(\rho\check{u}\check{H}) = 0. \quad (4.43)$$

$$\frac{d}{d\check{x}}(\rho\check{u}Y_2) = \rho(1 - Y_2)e^{-\Theta\rho/p}. \quad (4.44)$$

Equations (4.40–4.41, 4.43–4.44) are exactly the same as Eqs. (4.8–4.11) for the ZND detonation, and therefore have the same solution. Equations (4.42, 4.40) are easily solved to give $\rho\check{u}\check{v} = \text{constant}$ and $\rho\check{u} = \text{constant}$. Combining the two it is apparent that

$$\check{v} = \text{constant}. \quad (4.45)$$

Thus, the straight shock, curved wall detonation solution is simply the one-dimensional ZND solution with an added constant \check{v} component. Note that since shock

curvature is the only vorticity production mechanism in an inviscid flow, the flow is irrotational

$$\omega_z = \nabla \times \vec{V} = \frac{\partial \check{v}}{\partial \check{x}} - \frac{\partial \check{u}}{\partial \check{y}} = 0. \quad (4.46)$$

Since there is no flow through a streamline, the wall function $\check{y}_w(\check{x})$ may be obtained by finding any streamline in the flow. The velocity vector must be tangent to the streamlines; therefore, the slope of the streamlines and velocity vector must be equal

$$\frac{d\check{y}_w}{d\check{x}} = \frac{\check{v}}{\check{u}(\check{x})}. \quad (4.47)$$

Taking the origin to be at the wedge tip so $\check{y}_w(0) = 0$, this can be solved to give

$$\check{y}_w(\check{x}) = \check{v} \int_0^{\check{x}} \frac{d\hat{x}}{\check{u}(\hat{x})}. \quad (4.48)$$

Much in the same way as one-dimensional detonations, oblique detonations are typically classified using the results of a Rankine-Hugoniot analysis. Figure 4.7 illustrates the case in which an oblique detonation is considered to be a shock discontinuity with heat release [34, 64]. Just as in oblique shock theory, the equations of motion are reduced to algebraic relations via a Rankine-Hugoniot discontinuity analysis, and then manipulated to give a relation between the detonation wave angle and the supporting wedge angle. This equation contains an extra parameter for the heat release $Y_2 q$, and reduces to the oblique shock relation when $Y_2 = 0$. Unlike the inert state, adding heat to the flow leads to closed curves in which decreasing the wedge angle below a certain minimum value results in increasing wave angles (see Fig. 4.7). Increasing the amount of reaction results in the closed area shrinking toward the upper left corner of the plot. For a given q , detonation angles below a certain angle β_{CJ} are not possible. At $\beta = \beta_{CJ}$ there is one solution, and above β_{CJ} there are two. As in oblique shock theory, there is a strong regime (**S**) in which the Mach number following the detonation (\mathcal{M}_2) is subsonic, and a weak regime (**W**) in which $\mathcal{M}_2 > 1$ but the Mach number normal to the wave (\mathcal{M}_{2n}) is less than unity. Following one-

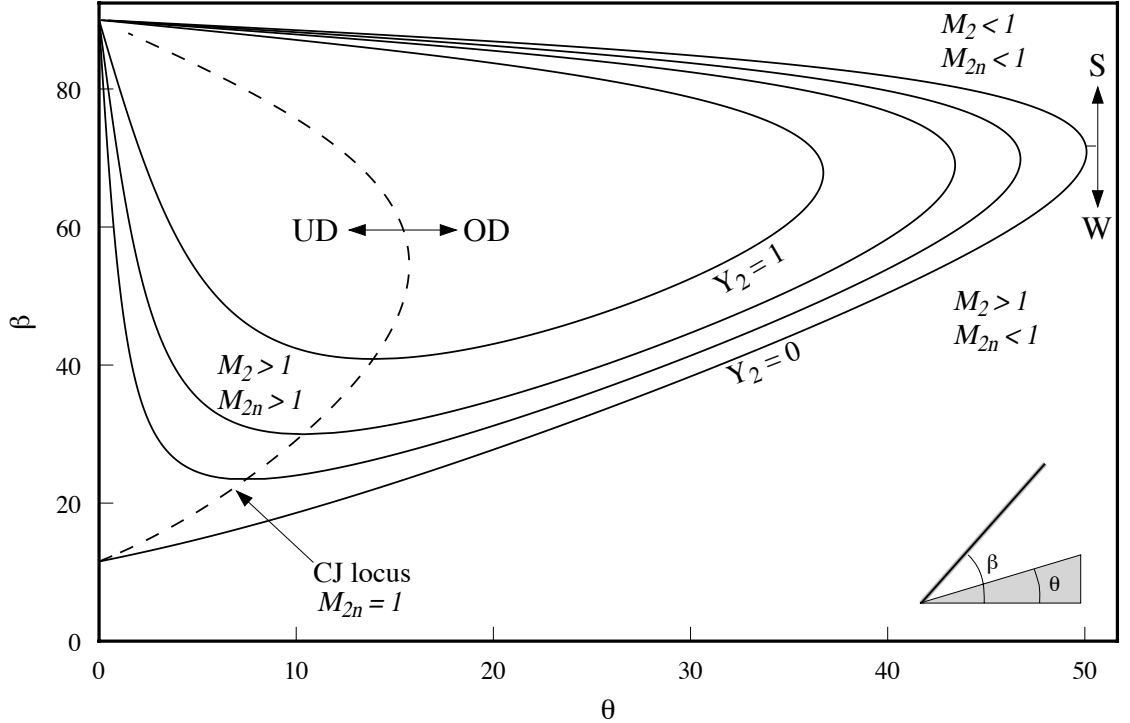


Figure 4.7: Variation of oblique detonation wave angle with wedge angle and heat release ($\gamma = 1.2$, $\mathcal{M}_0 = 5$).

dimensional detonation theory, there is a state when the Mach number normal to the wave (\mathcal{M}_{2n}) is unity. This leads to an underdriven (**UD**) regime where $\mathcal{M}_{2n} > 1$, and an overdriven (**OD**) regime where $\mathcal{M}_{2n} < 1$. Thus, oblique detonations are classified as strong (which is always overdriven), weak overdriven, and weak underdriven. As in the one-dimensional theory, the weak underdriven state is inaccessible from the shocked state for a one-step, irreversible reaction. Recent work [62], however, has demonstrated that when a two-step, exothermic-endothermic scheme is considered, the weak underdriven solution is accessible from the shocked state. Lastly, the straight shock, curved wall solution can be classified according to an overdrive parameter, in this case termed the normal overdrive $f_n = (\mathcal{M}_{0n}/\mathcal{M}_{0CJ})^2$, where \mathcal{M}_{0n} is the component of the freestream Mach number normal to the detonation.

Figure 4.8 shows a weak overdriven straight shock, curved wall detonation solution for $f_n = 1.6$. The plots show contours of constant pressure and product mass

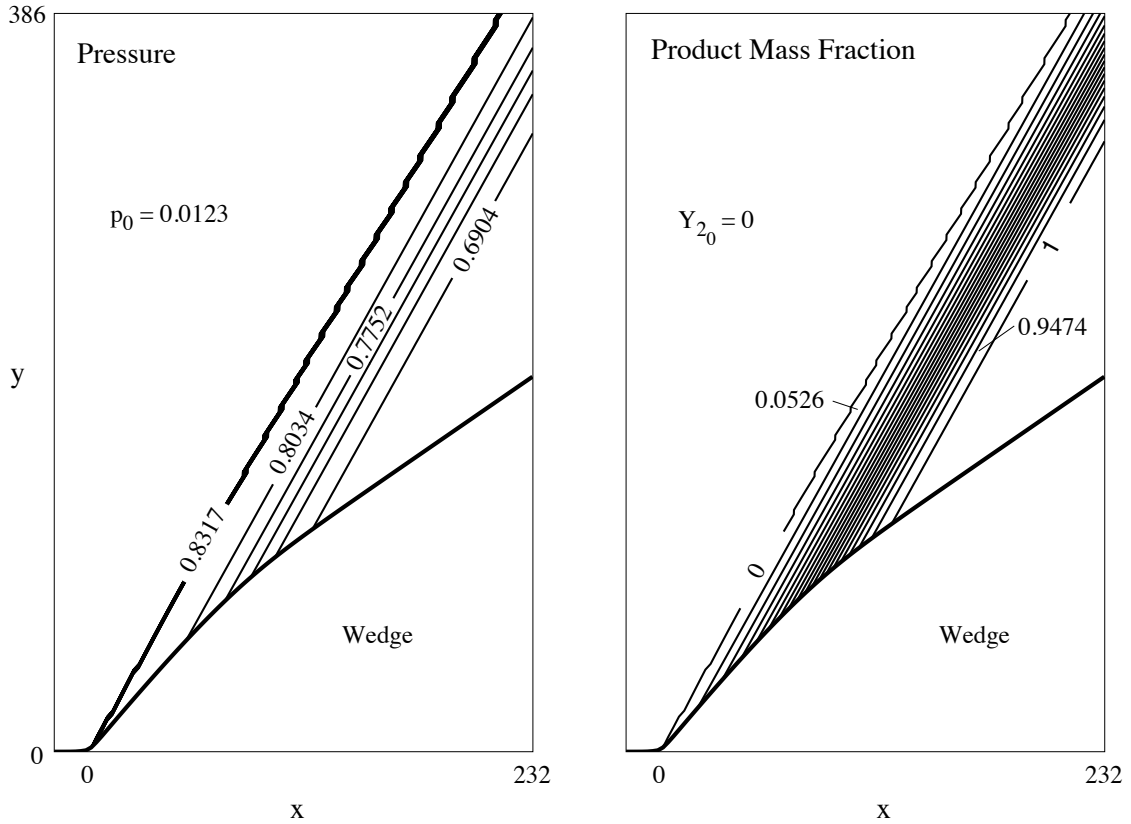


Figure 4.8: Weak, overdriven straight shock, curved wall detonation solutions for pressure and product mass fraction ($f_n = 1.6$, $\mathcal{M}_0 = 9$, $q = 0.719$, $\Theta = 0.719$, $\gamma = 1.2$).

fraction, as well as the wall shape. As expected the solution contours are straight (any jaggedness in the contours is an artifact of the discretization necessary for plotting), while the wall curves. The detonation angle is $\beta = 61^\circ$, while the wall angle is a maximum at the apex ($\theta_{max} = 50^\circ$) and decreases to a minimum constant value at the end of the reaction zone ($\theta_{min} = 35^\circ$). Figure 4.9 shows a strong solution with the same normal overdrive value of $f_n = 1.6$. In this case the detonation angle is greater at $\beta = 79^\circ$, the initial wall angle is the same to this precision, $\theta_{max} = 50^\circ$, and the final wall angle is less at $\theta_{min} = 24^\circ$.

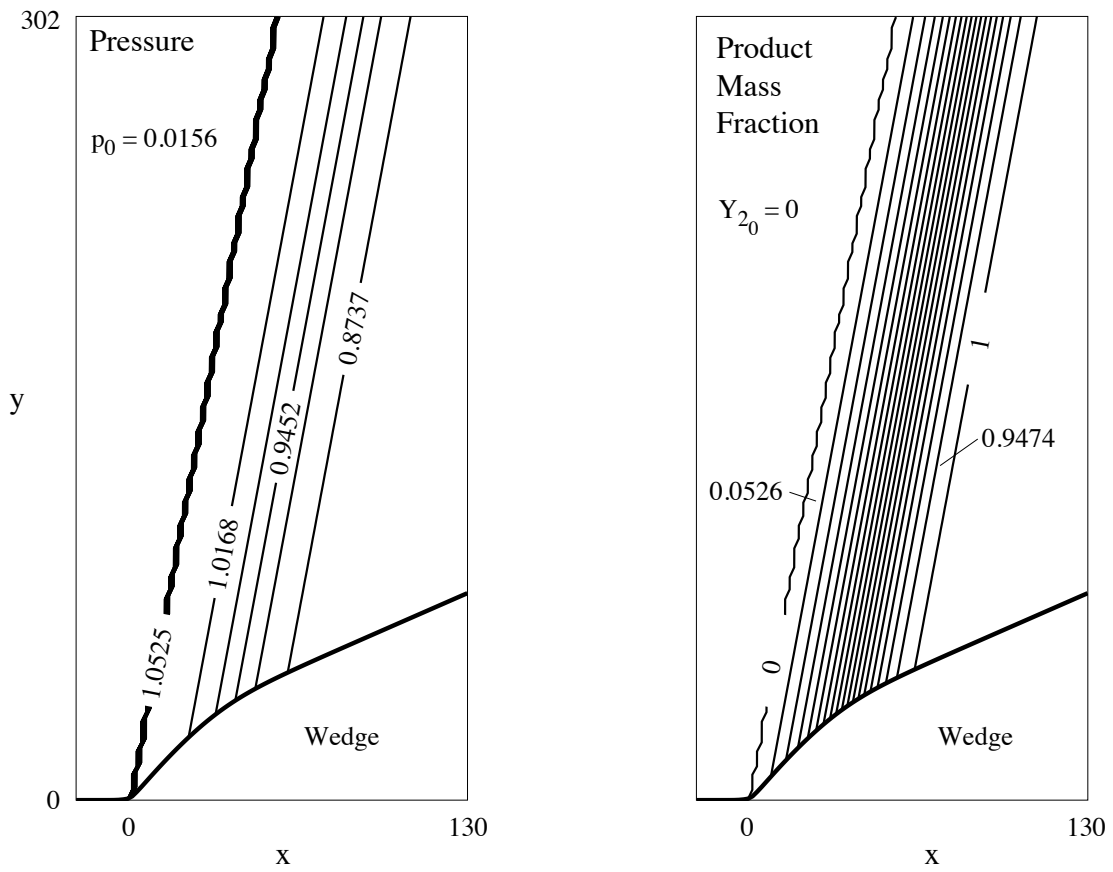


Figure 4.9: Strong straight shock, curved wall detonation solutions for pressure and product mass fraction ($f_n = 1.6$, $\mathcal{M}_0 = 8$, $q = 0.719$, $\Theta = 0.719$, $\gamma = 1.2$).

4.2.2 Curved Shock, Straight Wall

The straight shock, curved wall detonation solution was simple to obtain because it was only pseudo-two-dimensional; the assumption of a straight shock leads to an essentially one-dimensional solution. Powers and Stewart [63] considered the more complex problem of the oblique detonation resulting from a wedge with a straight wall. In this case the oblique detonation has curvature and a truly two-dimensional character. A general description of their solution procedure will be presented here, followed by some example solutions. The reader is referred to [63] for the complete, detailed analysis.

The governing equations studied were the steady versions of Eqs. (3.18–3.22). This model was studied in the limit of high Mach number, linearized about the inert oblique shocked state. In this limit the kinetic energy of the flow is much greater than the heat release from chemical reaction. The leading order solution was an inert shock, and the linear asymptotic theory corrected for the effects of small heat release. Also in this limit the induction zone length was effectively zero; the assumption of large activation energy, which gives rise to a thick induction zone and thin reaction zone, was not made. Consequently, a simple leading order solution of the kinetic rate law was available. At the following order, acoustic equations with chemical reaction forcing terms generated at leading order were solved to determine the pressure and velocity fields. The resulting solution was rotational and characterized by a curved shock attached to a straight wedge.

The solution procedure of [63] was as follows. To simplify application of the boundary conditions, the Euler equations were first transformed to a non-orthogonal coordinate system fixed to the inert shock and wedge. The equations and shock conditions were then written as linear equations in the high Mach number limit. Assuming the oblique shock was weak and the trailing flow was supersonic allowed the equations to be written in characteristic form. These equations were then solved

with the shock position function as a parameter. The shock position function was then specified such that a downstream boundary condition on the wedge surface was met.

Figures 4.10 and 4.11 show asymptotic solution contours for vorticity and product mass fraction. The flow features follow the description of an oblique detonation given in Ch. 1 (see Fig. 1.1). As shown by the vorticity contours, the detonation front is curved; curvature is a maximum near the wedge apex, and then decreases to zero away from the apex. The curved detonation front generates vorticity which then is convected downstream along the wedge. The highest vorticity is generated by the maximum curvature near the wedge apex, and then convected downstream along the wedge surface. The vorticity decreases away from the wedge surface, eventually reaching zero when the shock curvature reaches zero. The product mass fraction contours show the reaction initiated by the front, and then proceeding to completion over a finite distance.

Figure 4.12 shows the asymptotic solution contours for pressure. The minimum detonation pressure occurs near the wedge apex where the detonation front is essentially an inert shock. The maximum pressure occurs where the curvature is zero and the detonation front has reached its maximum angle. Just as in a ZND detonation, the pressure peaks at the front and then decreases as the reaction progresses.

Grismer and Powers [32] demonstrated the utility of the asymptotic solutions as benchmarks to which numerical solutions can be compared. Figure 4.13 shows a numerical pressure solution obtained with RPLUS, a code under development at NASA Lewis Research Center [70]. Comparing Figs. 4.12 and 4.13 it is apparent that the numerical solution is very similar to the asymptotic solution, but some of the details in the flow are somewhat different. In order to better quantify the comparison between the asymptotic and numerical solutions, the asymptotic solution was “written” on the same grid used for the numerical calculations. The two solutions

were then compared using a fractional error [86] defined as

$$e_1 = \sum_{i=1}^N |p_{a_i} - p_{n_i}|/N, \quad (4.49)$$

where p_{a_i} is the asymptotic pressure at a point, p_{n_i} is the numerical solution at a point, and N is the total number of points. Thus, e_1 is essentially an average pressure difference between the two solutions at a given point.

Figure 4.14 shows a plot of e_1 versus freestream Mach number \mathcal{M}_0 . Asymptotic and numerical solutions were calculated for different cases of freestream Mach number and heat release, and then the corresponding e_1 values determined (filled symbols). To aid in interpreting the results, asymptotic and numerical solutions for zero heat release were also compared to the exact oblique shock solution (open symbols). Finally, the predicted order of the error of the asymptotic and numerical solutions was plotted. The error in the asymptotic solution is the next highest order not considered, or $O(\epsilon^2) = 1/\mathcal{M}_0^4$. The error in the numerical solution is the order of accuracy of the finite differencing scheme used, or $O(\Delta x^2)$. Since the same fixed grid was used for all calculations, this was constant. Measured order of accuracy results in Ch. 6 and from other researchers [86] suggest that for nonlinear flows with discontinuities the actual order of accuracy is much lower; however, as no order of accuracy testing was done with RPLUS, the initial estimate was retained.

The trend of the comparisons to the exact oblique shock solution highlights the behavior of the asymptotic and numerical solutions. The exact/numerical comparison shows that the error of the numerical solution remains nearly constant regardless of the Mach number. On the other hand, the exact/asymptotic comparison shows that the error of the asymptotic solution decreases with increasing Mach number. These are both consistent with the behavior of the respective predicted error curves shown. The large difference between the magnitude of the predicted errors and the calculated errors is likely due to the smeared shock profile of the numerical solution and the error in the prediction of the shock location of the asymptotic solution. Examining

the asymptotic/numerical errors for $\bar{q} = 0$ in light of the comparisons to the exact solution, it is apparent that for low Mach number the difference in the solutions is due to the error of the asymptotic method. At high Mach number the difference is due to the error of the numerical method. Adding a small amount of heat to the flow ($\bar{q} = 1$) retains the same behavior. Adding a larger amount of heat to the flow ($\bar{q} = 10$) results in a curve with the same trends as $\bar{q} = 0$ and 1, but having much higher values. This is due to large errors in the asymptotic solution; this level of heat release is inconsistent with the initial assumptions made in obtaining the asymptotic solution. Thus, for low heat release and high Mach number, the asymptotic solution was shown to be a useful quantitative benchmark for numerical solutions of high speed flows. At lower Mach numbers or high heat release, it has value as a qualitative tool for predicting general flowfield features.

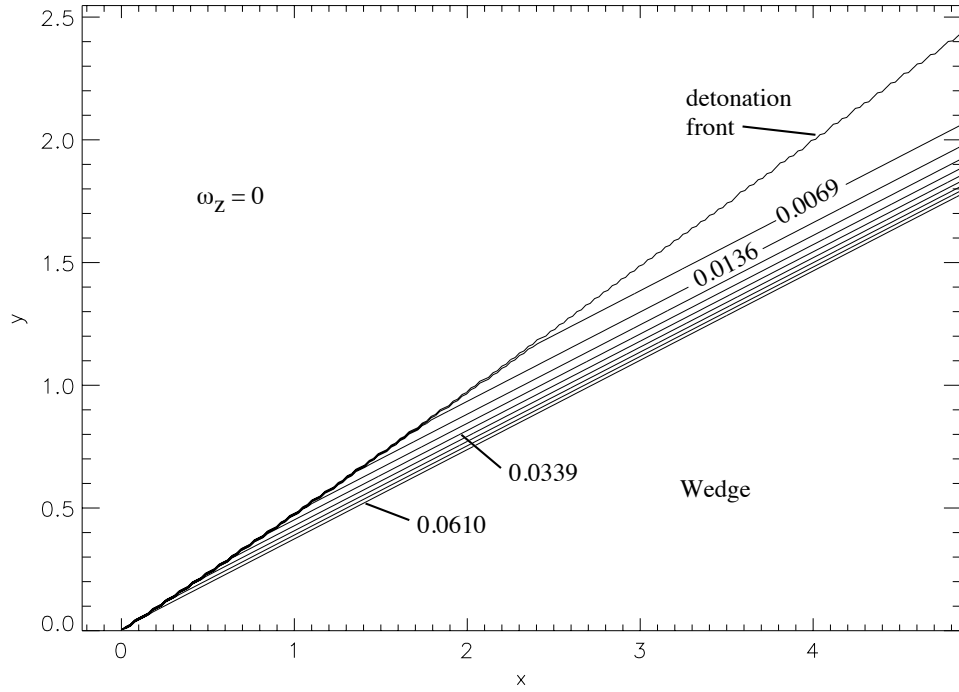


Figure 4.10: Oblique detonation vorticity contours from asymptotic solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).

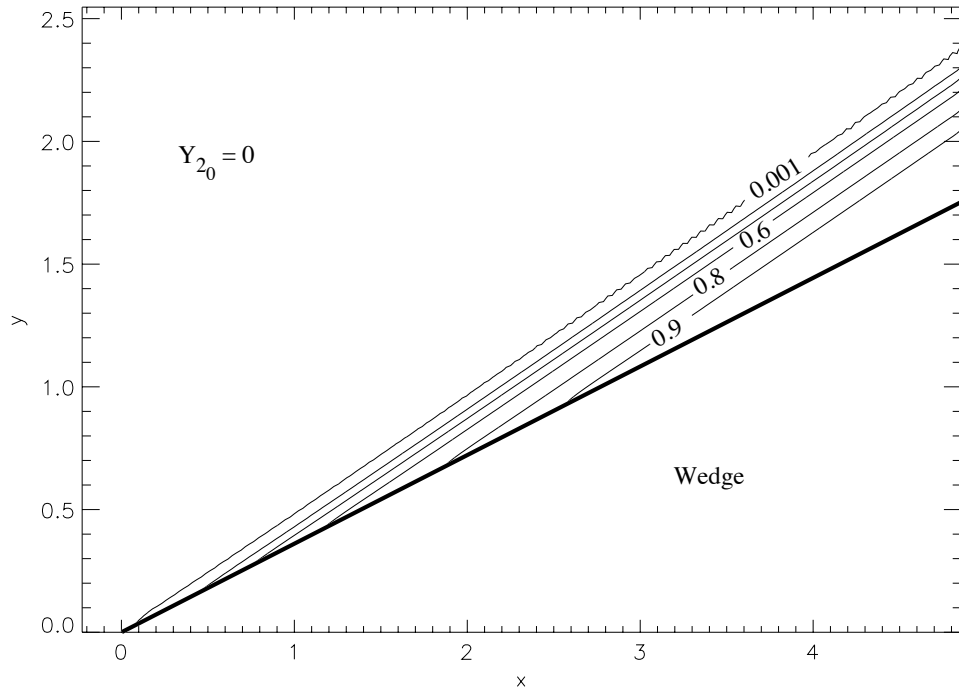


Figure 4.11: Oblique detonation product mass fraction contours from asymptotic solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).

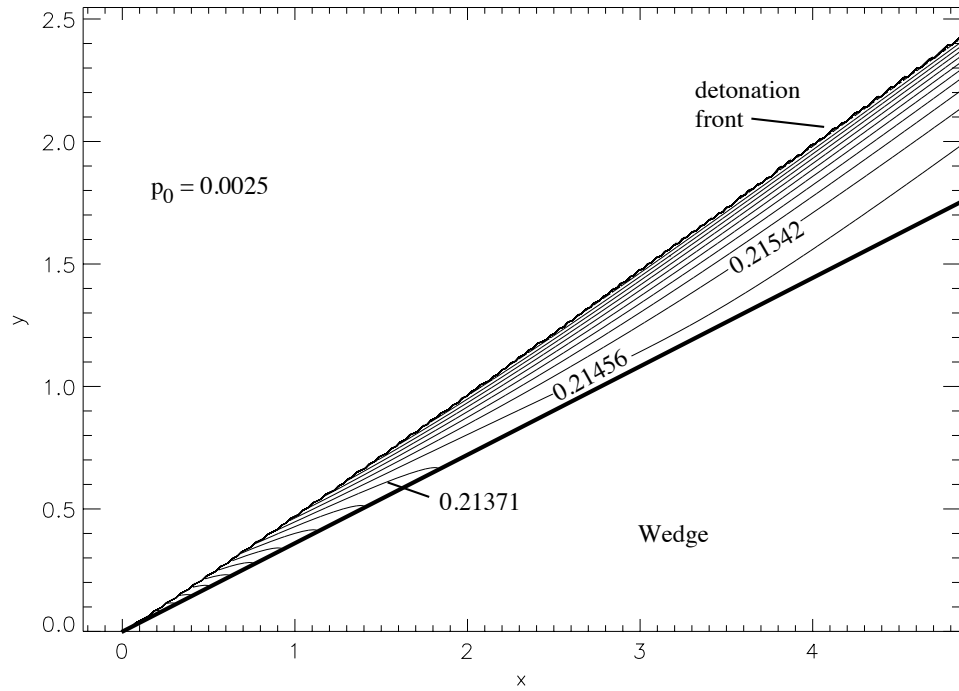


Figure 4.12: Oblique detonation pressure contours from asymptotic solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).

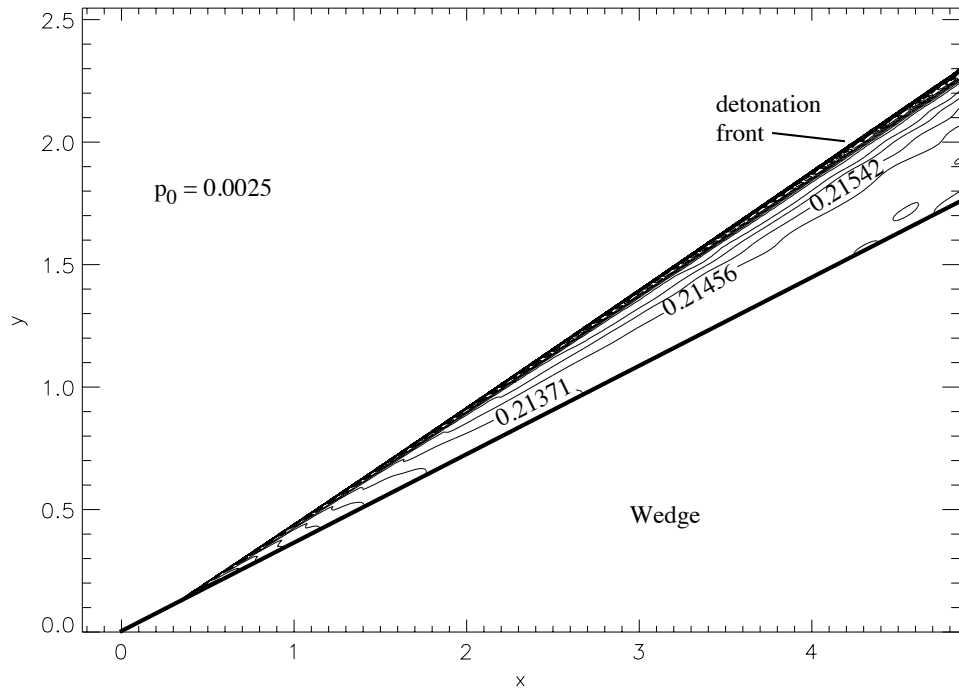


Figure 4.13: Oblique detonation pressure contours from numerical solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).

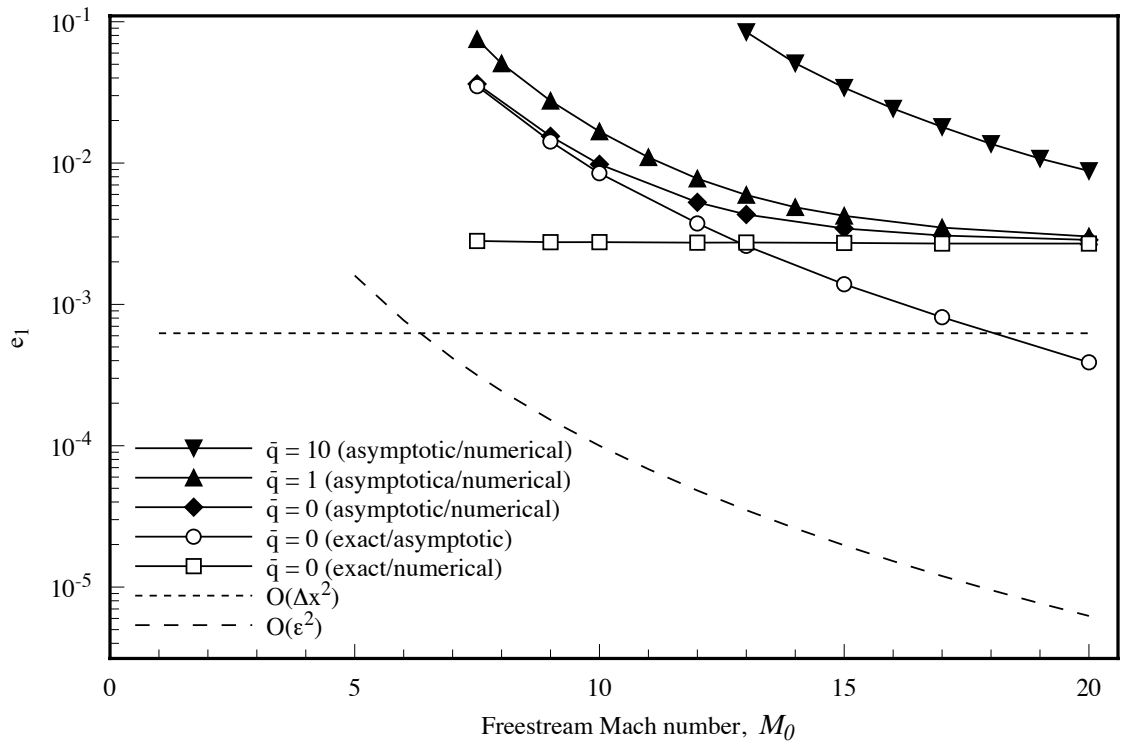


Figure 4.14: Comparison of exact, asymptotic and numerical solutions using average pressure difference e_1 .

Chapter 5

NUMERICAL METHOD

This chapter will present the numerical algorithm implemented for this study. The chapter will be broken into four sections: a brief description of Godunov schemes, a discussion of Roe's approximate Riemann solver and its use in a first order scheme, extension of the scheme to second order accuracy in space and time, and the actual implementation of the second order scheme for the reactive Euler equations in the two-dimensional generalized coordinate system.

5.1 Godunov Schemes

Roe's scheme is part of a broader class of methods referred to as Godunov schemes. Unlike finite difference methods, in which the solution is assumed to be smooth, Godunov's [31] idea was to average a series of exact, nonlinear, discontinuous solutions of the conservation equations to arrive at an overall solution. This type of scheme captures discontinuous flow solutions, such as shocks, naturally without resorting to the artificial dissipation methods necessary in finite difference methods. Godunov chose the exact solution of the Riemann problem as his discontinuous solution. The Riemann problem may be thought of as follows: consider a tube filled with inviscid gas and divided into two sections by a diaphragm. In each section the gas is at different conditions of pressure, density, and/or temperature. At $\tilde{t} = 0$ the diaphragm breaks, sending a shock into the gas of lower pressure. Convected along at the particle velocity

behind the shock is a contact discontinuity, and moving in the opposite direction into

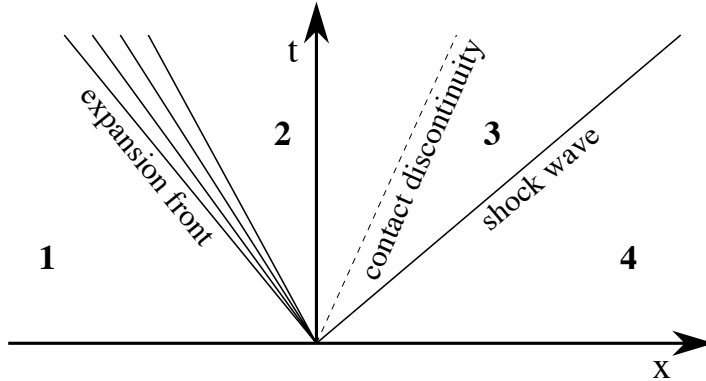


Figure 5.1: x - t diagram of the Riemann problem.

the high pressure side is an expansion wave. This is illustrated in Fig. 5.1; each of the waves moves at a constant velocity, thus the curves in the x - t diagram are linear. The solution between the waves, denoted by **1**–**4** in Fig. 5.1, is a constant state. Details of the Riemann solution are well known [40]. In fact, the Riemann solution is a similarity solution; along lines of $x/t = \text{constant}$ the solution remains constant. Figure 5.2 shows a Riemann problem solution at $\tilde{t} = 0.0061$ s in which the diaphragm was located at $\tilde{x} = 0$, the initial velocity was zero everywhere, the initial ratio between the left and right state density was 8, the initial pressure ratio was 10, $\gamma = 1.4$, and the total dimensional domain length was 10 m. These parameters were chosen for direct comparison with published results [40]. The variables have been scaled by their peak values, and the spatial coordinate has been scaled by the domain length. The contact discontinuity (evident as the first step in the density plot) and shock are moving to the right, while the expansion wave moves to the left. The solution varies smoothly and continuously through the expansion region, while remaining constant or varying discontinuously in the other regions.

Godunov utilized the Riemann solution by considering the flow variables to be constant across a given computational cell i , but different on a cell by cell basis (see Fig. 5.3). The different constant states in each cell led to a Riemann problem at the

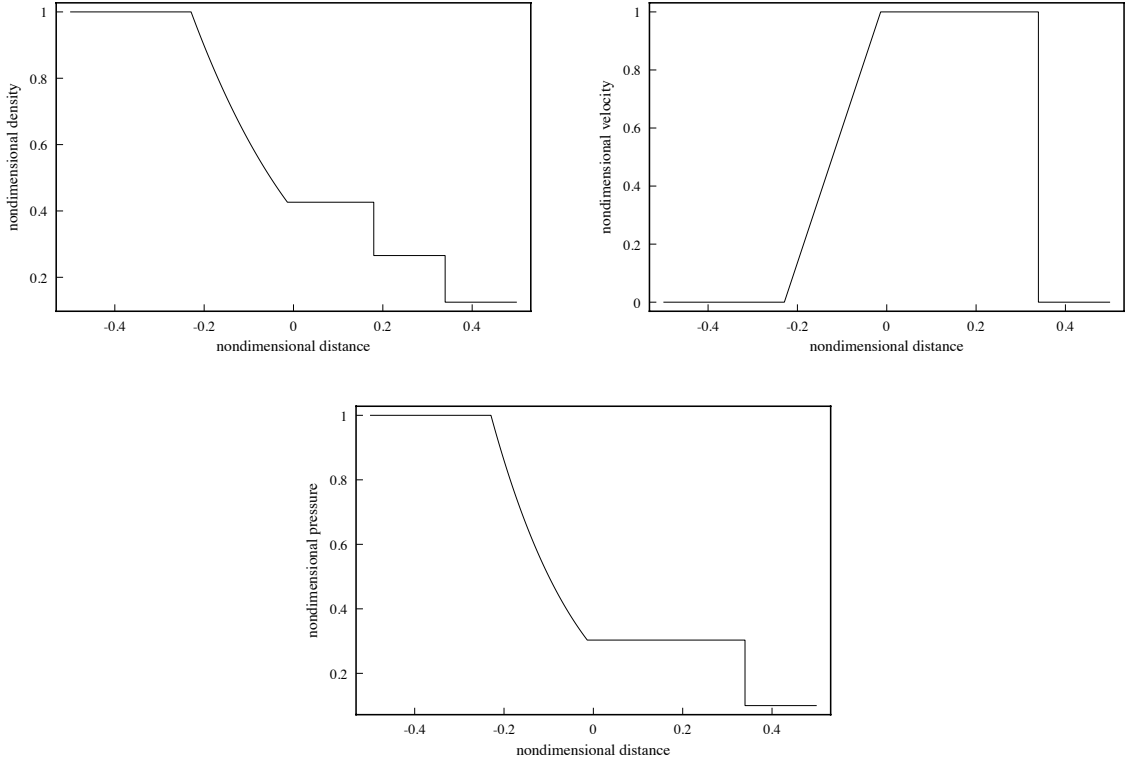


Figure 5.2: Exact solution to Riemann problem at $\tilde{t} = 0.0061$ s (diaphragm initially at $\tilde{x} = 0$, initial pressure ratio = 10, initial density ratio = 8, initial velocity = 0, $\gamma = 1.4$, total domain length = 10 m).

cell interfaces $(i - 1/2)$ and $(i + 1/2)$. The Riemann problem for each interface was then solved, taking Δt sufficiently small such that there was no interaction between adjacent interface solutions. Finally, the resulting Riemann solutions were averaged in each cell to obtain the overall flow solution for that time step.

The perceived deficiency of Godunov's method was that each Riemann problem must be solved with an iteration process. For a large number of cells this requires significant computational time, making the scheme inefficient compared to other common methods. However, since the detail of each Riemann problem is lost in the averaging process, the necessity of obtaining the *exact* solution is questionable. In the interest of efficiency, a number of *approximate* Riemann solutions have been developed that do not require iteration.

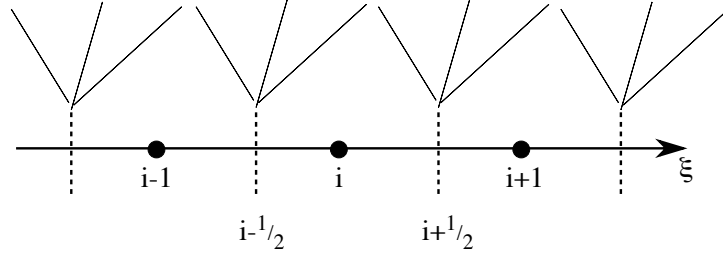


Figure 5.3: Illustration of Godunov’s method.

5.2 The Roe Scheme

Roe [66] published his approximate Riemann solution in 1981, and since that time it has been widely used, *e.g.* [84, 80, 5, 46, 54]. In particular, in a recent article on unsteady detonations [19], Roe’s method was chosen following a detailed comparison between a number of different schemes. The largest concern in the study was that the shock/detonation front be captured cleanly and oscillation free without resorting to artificial dissipation methods. Artificial dissipation is undesirable because it could dampen physical oscillations in the unsteady solutions of interest. Roe was able to show that his approximate Riemann solver reduces to the standard Rankine-Hugoniot jump conditions for a shock discontinuity; thus it seemed well equipped to handle our problem. Much of the following discussion is based upon a book by LeVeque [50], which focuses on numerical methods for hyperbolic systems of conservation equations.

For the control volume defined by $i - 1/2$ and $i + 1/2$ in Fig. 5.3, the integral form of the one-dimensional conservation equations (Eqs. 3.46) is

$$\frac{d}{d\tau} \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \hat{\mathbf{q}}(\xi, \tau) d\xi = \hat{\mathbf{f}}(\hat{\mathbf{q}}(\xi_{i-1/2}, \tau)) - \hat{\mathbf{f}}(\hat{\mathbf{q}}(\xi_{i+1/2}, \tau)). \quad (5.1)$$

where $\xi_i = i(\Delta\xi)$. It is noted that this form is actually more fundamental in that the governing partial differential equations are typically derived from an integral formulation after the assumption of a continuously varying integrand is made. The integral form allows discontinuities to be present in the domain. Equation (5.1) can

be integrated to give

$$\begin{aligned} \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \hat{\mathbf{q}}(\xi, \tau^{n+1}) d\xi &= \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \hat{\mathbf{q}}(\xi, \tau^n) d\xi + \int_{\tau^n}^{\tau^{n+1}} \hat{\mathbf{f}}(\hat{\mathbf{q}}(\xi_{i-1/2}, \tau)) d\tau \\ &\quad - \int_{\tau^n}^{\tau^{n+1}} \hat{\mathbf{f}}(\hat{\mathbf{q}}(\xi_{i+1/2}, \tau)) d\tau. \end{aligned} \quad (5.2)$$

By defining

$$\hat{\mathbf{Q}}_i^n = \frac{1}{\Delta\xi} \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \hat{\mathbf{q}}(\xi, \tau^n) d\xi, \quad (5.3)$$

$$\hat{\mathbf{F}}(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n) = \frac{1}{\Delta\tau} \int_{\tau^n}^{\tau^{n+1}} \hat{\mathbf{f}}(\hat{\mathbf{q}}(\xi_{i+1/2}, \tau)) d\tau \quad (5.4)$$

$$\hat{\mathbf{F}}(\hat{\mathbf{Q}}_{i-1}^n, \hat{\mathbf{Q}}_i^n) = \frac{1}{\Delta\tau} \int_{\tau^n}^{\tau^{n+1}} \hat{\mathbf{f}}(\hat{\mathbf{q}}(\xi_{i-1/2}, \tau)) d\tau \quad (5.5)$$

Eqs. (5.2) become

$$\hat{\mathbf{Q}}_i^{n+1} = \hat{\mathbf{Q}}_i^n - \frac{\Delta\tau}{\Delta\xi} [\hat{\mathbf{F}}(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n) - \hat{\mathbf{F}}(\hat{\mathbf{Q}}_{i-1}^n, \hat{\mathbf{Q}}_i^n)]. \quad (5.6)$$

Equations (5.3) define an average $\hat{\mathbf{q}}$ over the spatial interval $\xi_{i-1/2}$ to $\xi_{i+1/2}$. Equations (5.4) and (5.5) define average fluxes over the temporal interval τ^n to τ^{n+1} . For Godunov's scheme and the Roe scheme to be used here, the initial data $\hat{\mathbf{q}}(\xi_i, \tau^n)$ is considered to be constant across the interval, thus $\hat{\mathbf{Q}}_i^n = \hat{\mathbf{q}}(\xi_i, \tau^n)$. The integral in Eqs. (5.4) is actually simpler than it appears. For Godunov's scheme, the value of $\hat{\mathbf{q}}(\xi_{i+1/2}, \tau)$ remains constant over the interval τ^n to τ^{n+1} due to the similarity nature of the Riemann solution (along $(\xi - \xi_{i+1/2})/\tau = \text{constant} = 0$ the solution is constant). This is also true of the approximate Riemann solution of Roe. Thus,

$$\hat{\mathbf{F}}(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n) = \hat{\mathbf{f}}(\hat{\mathbf{q}}(\xi_{i+1/2}, \tau)) = \hat{\mathbf{f}}(\hat{\mathbf{q}}^*(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n)), \quad (5.7)$$

where $\hat{\mathbf{q}}^*(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n)$ denotes the value of $\hat{\mathbf{q}}$ at the interface due to the Riemann solution between cell i and $i+1$. If the piecewise constant solution for \mathbf{q} at the new time is taken to be the averaged value, $\hat{\mathbf{q}}(\xi_i, \tau^{n+1}) = \hat{\mathbf{Q}}_i^{n+1}$, then Eqs. (5.6) become

$$\hat{\mathbf{q}}_i^{n+1} = \hat{\mathbf{q}}_i^n - \frac{\Delta\tau}{\Delta\xi} (\hat{\mathbf{f}}_{i+1/2}^n - \hat{\mathbf{f}}_{i-1/2}^n), \quad (5.8)$$

where $\hat{\mathbf{q}}_i^n = \hat{\mathbf{q}}(\xi_i, \tau^n)$ and $\hat{\mathbf{f}}_{i+1/2}^n = \hat{\mathbf{f}}(\hat{\mathbf{q}}^*(\hat{\mathbf{Q}}_i^n, \hat{\mathbf{Q}}_{i+1}^n))$. Equations (5.8) define the numerical scheme for a general Godunov-type method; the way in which the interface fluxes $\hat{\mathbf{f}}_{i+1/2}^n$ are determined distinguishes the various schemes.

Roe's approach was to approximate the Riemann solution by considering a linear system of equations:

$$\begin{aligned} \frac{\partial \hat{\mathbf{q}}}{\partial \tau} + \bar{\mathbf{A}} \frac{\partial \hat{\mathbf{q}}}{\partial \xi} &= 0, \\ \hat{\mathbf{q}}(\xi, 0) &= \hat{\mathbf{q}}_0(\xi), \end{aligned} \tag{5.9}$$

where $\bar{\mathbf{A}}$ is a constant matrix. Note that since $\bar{\mathbf{A}}$ is constant, the flux vector for this system is simply $\bar{\mathbf{f}}(\hat{\mathbf{q}}) = \bar{\mathbf{A}}\hat{\mathbf{q}}$. For the moment, consider this to be the one-dimensional, linear version of Eqs. (3.48) without the source term $\hat{\mathbf{w}}$. This system of equations has an exact solution which can be more easily seen by considering a simpler one equation system:

$$\begin{aligned} \frac{\partial \hat{q}}{\partial \tau} + \bar{A} \frac{\partial \hat{q}}{\partial \xi} &= 0, \\ \hat{q}(\xi, 0) &= \hat{q}_0(\xi), \end{aligned} \tag{5.10}$$

where \bar{A} is a scalar constant. A characteristic solution for this can be obtained by noting that

$$d\hat{q} = \frac{\partial \hat{q}}{\partial \tau} d\tau + \frac{\partial \hat{q}}{\partial \xi} d\xi, \tag{5.11}$$

and dividing through by $d\tau$

$$\frac{d\hat{q}}{d\tau} = \frac{\partial \hat{q}}{\partial \tau} + \frac{\partial \hat{q}}{\partial \xi} \frac{d\xi}{d\tau}. \tag{5.12}$$

Comparing Eqs. (5.10) and (5.12) it is apparent that along the characteristic $d\xi/d\tau = \bar{A}$

$$\frac{d\hat{q}}{d\tau} = 0. \tag{5.13}$$

Simply integrating Eq. (5.13) shows that \hat{q} is constant along $d\xi/d\tau = \bar{A}$, and can be determined everywhere using the initial condition

$$\hat{q}(\xi, \tau) = \hat{q}_0(\xi_0), \tag{5.14}$$

where ξ_0 is the intersection with the ξ -axis of the characteristic passing through (ξ, τ) . This can be written more simply by considering the solution for the characteristic equation

$$\xi - \xi_0 = \bar{\bar{\mathbf{A}}}(\tau - \tau_0). \quad (5.15)$$

Solving for ξ_0 ($\tau_0 = 0$)

$$\xi_0 = \xi - \bar{\bar{\mathbf{A}}}\tau, \quad (5.16)$$

and substituting leads to

$$\hat{q}(\xi, \tau) = \hat{q}_0(\xi - \bar{\bar{\mathbf{A}}}\tau). \quad (5.17)$$

Note that if the initial condition is discontinuous, $\hat{q}(\xi, \tau)$ will be discontinuous.

The same type of treatment can be used on Eqs. (5.9), provided the system of equations can be decoupled into a set of independent equations. This is accomplished by diagonalizing $\bar{\bar{\mathbf{A}}}$, which can always be done for the flux Jacobian matrix of a hyperbolic system of equations [50]. The matrix $\bar{\bar{\mathbf{A}}}$ can be written

$$\bar{\bar{\mathbf{A}}} = \bar{\bar{\mathbf{R}}}\bar{\bar{\boldsymbol{\lambda}}}\bar{\bar{\mathbf{R}}}^{-1}, \quad (5.18)$$

where $\bar{\bar{\mathbf{R}}}$ and $\bar{\bar{\boldsymbol{\lambda}}}$ are the right eigenvector matrix and eigenvalue matrix of $\bar{\bar{\mathbf{A}}}$, respectively. Substituting into Eqs. (5.9) and multiplying by $\bar{\bar{\mathbf{R}}}^{-1}$ results in

$$\bar{\bar{\mathbf{R}}}^{-1} \frac{\partial \hat{\mathbf{q}}}{\partial \tau} + \bar{\bar{\boldsymbol{\lambda}}}\bar{\bar{\mathbf{R}}}^{-1} \frac{\partial \hat{\mathbf{q}}}{\partial \xi} = 0. \quad (5.19)$$

Since $\bar{\bar{\mathbf{R}}}^{-1}$ is constant, it can be brought within the derivative. With a new variable \mathbf{v} defined as

$$\mathbf{v} = \bar{\bar{\mathbf{R}}}^{-1} \hat{\mathbf{q}}. \quad (5.20)$$

Eq. (5.19) transforms to

$$\frac{\partial \mathbf{v}}{\partial \tau} + \bar{\bar{\boldsymbol{\lambda}}} \frac{\partial \mathbf{v}}{\partial \xi} = 0. \quad (5.21)$$

Since $\bar{\bar{\boldsymbol{\lambda}}}$ is a diagonal matrix of eigenvalues, this is a decoupled system of equations which can be written

$$\frac{\partial v_m}{\partial \tau} + \bar{\lambda}_m \frac{\partial v_m}{\partial \xi} = 0 \quad m = 1, 2, \dots, N, \quad (5.22)$$

where N is the number of equations. The solutions are then simply

$$v_m(\xi, \tau) = v_m(\xi - \bar{\lambda}_m \tau) \quad m = 1, 2, \dots, N, \quad (5.23)$$

where the initial conditions for \mathbf{v} are found using Eqs. (5.20)

$$\mathbf{v}(\xi, 0) = \bar{\mathbf{R}}^{-1} \hat{\mathbf{q}}_0(\xi). \quad (5.24)$$

Finally, the solution in terms of the original variables is found by solving for \mathbf{q} in Eqs. (5.20):

$$\hat{\mathbf{q}}(\xi, \tau) = \bar{\mathbf{R}} \mathbf{v}(\xi, \tau). \quad (5.25)$$

Working out the matrix multiplication on the right hand side, this can be written as an eigenvector expansion

$$\hat{\mathbf{q}}(\xi, \tau) = \sum_{m=1}^N v_m(\xi, \tau) \bar{\mathbf{r}}_m, \quad (5.26)$$

where $\bar{\mathbf{r}}_m$ are the right eigenvectors.

The Riemann problem is simply a discontinuous initial condition

$$\hat{\mathbf{q}}_0(\xi) = \begin{cases} \hat{\mathbf{q}}_l & \xi < 0 \\ \hat{\mathbf{q}}_r & \xi \geq 0 \end{cases}, \quad (5.27)$$

where $\hat{\mathbf{q}}_l$ and $\hat{\mathbf{q}}_r$ are constant left and right states, respectively. Likewise, the characteristic variables have constant left and right initial states

$$v_m(\xi, 0) = \begin{cases} v_{m_l} & \xi < 0 \\ v_{m_r} & \xi \geq 0 \end{cases} \quad m = 1, 2, \dots, N, \quad (5.28)$$

which can be used in the eigenvector expansion of Eqs. (5.26) to give

$$\hat{\mathbf{q}}_l = \sum_{m=1}^N v_{m_l} \bar{\mathbf{r}}_m \quad \hat{\mathbf{q}}_r = \sum_{m=1}^N v_{m_r} \bar{\mathbf{r}}_m. \quad (5.29)$$

Following Eq. (5.23) the characteristic solution is simply

$$v_m(\xi, \tau) = \begin{cases} v_{m_l} & \text{if } \xi - \bar{\lambda}_m \tau < 0 \\ v_{m_r} & \text{if } \xi - \bar{\lambda}_m \tau \geq 0 \end{cases} \quad m = 1, 2, \dots, N. \quad (5.30)$$

Assuming the eigenvalues $\bar{\lambda}_m$ are arranged such that

$$\bar{\lambda}_1 \leq \bar{\lambda}_2 \leq \dots \leq \bar{\lambda}_N, \quad (5.31)$$

the solution for $\hat{\mathbf{q}}$ can be written

$$\hat{\mathbf{q}}(\xi, \tau) = \sum_{m=1}^N v_m \bar{\mathbf{r}}_m = \sum_{m=1}^{\ell(\xi, \tau)} v_{m_r} \bar{\mathbf{r}}_m + \sum_{m=\ell(\xi, \tau)+1}^N v_{m_l} \bar{\mathbf{r}}_m, \quad (5.32)$$

where $\ell(\xi, \tau)$ is defined as the maximum value of m such that $\xi - \bar{\lambda}_m \tau \geq 0$. Figure 5.4 shows an example of a solution; characteristics emanating from the negative side

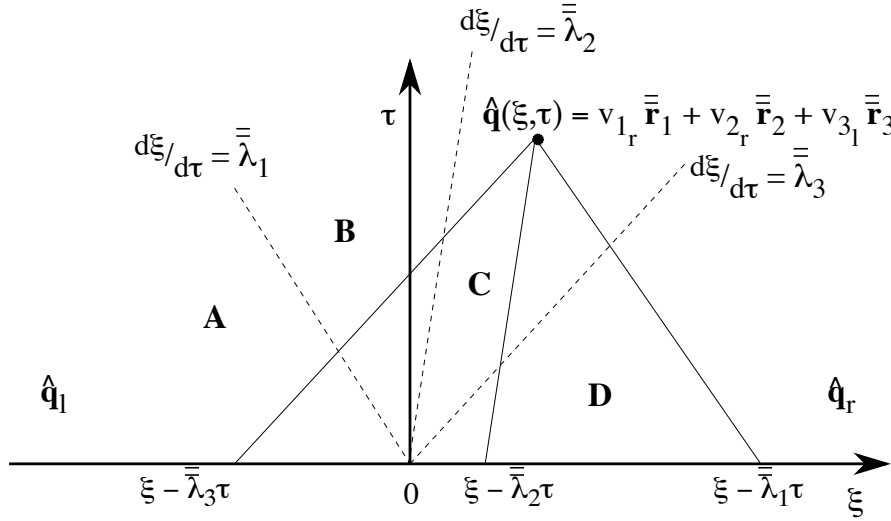


Figure 5.4: Diagram of linear Riemann solution.

of the ξ -axis result in $v_m = v_{m_l}$, while those emanating from the positive side lead to $v_m = v_{m_r}$. Thus, $v_1 = v_{1_r}$, $v_2 = v_{2_r}$, and $v_3 = v_{3_l}$, and the solution from Eqs. (5.32) is

$$\hat{\mathbf{q}}(\xi, \tau) = v_{1_r} \bar{\mathbf{r}}_1 + v_{2_r} \bar{\mathbf{r}}_2 + v_{3_l} \bar{\mathbf{r}}_3. \quad (5.33)$$

Additionally, the value of $\hat{\mathbf{q}}$ is constant within the wedge defined by $d\xi/d\tau = \bar{\lambda}_2$ and $d\xi/d\tau = \bar{\lambda}_3$ (region **C** of Fig. 5.4). Since the characteristics emanating from the points $\xi - \bar{\lambda}_m \tau$ (solid lines) parallel the characteristics $d\xi/d\tau = \bar{\lambda}_m$ at the origin (dashed lines), moving the point of interest (ξ, τ) within region **C** will not result in any of the emanating characteristics (solid lines) crossing the origin; therefore the values of v_m

will not change. The same reasoning applies to each of the four regions defined by $d\xi/d\tau = \bar{\lambda}_m$.

Figure 5.5 indicates the constant value of $\hat{\mathbf{q}}$ for each region. The solution changes

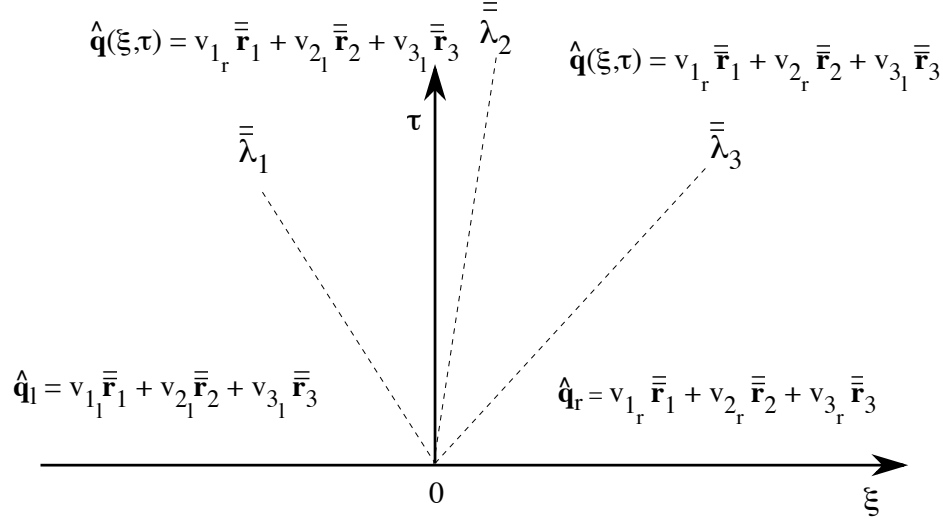


Figure 5.5: Linear Riemann solution by quadrant.

discontinuously across each characteristic; the jump across the m^{th} characteristic is ($[]$ denotes the difference operator)

$$[\hat{\mathbf{q}}] = (v_{m_r} - v_{m_l})\bar{\mathbf{r}}_m. \quad (5.34)$$

By beginning with the left state and summing the differences for the characteristics that would be crossed to reach the point of interest, the Riemann problem solution may be written

$$\hat{\mathbf{q}}(\xi, \tau) = \hat{\mathbf{q}}_l + \sum_{\bar{\lambda}_m < \xi/\tau} (v_{m_r} - v_{m_l})\bar{\mathbf{r}}_m. \quad (5.35)$$

Likewise, the solution may be written by beginning with the right state and subtracting the differences for the characteristics crossed:

$$\hat{\mathbf{q}}(\xi, \tau) = \hat{\mathbf{q}}_r - \sum_{\bar{\lambda}_m \geq \xi/\tau} (v_{m_r} - v_{m_l})\bar{\mathbf{r}}_m. \quad (5.36)$$

The jump from the left to right state is then made up of a series of smaller jumps

$$\hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l = (v_{1_r} - v_{1_l})\bar{\mathbf{r}}_1 + (v_{2_r} - v_{2_l})\bar{\mathbf{r}}_2 + (v_{3_r} - v_{3_l})\bar{\mathbf{r}}_3. \quad (5.37)$$

Since the eigenvectors $\bar{\mathbf{r}}_m$ and left and right states $\hat{\mathbf{q}}_l$ and $\hat{\mathbf{q}}_r$ are known, this equation can be used to solve for $\alpha_m = v_{m_r} - v_{m_l}$. Adding the terms together and rearranging, Eqs. (5.37) can be written

$$\hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l = \bar{\mathbf{R}}\boldsymbol{\alpha}, \quad (5.38)$$

which is then easily solved by multiplying by the inverse eigenvector matrix

$$\boldsymbol{\alpha} = \bar{\mathbf{R}}^{-1}(\hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l). \quad (5.39)$$

The previous derivation is true for any constant matrix $\bar{\mathbf{A}}$ with real eigenvalues. In order to make the linear Riemann solution and the actual nonlinear Riemann solution of the Euler equations as alike as possible, it would make sense to define $\bar{\mathbf{A}}$ as some function of the left and right states, $\bar{\mathbf{A}}(\hat{\mathbf{q}}_l, \hat{\mathbf{q}}_r)$. An important contribution of Roe was defining $\bar{\mathbf{A}}(\hat{\mathbf{q}}_l, \hat{\mathbf{q}}_r)$ according to the following criteria [66]:

1. It constitutes a linear mapping from $\hat{\mathbf{q}}$ to $\hat{\mathbf{f}}$.
2. As $\hat{\mathbf{q}}_l \rightarrow \hat{\mathbf{q}}_r \rightarrow \hat{\mathbf{q}}$, then $\bar{\mathbf{A}}(\hat{\mathbf{q}}_l, \hat{\mathbf{q}}_r) \rightarrow \hat{\mathbf{A}}(\hat{\mathbf{q}})$ where $\hat{\mathbf{A}} = \partial\hat{\mathbf{f}}/\partial\hat{\mathbf{q}}$.
3. For any $\hat{\mathbf{q}}_l, \hat{\mathbf{q}}_r$, $\bar{\mathbf{A}} \cdot (\hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l) = \hat{\mathbf{f}}_r - \hat{\mathbf{f}}_l$ (\cdot indicates matrix multiplication for clarity).
4. The eigenvectors of $\bar{\mathbf{A}}$ are linearly independent.

Conditions 1, 2, and 4 are satisfied by straightforward averaging processes such as $\bar{\mathbf{A}} = \frac{1}{2}(\hat{\mathbf{A}}_l + \hat{\mathbf{A}}_r)$, but Condition 3 is not. Condition 3 is motivated by the desire to have $\bar{\mathbf{A}}$ satisfy the Rankine-Hugoniot jump conditions.

The general Rankine-Hugoniot conditions can be derived by considering a one-dimensional shock moving at speed s (see Fig. 5.6). The shock has travelled a distance $s\tau$, and the states to the left and right of the shock are constant at $\hat{\mathbf{q}}_l$ and $\hat{\mathbf{q}}_r$, respectively. The Rankine-Hugoniot equations relating the states can be obtained by

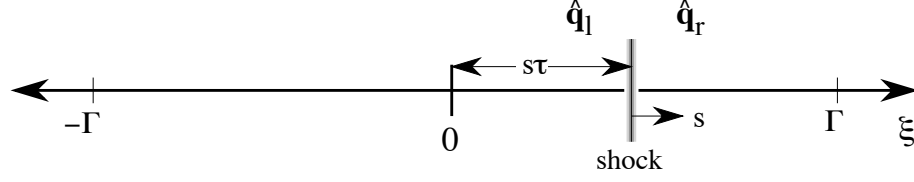


Figure 5.6: Illustration of one-dimensional travelling shock.

considering the integral form of the conservation equations over the control volume $(-\Gamma, \Gamma)$ at time τ (where $\Gamma \gg s\tau$):

$$\frac{d}{d\tau} \int_{-\Gamma}^{\Gamma} \hat{\mathbf{q}}(\xi, \tau) d\xi = \hat{\mathbf{f}}(\hat{\mathbf{q}}_l) - \hat{\mathbf{f}}(\hat{\mathbf{q}}_r) = \hat{\mathbf{f}}_l - \hat{\mathbf{f}}_r. \quad (5.40)$$

The integral in this equation is divided as

$$\int_{-\Gamma}^{\Gamma} \hat{\mathbf{q}}(\xi, \tau) d\xi = \int_{-\Gamma}^{s\tau} \hat{\mathbf{q}}(\xi, \tau) d\xi + \int_{s\tau}^{\Gamma} \hat{\mathbf{q}}(\xi, \tau) d\xi, \quad (5.41)$$

and since $\hat{\mathbf{q}}$ is constant in each interval

$$\int_{-\Gamma}^{\Gamma} \hat{\mathbf{q}}(\xi, \tau) d\xi = (\Gamma + s\tau)\hat{\mathbf{q}}_l + (\Gamma - s\tau)\hat{\mathbf{q}}_r. \quad (5.42)$$

Taking the derivative with respect to time of the right-hand side results in the following Rankine-Hugoniot relation

$$s(\hat{\mathbf{q}}_l - \hat{\mathbf{q}}_r) = \hat{\mathbf{f}}_l - \hat{\mathbf{f}}_r, \quad (5.43)$$

or, in terms of the difference operator,

$$s[\hat{\mathbf{q}}] = [\hat{\mathbf{f}}]. \quad (5.44)$$

For the linear Riemann solution the flux is simply $\bar{\mathbf{f}} = \bar{\bar{\mathbf{A}}}\hat{\mathbf{q}}$, and thus the difference in the flux is $[\bar{\mathbf{f}}] = \bar{\bar{\mathbf{A}}}[\hat{\mathbf{q}}]$. Using Eqs. (5.34), the difference in the linear flux across the m^{th} characteristic can be written

$$\begin{aligned} [\bar{\mathbf{f}}] &= \bar{\bar{\mathbf{A}}}[\hat{\mathbf{q}}] \\ &= \bar{\bar{\mathbf{A}}}\alpha_m \bar{\mathbf{r}}_m. \end{aligned} \quad (5.45)$$

Using $\bar{\bar{\mathbf{A}}}\bar{\mathbf{r}}_m = \bar{\lambda}_m\bar{\mathbf{r}}_m$ results in

$$\begin{aligned} [\bar{\mathbf{f}}] &= \bar{\lambda}_m\alpha_m\bar{\mathbf{r}}_m \\ &= \bar{\lambda}_m[\hat{\mathbf{q}}]. \end{aligned} \quad (5.46)$$

Since each jump is propagating at $\bar{\lambda}_m$, this shows that each jump satisfies the Rankine-Hugoniot conditions. If the initial jump $\hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l$ is an eigenvector of $\bar{\bar{\mathbf{A}}}$, then it will propagate with the speed of one eigenvalue $\bar{\lambda}_m$, and the other jumps will have zero strength. Returning to Roe's conditions on $\bar{\bar{\mathbf{A}}}$, number 3 requires that $\bar{\bar{\mathbf{A}}}$ be defined such that $[\bar{\mathbf{f}}] = [\hat{\mathbf{f}}]$. All shock waves must satisfy Eq. (5.44), therefore the shock speed must be an eigenvalue of $\bar{\bar{\mathbf{A}}}$, $s = \bar{\lambda}_m$. Condition 4 requires that $\bar{\lambda}_m$ correspond to only one eigenvector equal to $\hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l$. Thus, any shock encountered would be captured exactly with one eigenvector of $\bar{\bar{\mathbf{A}}}$, and automatically satisfy the Rankine-Hugoniot conditions.

It is now possible to find the flux at an interface from the approximate solution. Using Eqs. (5.37) with condition 3 results in

$$\begin{aligned} \hat{\mathbf{f}}_r - \hat{\mathbf{f}}_l &= \bar{\bar{\mathbf{A}}}(\hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l) \\ &= \bar{\bar{\mathbf{A}}} \sum_{m=1}^N \alpha_m \bar{\mathbf{r}}_m \\ &= \sum_{m=1}^N \alpha_m \bar{\lambda}_m \bar{\mathbf{r}}_m \\ &= \sum_{m=1}^N d\hat{\mathbf{f}}_m \end{aligned} \quad (5.47)$$

where $d\hat{\mathbf{f}}_m = \alpha_m \bar{\lambda}_m \bar{\mathbf{r}}_m$ is the jump in $\hat{\mathbf{f}}$ across each characteristic in Fig. 5.5. If $\xi = 0$ is considered a cell interface in Fig. 5.5, the flux at the interface, denoted $\bar{\bar{\mathbf{f}}}_{i+1/2}$, can be determined from Eqs. (5.47) by beginning with the left state and summing over only the negatively sloped characteristics

$$\bar{\bar{\mathbf{f}}}_{i+1/2} = \hat{\mathbf{f}}_i + \sum_{\bar{\lambda}_m < 0} \alpha_m \bar{\lambda}_m \bar{\mathbf{r}}_m. \quad (5.48)$$

Likewise, the interface flux can be determined by beginning with the right state and summing over only the positively sloped characteristics

$$\bar{\hat{\mathbf{f}}}_{i+1/2} = \hat{\mathbf{f}}_r - \sum_{\bar{\lambda}_m > 0} \alpha_m \bar{\lambda}_m \bar{\mathbf{r}}_m. \quad (5.49)$$

A third form can be obtained by adding Eqs. (5.48) and (5.49) together and dividing by two

$$\bar{\hat{\mathbf{f}}}_{i+1/2} = \frac{1}{2}(\hat{\mathbf{f}}_l + \hat{\mathbf{f}}_r) - \frac{1}{2} \sum_{m=1}^N \alpha_m |\bar{\lambda}_m| \bar{\mathbf{r}}_m. \quad (5.50)$$

All that remains is to determine the proper averaging process such that $\bar{\mathbf{A}}$ satisfies conditions 1–4. Following Roe, this is accomplished for the reactive Euler equations by defining another vector

$$\mathbf{z} = \sqrt{\frac{\rho}{J}} \begin{bmatrix} 1 \\ u \\ v \\ H \\ Y_2 \end{bmatrix}, \quad (5.51)$$

and writing $\hat{\mathbf{q}}$ and $\hat{\mathbf{f}}$ in terms of it (z_1, z_2, \dots, z_5 are the components of \mathbf{z}):

$$\hat{\mathbf{q}} = J^{-1} \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \\ \rho Y_2 \end{bmatrix} = \begin{bmatrix} z_1^2 \\ z_1 z_2 \\ z_1 z_3 \\ z_1 z_4 / \gamma - \frac{\gamma-1}{\gamma} z_1 z_5 q + \frac{\gamma-1}{2\gamma} (z_2^2 + z_3^2) \\ z_1 z_5 \end{bmatrix}, \quad (5.52)$$

$$\hat{\mathbf{f}} = J^{-1} \begin{bmatrix} \rho U^c \\ \rho u U^c + \xi_x p \\ \rho v U^c + \xi_y p \\ \rho H U^c - \xi_t p \\ \rho Y_2 U^c \end{bmatrix} = \begin{bmatrix} \xi_x z_1 z_2 + \xi_y z_1 z_3 \\ \xi_x z_2^2 + \xi_y z_2 z_3 + \xi_x \frac{\gamma-1}{\gamma} [z_1 z_4 + z_1 z_5 q - \frac{1}{2}(z_2^2 + z_3^2)] \\ \xi_x z_2 z_3 + \xi_y z_3^2 + \xi_y \frac{\gamma-1}{\gamma} [z_1 z_4 + z_1 z_5 q - \frac{1}{2}(z_2^2 + z_3^2)] \\ z_4 (\xi_x z_2 + \xi_y z_3) \\ z_5 (\xi_x z_2 + \xi_y z_3) \end{bmatrix}. \quad (5.53)$$

In general, the jump in $\hat{\mathbf{q}}$, $[\hat{\mathbf{q}}] = \hat{\mathbf{q}}_r - \hat{\mathbf{q}}_l$, can be written in terms of the jump in \mathbf{z} , $[\mathbf{z}] = \mathbf{z}_r - \mathbf{z}_l$, as follows

$$[\hat{\mathbf{q}}] = \bar{\mathbf{C}}[\mathbf{z}], \quad (5.54)$$

where $\bar{\mathbf{C}}$ is a matrix. $\bar{\mathbf{C}}$ is determined by writing the difference for each component of $\hat{\mathbf{q}}$ in terms of the components of \mathbf{z} . In doing this, the following identities are useful:

$$[ab] = a_r b_r - a_l b_l$$

$$\begin{aligned}
&= \frac{1}{2}(a_r + a_l)(b_r - b_l) + \frac{1}{2}(b_r + b_l)(a_r - a_l) \\
&= \bar{a}[b] + \bar{b}[a],
\end{aligned} \tag{5.55}$$

$$\begin{aligned}
[a^2] &= [aa] \\
&= \bar{a}[a] + \bar{a}[a] \\
&= 2\bar{a}[a],
\end{aligned} \tag{5.56}$$

where $\bar{x} = (x_l + x_r)/2$ is the arithmetic mean. Thus, for example,

$$\begin{aligned}
[\hat{q}_1] &= [z_1^2] = 2\bar{z}_1[z_1] \\
[\hat{q}_2] &= [z_1 z_2] = \bar{z}_1[z_2] + \bar{z}_2[z_1],
\end{aligned} \tag{5.57}$$

which leads to

$$\bar{\mathbf{C}} = \begin{bmatrix} 2\bar{z}_1 & 0 & 0 & 0 & 0 \\ \bar{z}_2 & \bar{z}_1 & 0 & 0 & 0 \\ \bar{z}_3 & 0 & \bar{z}_1 & 0 & 0 \\ \bar{z}_4/\gamma - \frac{\gamma-1}{\gamma}q\bar{z}_5 & \frac{\gamma-1}{\gamma}\bar{z}_2 & \frac{\gamma-1}{\gamma}\bar{z}_3 & \bar{z}_1/\gamma & -\frac{\gamma-1}{\gamma}q\bar{z}_1 \\ \bar{z}_5 & 0 & 0 & 0 & \bar{z}_1 \end{bmatrix}. \tag{5.58}$$

In the same way the change in $\hat{\mathbf{f}}$ can be written in terms of the change in \mathbf{z}

$$[\hat{\mathbf{f}}] = \bar{\mathbf{D}}[\mathbf{z}], \tag{5.59}$$

where $\bar{\mathbf{D}}$ can be determined to be

$$\bar{\mathbf{D}} = \begin{bmatrix} \xi_x \bar{z}_2 + \xi_y \bar{z}_3 & \xi_x \bar{z}_1 & \xi_y \bar{z}_1 & 0 & 0 \\ \xi_x \frac{\gamma-1}{\gamma}(\bar{z}_4 + q\bar{z}_5) & \frac{\gamma+1}{\gamma}\xi_x \bar{z}_2 + \xi_y \bar{z}_3 & \xi_y \bar{z}_2 - \frac{\gamma-1}{\gamma}\xi_x \bar{z}_3 & \frac{\gamma-1}{\gamma}\xi_x \bar{z}_1 & \frac{\gamma-1}{\gamma}\xi_x q\bar{z}_1 \\ \xi_y \frac{\gamma-1}{\gamma}(\bar{z}_4 + q\bar{z}_5) & \xi_x \bar{z}_3 - \frac{\gamma-1}{\gamma}\xi_y \bar{z}_2 & \xi_x \bar{z}_2 + \frac{\gamma+1}{\gamma}\xi_y \bar{z}_3 & \frac{\gamma-1}{\gamma}\xi_y \bar{z}_1 & \frac{\gamma-1}{\gamma}\xi_y q\bar{z}_1 \\ 0 & \xi_x \bar{z}_4 & \xi_y \bar{z}_4 & \xi_x \bar{z}_2 + \xi_y \bar{z}_3 & 0 \\ 0 & \xi_x \bar{z}_5 & \xi_y \bar{z}_5 & 0 & \xi_x \bar{z}_2 + \xi_y \bar{z}_3 \end{bmatrix}. \tag{5.60}$$

Roe's matrix $\bar{\mathbf{A}}$ can now be obtained by combining Eqs. (5.54) and (5.59) to get

$$[\hat{\mathbf{f}}] = \bar{\mathbf{D}}\bar{\mathbf{C}}^{-1}[\hat{\mathbf{q}}], \tag{5.61}$$

which matches condition 3 for $\bar{\bar{\mathbf{A}}} = \bar{\mathbf{D}}\bar{\mathbf{C}}^{-1}$. Inverting $\bar{\mathbf{C}}$ and multiplying by $\bar{\mathbf{D}}$ results in the following element of $\bar{\bar{\mathbf{A}}}$, for example:

$$\begin{aligned}\bar{\bar{\mathbf{A}}}_{2,1} &= \frac{\xi_x(\gamma - 1)(\bar{z}_2^2 + \bar{z}_3^2) - 2(\xi_x\bar{z}_2^2 + \xi_y\bar{z}_2\bar{z}_3)}{2\bar{z}_1^2} \\ &= \frac{1}{2}\xi_x(\gamma - 1)\left(\frac{\bar{z}_2^2}{\bar{z}_1^2} + \frac{\bar{z}_3^2}{\bar{z}_1^2}\right) - \frac{\bar{z}_2}{\bar{z}_1}\left(\xi_x\frac{\bar{z}_2}{\bar{z}_1} + \xi_y\frac{\bar{z}_3}{\bar{z}_1}\right).\end{aligned}\quad (5.62)$$

Writing out the terms in detail

$$\frac{\bar{z}_2^2}{\bar{z}_1^2} = \frac{\frac{1}{4}[(\sqrt{\frac{\rho}{J}}u)_r + (\sqrt{\frac{\rho}{J}}u)_l]^2}{\frac{1}{4}[(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l]^2} = \left[\frac{(\sqrt{\frac{\rho}{J}}u)_r + (\sqrt{\frac{\rho}{J}}u)_l}{(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l}\right]^2 = \bar{u}^2, \quad (5.63)$$

$$\frac{\bar{z}_3^2}{\bar{z}_1^2} = \frac{\frac{1}{4}[(\sqrt{\frac{\rho}{J}}v)_r + (\sqrt{\frac{\rho}{J}}v)_l]^2}{\frac{1}{4}[(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l]^2} = \left[\frac{(\sqrt{\frac{\rho}{J}}v)_r + (\sqrt{\frac{\rho}{J}}v)_l}{(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l}\right]^2 = \bar{v}^2, \quad (5.64)$$

$$\frac{\bar{z}_2}{\bar{z}_1} = \bar{u}, \quad (5.65)$$

$$\frac{\bar{z}_3}{\bar{z}_1} = \bar{v}, \quad (5.66)$$

it is apparent that each term is essentially a mass weighted average of the left and right state velocity component. This leads to

$$\bar{\bar{\mathbf{A}}}_{2,1} = \frac{1}{2}\xi_x(\gamma - 1)(\bar{u}^2 + \bar{v}^2) - \bar{u}(\xi_x\bar{u} + \xi_y\bar{v}), \quad (5.67)$$

which has exactly the same form as $\hat{\mathbf{A}}_{2,1}$ (since $\xi_t = 0$), except u and v have been replaced by their averaged quantities, \bar{u} and \bar{v} . In fact, all the elements of $\bar{\bar{\mathbf{A}}}$ match those of $\hat{\mathbf{A}}$ if the variables are replaced by the following ‘‘Roe’’ averaged variables:

$$\begin{aligned}\bar{u} &= \frac{(\sqrt{\frac{\rho}{J}}u)_r + (\sqrt{\frac{\rho}{J}}u)_l}{(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l}, & \bar{v} &= \frac{(\sqrt{\frac{\rho}{J}}v)_r + (\sqrt{\frac{\rho}{J}}v)_l}{(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l}, \\ \bar{H} &= \frac{(\sqrt{\frac{\rho}{J}}H)_r + (\sqrt{\frac{\rho}{J}}H)_l}{(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l}, & \bar{Y}_2 &= \frac{(\sqrt{\frac{\rho}{J}}Y_2)_r + (\sqrt{\frac{\rho}{J}}Y_2)_l}{(\sqrt{\frac{\rho}{J}})_r + (\sqrt{\frac{\rho}{J}})_l}.\end{aligned}\quad (5.68)$$

Since $\bar{\bar{\mathbf{A}}}$ and $\hat{\mathbf{A}}$ have the same form, their eigenvalues have the same form

$$\begin{aligned}\bar{\lambda}_1 &= \bar{\lambda}_2 = \bar{\lambda}_3 = \bar{U}^c, \\ \bar{\lambda}_4 &= \bar{U}^c - \bar{c}\sqrt{\xi_x^2 + \xi_y^2}, \\ \bar{\lambda}_5 &= \bar{U}^c + \bar{c}\sqrt{\xi_x^2 + \xi_y^2},\end{aligned}\quad (5.69)$$

where $\bar{U}^c = \xi_x \bar{u} + \xi_y \bar{v}$ and $\bar{c} = \sqrt{(\gamma - 1)[\bar{H} + \bar{Y}_2 q - (\bar{u}^2 + \bar{v}^2)/2]}$. Likewise, the eigenvector array $\bar{\mathbf{R}}$ retains the same form as $\hat{\mathbf{R}}$, except the variables have been replaced by their Roe averaged quantities

$$\bar{\mathbf{R}} = \begin{bmatrix} \xi_x & 0 & 1 & 1 & 0 \\ \bar{U}^c & \xi_y & \bar{u} - \frac{\xi_x}{\sqrt{\xi_x^2 + \xi_y^2}} \bar{c} & \bar{u} + \frac{\xi_x}{\sqrt{\xi_x^2 + \xi_y^2}} \bar{c} & 0 \\ 0 & -\xi_x & \bar{v} - \frac{\xi_y}{\sqrt{\xi_x^2 + \xi_y^2}} \bar{c} & \bar{v} + \frac{\xi_y}{\sqrt{\xi_x^2 + \xi_y^2}} \bar{c} & 0 \\ (\bar{u}^2 - \bar{v}^2) \frac{\xi_x}{2} + \xi_y \bar{u} \bar{v} & \xi_y \bar{u} - \xi_x \bar{v} & \bar{H} - \frac{\bar{c}}{\sqrt{\xi_x^2 + \xi_y^2}} \bar{U}^c & \bar{H} + \frac{\bar{c}}{\sqrt{\xi_x^2 + \xi_y^2}} \bar{U}^c & -q \\ 0 & 0 & \bar{Y}_2 & \bar{Y}_2 & 1 \end{bmatrix}. \quad (5.70)$$

Thus, Roe's matrix $\bar{\mathbf{A}}$ is simply the flux Jacobian matrix $\hat{\mathbf{A}}$ with Roe averaged variables, and as such satisfies Roe's conditions 1–4.

The implementation of the first order Roe scheme in one dimension is relatively straightforward. Each cell in the domain is initialized to some value. Using the cell quantities to the left and right of each cell interface as the left and right states, the appropriate Roe averaged variables, eigenvalues, and eigenvectors are calculated. The flux at each interface is then calculated using one of Eqs. (5.48, 5.49, or 5.50), and the solution advanced in time using the appropriate form of Eqs. (5.8):

$$\hat{\mathbf{q}}_i^{n+1} = \hat{\mathbf{q}}_i^n - \frac{\Delta \tau}{\Delta \xi} (\bar{\mathbf{f}}_{i+1/2}^n - \bar{\mathbf{f}}_{i-1/2}^n). \quad (5.71)$$

5.3 Extension of the Roe Scheme to Second-Order Accuracy

The scheme presented in the previous section was first-order accurate in both time and space. The first order spatial accuracy is the result of assuming piece-wise constant states in each cell, which is entirely decoupled from the Riemann solution step. Second-order accuracy has been achieved for these types of schemes in two ways:

extrapolating the cell-centered dependent variables to the cell interfaces, and extrapolating the cell-centered fluxes to the cell interfaces [40]. The spatial accuracy of the scheme then depends upon the accuracy of the extrapolation; assuming a linear variation results in second-order accuracy, while a quadratic variation leads to third-order accuracy. Essentially, the accuracy of these schemes is increased by using more points, much like a finite-difference method.

The second-order scheme chosen for this research was based upon flux extrapolation. The linear extrapolation of a variable s (see Fig. 5.7) to the interface $i + 1/2$ is

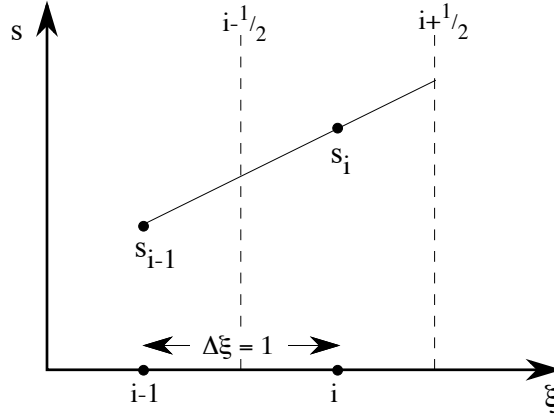


Figure 5.7: Linear extrapolation for $s_{i+1/2}$.

simply

$$s_{i+1/2} - s_i = m_s(\xi_{i+1/2} - \xi_i). \quad (5.72)$$

Using the cell-centered values, $m_s = (s_i - s_{i-1})/\Delta\xi = s_i - s_{i-1}$ and $\xi_{i+1/2} - \xi_i = \Delta\xi/2 = 1/2$, since $\Delta\xi \equiv 1$. This leads to

$$s_{i+1/2} = s_i + \frac{1}{2}(s_i - s_{i-1}). \quad (5.73)$$

The general second-order scheme can be derived as follows [40]: the flux at a cell center can be divided into contributions from waves travelling along positive and negative characteristics

$$\hat{\mathbf{f}}_i = \hat{\mathbf{f}}_i^- + \hat{\mathbf{f}}_i^+. \quad (5.74)$$

One way of obtaining a higher-order flux at the interface $i + 1/2$ is to linearly extrapolate fluxes from positive (rightward travelling) characteristics using cells i and $i - 1$,

$$\hat{\mathbf{f}}_{i+1/2}^{+b} = \hat{\mathbf{f}}_i^+ + \frac{1}{2}(\hat{\mathbf{f}}_i^+ - \hat{\mathbf{f}}_{i-1}^+), \quad (5.75)$$

while linearly extrapolating fluxes from negative (leftward travelling) characteristics using cells $i + 1$ and $i + 2$,

$$\hat{\mathbf{f}}_{i+1/2}^{-f} = \hat{\mathbf{f}}_{i+1}^- - \frac{1}{2}(\hat{\mathbf{f}}_{i+2}^- - \hat{\mathbf{f}}_{i+1}^-). \quad (5.76)$$

The second-order flux at the interface is then (from Eqs. (5.74))

$$\begin{aligned} \bar{\mathbf{f}}_{i+1/2}^{(2)} &= \hat{\mathbf{f}}_{i+1/2}^{-f} + \hat{\mathbf{f}}_{i+1/2}^{+b} \\ &= \hat{\mathbf{f}}_{i+1}^- + \hat{\mathbf{f}}_i^+ - \frac{1}{2}(\hat{\mathbf{f}}_{i+2}^- - \hat{\mathbf{f}}_{i+1}^-) + \frac{1}{2}(\hat{\mathbf{f}}_i^+ - \hat{\mathbf{f}}_{i-1}^+) \end{aligned} \quad (5.77)$$

To relate this to the first-order scheme, the differences in the positive and negative fluxes can be defined as the difference between a first-order numerical flux and the physical flux

$$\bar{\mathbf{f}}_{i+1/2}^{(1)} - \hat{\mathbf{f}}_i = \hat{\mathbf{f}}_{i+1}^- - \hat{\mathbf{f}}_i^-, \quad (5.78)$$

$$\hat{\mathbf{f}}_i - \bar{\mathbf{f}}_{i-1/2}^{(1)} = \hat{\mathbf{f}}_i^+ - \hat{\mathbf{f}}_{i-1}^+. \quad (5.79)$$

Using Eqs. (5.78) and (5.79) in Eqs. (5.77) results in

$$\bar{\mathbf{f}}_{i+1/2}^{(2)} = \bar{\mathbf{f}}_{i+1/2}^{(1)} + \frac{1}{2}(\hat{\mathbf{f}}_{i+1}^- - \bar{\mathbf{f}}_{i+3/2}^{(1)}) + \frac{1}{2}(\hat{\mathbf{f}}_i^+ - \bar{\mathbf{f}}_{i-1/2}^{(1)}). \quad (5.80)$$

The second-order flux using Roe's first-order flux is simply

$$\bar{\mathbf{f}}_{i+1/2}^{(2)} = \bar{\mathbf{f}}_{i+1/2}^{(1)} + \frac{1}{2}(\hat{\mathbf{f}}_{i+1}^- - \bar{\mathbf{f}}_{i+3/2}^{(1)}) + \frac{1}{2}(\hat{\mathbf{f}}_i^+ - \bar{\mathbf{f}}_{i-1/2}^{(1)}), \quad (5.81)$$

which can be written as

$$\begin{aligned} \bar{\mathbf{f}}_{i+1/2}^{(2)} &= \bar{\mathbf{f}}_{i+1/2}^{(1)} + \frac{1}{2}[(\sum_{\bar{\lambda}_m > 0} \alpha_m \bar{\lambda}_m \bar{\mathbf{F}}_m)_{i-1/2} - (\sum_{\bar{\lambda}_m < 0} \alpha_m \bar{\lambda}_m \bar{\mathbf{F}}_m)_{i+3/2}], \\ &= \bar{\mathbf{f}}_{i+1/2}^{(1)} + \frac{1}{2}[\sum_{m=1}^N d\hat{\mathbf{f}}_{i-1/2}^{m+} - \sum_{m=1}^N d\hat{\mathbf{f}}_{i+3/2}^{m-}] \end{aligned} \quad (5.82)$$

using Eqs. (5.48) and (5.49). Thus, the second-order interface flux can be seen to be the first-order flux plus corrections from the surrounding flux values.

Unfortunately, replacing the first-order flux terms in Eqs. (5.71) with the second-order flux as defined by Eqs. (5.82) results in solutions that have unphysical oscillations near discontinuities [40]. This is a general property of linear, second-order upwind schemes [26]. One approach for avoiding this behavior is to “limit” the schemes in such a way as to make them only first-order accurate near discontinuities, while remaining second-order accurate in smooth regions of the flow. This is the general approach for so-called high resolution methods [50], which have second-order accuracy in smooth regions of the flow and resolve discontinuities crisply and without oscillations.

Hirsch [40] contains a fairly complete review of the basis for high resolution methods; what follows is a brief synopsis of the underlying ideas as presented by Hirsch. It can be shown in one dimension that physical solutions of the Euler and Navier-Stokes equations do not admit new extrema in the evolution of the flow; this is thought to be true of multidimensional flows also. Thus, physical solutions of the conservation equations are monotone, and therefore numerical approximations to the solutions should also be monotone. Godunov [31], however, showed that monotone linear schemes are at most first-order accurate. This is a severe restriction, since first-order schemes are very dissipative and have solutions with very smeared shocks. This suggests that in order to achieve higher accuracy, the numerical scheme must use nonlinear limiting.

A less restrictive criteria than monotonicity is based on the solutions of a scalar, one-dimensional conservation law: any physical solution has the property that its total variation

$$\text{TV}(q) = \int \left| \frac{\partial q}{\partial \xi} \right| d\xi, \quad (5.83)$$

does not increase with time. For a discrete solution, the total variation can be ex-

pressed as

$$\text{TV}(q) = \sum_m |q_{m+1}^n - q_m^n|, \quad (5.84)$$

and a numerical scheme is described as total variation diminishing (TVD) if

$$\text{TV}(q^{n+1}) \leq \text{TV}(q^n). \quad (5.85)$$

A scheme is described as monotonicity preserving if in the evolution of the solution no new local extrema are created, and a local minimum or maximum is non-decreasing or non-increasing, respectively. Put another way, if q^n is monotone, then q^{n+1} will also be monotone. The monotonicity and TVD property are related in the following way: all monotone schemes are TVD, and all TVD schemes are monotonicity preserving. As alluded to above, a scheme can be made TVD by using limiting functions (“limiters”) to reduce to first-order accuracy near discontinuities. For linear limiters (and thus a linear scheme), monotonicity preservation leads to the same conditions as monotonicity, and thus the scheme is first-order accurate. This does not apply to nonlinear schemes, however. Thus, if a second-order scheme can be made TVD using nonlinear limiters, it will be monotonicity preserving and remain higher-order in smooth regions of the flow. Since the initial data for the scheme can be specified as monotone, any resulting solution will be monotone and (presumably) physically correct. This presumes that the scheme satisfies the entropy condition, which is the last criteria for a high resolution scheme. In fact, for the case of an rarefaction through a sonic point, Roe’s scheme does not satisfy the entropy condition and a correction must be included [40]. Except for a one-dimensional test case, the problems studied in this research did not have any rarefactions of this type, so the correction was not included in the general two-dimensional code.

Chakravarthy and Osher [17] have determined a “TVD” form of Eqs. (5.82). They note that the TVD property can so far only be rigorously proven for scalar equations or systems of linear equations in one (spatial) dimension. Their TVD formulation of

Eqs. (5.82) is

$$\bar{\mathbf{f}}_{i+1/2}^{(2)} = \bar{\mathbf{f}}_{i+1/2} + \frac{1}{2} \left[\sum_{m=1}^N d\check{\mathbf{f}}_{i-1/2}^{m+} - \sum_{m=1}^N d\check{\mathbf{f}}_{i+3/2}^{m-} \right], \quad (5.86)$$

where $d\check{\mathbf{f}}$ and $d\check{\check{\mathbf{f}}}$ may now be limited based upon neighboring values. The limited flux increments are defined as follows:

$$d\check{\mathbf{f}}_{i-1/2}^{m+} = \check{\sigma}_{i-1/2}^{m+}(\bar{\mathbf{r}}_m)_{i+1/2}, \quad (5.87)$$

$$d\check{\check{\mathbf{f}}}_{i+3/2}^{m-} = \check{\check{\sigma}}_{i+3/2}^{m-}(\bar{\mathbf{r}}_m)_{i+1/2}, \quad (5.88)$$

where

$$\check{\sigma}_{i-1/2}^{m+} = L[\sigma_{i-1/2}^{m+}, \sigma_{i+1/2}^{m+}], \quad (5.89)$$

$$\check{\check{\sigma}}_{i+3/2}^{m-} = L[\sigma_{i+3/2}^{m-}, \sigma_{i+1/2}^{m-}], \quad (5.90)$$

and

$$\sigma_{i+1/2}^{m\pm} = (\alpha_m \bar{\lambda}_{m\pm})_{i+1/2}. \quad (5.91)$$

The $L[\]$ indicates a nonlinear limiting function, and $\bar{\lambda}_{m\pm}$ indicates eigenvalues with positive or negative signs, respectively. The choice of a limiting function is not unique; Whitfield [84] suggests one of the following:

$$\begin{aligned} \text{Superbee : } \quad L[a, b] &= \text{sign}(a) \max\{0, \min[|a|, \psi b \text{sign}(a)], \min[\psi|a|, b \text{sign}(a)]\} \\ &\quad \text{where } 1 \leq \psi \leq 2, \end{aligned} \quad (5.92)$$

$$\text{van Leer : } \quad L[a, b] = \frac{ab + |ab|}{a + b}. \quad (5.93)$$

In the next chapter the Riemann problem was used to differentiate the solutions obtained with the various limiters; based upon this the van Leer limiter was chosen for the numerical studies.

The implementation of the second-order scheme remains the same, except the first-order fluxes are simply replaced by their second-order counterparts:

$$\hat{\mathbf{q}}_i^{n+1} = \hat{\mathbf{q}}_i^n - \frac{\Delta\tau}{\Delta\xi} (\bar{\mathbf{f}}_{i+1/2}^{(2)n} - \bar{\mathbf{f}}_{i-1/2}^{(2)n}). \quad (5.94)$$

For both the first and second order schemes, the time step is determined by the restriction that Roe's approximate Riemann solution does not account for the interaction of multiple Riemann problems. Thus, the time step must be chosen such that the Riemann problems at neighboring cell interfaces do not interact. This can be done by basing the time step on the time it takes for the fastest wave in the domain to cross half of a computational cell. Since the waves propagate with a speed equal to the magnitude of the eigenvalues, and $\Delta\xi = \Delta\eta \equiv 1$

$$\Delta\tau = \frac{0.5}{\max[\max(k_{i,j}), \max(l_{i,j})]}. \quad (5.95)$$

where

$$k_{i,j} = [\max(|\bar{\lambda}_1|, |\bar{\lambda}_2|, \dots, |\bar{\lambda}_5|)]_{i,j} \quad (5.96)$$

$$l_{i,j} = [\max(|\bar{\mu}_1|, |\bar{\mu}_2|, \dots, |\bar{\mu}_5|)]_{i,j} \quad (5.97)$$

In practice, to obtain a slightly more restrictive criteria, the following formulation was used

$$\Delta\tau = \frac{0.5}{\max(m)}, \quad (5.98)$$

where $m_{i,j} = \sqrt{k_{i,j}^2 + l_{i,j}^2}$. Rearranging this

$$\frac{\max(m) \Delta\tau}{\Delta\xi} = 0.5 = \text{CFL}, \quad (5.99)$$

it is apparent that the left-hand side is the Courant-Friedrichs-Lewy (CFL) criteria. Thus, the maximum attainable CFL number for this scheme is 0.5; typically a more conservative estimate of $\text{CFL} = 0.4$ was used in the numerical studies.

Higher temporal accuracy may be achieved by using a two step procedure, in this case a second-order Runge-Kutta scheme [65]. Basically, the solution is advanced a half time step, and the resulting information is used to obtain an improved approximation to the solution over an entire time step. This is illustrated as follows:

$$\hat{\mathbf{q}}_i^\dagger = \hat{\mathbf{q}}_i^n - 0.5 \frac{\Delta\tau}{\Delta\xi} (\bar{\mathbf{f}}_{i+1/2}^{(2)n} - \bar{\mathbf{f}}_{i-1/2}^{(2)n}) \quad (5.100)$$

$$\hat{\mathbf{q}}_i^{n+1} = \hat{\mathbf{q}}_i^n - \frac{\Delta\tau}{\Delta\xi} (\bar{\mathbf{f}}_{i+1/2}^{(2)\dagger} - \bar{\mathbf{f}}_{i-1/2}^{(2)\dagger}), \quad (5.101)$$

where $\hat{\mathbf{f}}^{(2)\dagger}$ is the second-order flux determined from the half time step solution, $\hat{\mathbf{q}}^\dagger$.

5.4 The Roe Scheme for the Reactive Euler Equations in a Two-Dimensional, Curvilinear Coordinate System

The proceeding development was essentially for the one-dimensional Euler equations. The following issues remain to be addressed in order to obtain solutions of the reactive Euler equations: the scheme must be extended to more than one spatial dimension, the source term in the reactive Euler equations must be incorporated into the scheme, and lastly, inflow, outflow, and wall boundary conditions must be implemented in a rational way.

Fortuitously, a method exists whereby the first and second issues may be addressed; it is commonly called fractional stepping [40, 50]. Instead of solving the complete two-dimensional problem in one step, it is broken into a number of one-dimensional problems in the respective coordinate directions. Each of these is then solved one after another to arrive at the final solution. Thus, Eqs. (3.48) would become

$$\frac{\partial \hat{\mathbf{q}}}{\partial \tau} + \frac{\partial \hat{\mathbf{f}}}{\partial \xi} = 0, \quad (5.102)$$

$$\frac{\partial \hat{\mathbf{q}}}{\partial \tau} + \frac{\partial \hat{\mathbf{g}}}{\partial \eta} = 0, \quad (5.103)$$

$$\frac{\partial \hat{\mathbf{q}}}{\partial \tau} = \hat{\mathbf{w}}. \quad (5.104)$$

In the solution of the first two equations the chemistry is assumed to be frozen, and thus Roe's scheme can be applied directly for each direction. The Roe averaged versions of the eigenvalue and eigenvector arrays (Eqs. (3.54, 3.55, 3.59)) would be used for the solution of Eqs. (5.102), while the Roe averaged versions of Eqs. (3.60, 3.61, 3.62) would be used in the solution of Eqs. (5.103).

Colella, *et al.* [20] suggest solving Eqs. (5.104) analytically. Writing out the individual equations

$$\frac{d}{d\tau} \left(\frac{\rho}{J} \right) = 0, \quad (5.105)$$

$$\frac{d}{d\tau} \left(\frac{\rho u}{J} \right) = 0, \quad (5.106)$$

$$\frac{d}{d\tau} \left(\frac{\rho v}{J} \right) = 0, \quad (5.107)$$

$$\frac{d}{d\tau} \left(\frac{\rho E}{J} \right) = 0, \quad (5.108)$$

$$\frac{d}{d\tau} \left(\frac{\rho Y_2}{J} \right) = \frac{\rho}{J} (1 - Y_2) e^{-\Theta \rho/p}, \quad (5.109)$$

it is apparent that ρ , u , v , and E are constant (J is constant for a fixed grid). Equation (5.109) can be simplified by bringing ρ/J out of the derivative

$$\frac{dY_2}{d\tau} = (1 - Y_2) e^{-\Theta \rho/p}, \quad (5.110)$$

which makes it apparent that the equation is separable and can be integrated directly

$$\int \frac{dY_2}{1 - Y_2} = \int e^{-\Theta \rho/p} d\tau. \quad (5.111)$$

The left-hand side can be integrated to give

$$\int_{Y_2^n}^{Y_2^{n+1}} \frac{dY_2}{1 - Y_2} = -\log(1 - Y_2) \Big|_{Y_2^n}^{Y_2^{n+1}} = \log \left(\frac{1 - Y_2^n}{1 - Y_2^{n+1}} \right), \quad (5.112)$$

and, assuming p remains constant over $\Delta\tau$, the right-hand side can be integrated to

$$\int_{\tau^n}^{\tau^{n+1}} e^{-\Theta \rho/p} d\tau = e^{-\Theta \rho^n/p^n} \Delta\tau. \quad (5.113)$$

Setting Eq. (5.112) equal to Eq. (5.113) and solving for Y_2^{n+1} results in

$$Y_2^{n+1} = 1 + (Y_2^n - 1) e^{-\Delta\tau \exp(-\Theta \rho^n/p^n)}. \quad (5.114)$$

The fractional stepping procedure can be written as follows: let $\mathcal{L}_\xi^{\Delta\tau}$, $\mathcal{L}_\eta^{\Delta\tau}$, and $\mathcal{L}_Y^{\Delta\tau}$ be the solution operators for Eqs. (5.102, 5.103, 5.104), respectively, over time $\Delta\tau$. One fractional stepping scheme would be

$$\hat{\mathbf{q}}^{n+1} = \mathcal{L}_Y^{\Delta\tau} \mathcal{L}_\eta^{\Delta\tau} \mathcal{L}_\xi^{\Delta\tau} \hat{\mathbf{q}}^n. \quad (5.115)$$

Unfortunately, splitting the equations in this manner results in a scheme that has only first-order temporal accuracy [50], *i.e.* even though each step of the solution has second-order temporal accuracy (or better), the actual splitting has an error of $O(\Delta\tau)$. Strang [75] found that the following type of splitting retained second-order temporal accuracy:

$$\hat{\mathbf{q}}^{n+1} = \mathcal{L}_\xi^{\Delta\tau/2} \mathcal{L}_\eta^{\Delta\tau/2} \mathcal{L}_Y^{\Delta\tau/2} \mathcal{L}_Y^{\Delta\tau/2} \mathcal{L}_\eta^{\Delta\tau/2} \mathcal{L}_\xi^{\Delta\tau/2} \hat{\mathbf{q}}^n. \quad (5.116)$$

Since the time step restriction for each step of the solution is still $\Delta\tau$, Colella, *et al.* [20] suggest the following for maximum efficiency:

$$\hat{\mathbf{q}}^{n+2} = \mathcal{L}_\xi^{\Delta\tau} \mathcal{L}_\eta^{\Delta\tau} \mathcal{L}_Y^{\Delta\tau} \mathcal{L}_Y^{\Delta\tau} \mathcal{L}_\eta^{\Delta\tau} \mathcal{L}_\xi^{\Delta\tau} \hat{\mathbf{q}}^n. \quad (5.117)$$

Thus, using the second-order Roe fluxes and the second-order Runge-Kutta time stepping with the splitting procedure of Eqs. (5.117) results in a scheme with second-order spatial and temporal accuracy in smooth regions of the flow.

The issue of boundary conditions is somewhat more difficult to address. Figure 5.8 indicates the types of boundary conditions needed for the various boundaries of the computational space. The flow was considered to be entering the domain from

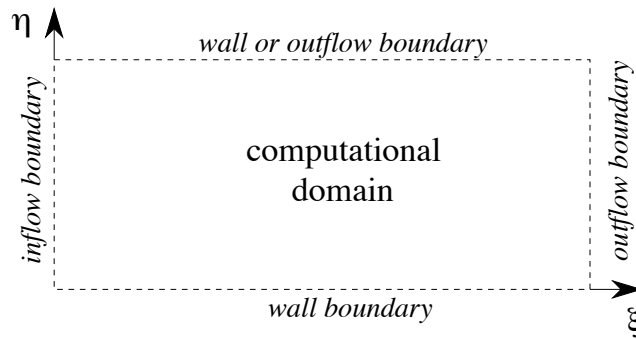


Figure 5.8: Types of conditions required for the computational domain boundaries.

the left at the inflow boundary, and exiting the domain on the right at the outflow boundary. The incoming flow was always supersonic, while the outgoing flow could be either supersonic or subsonic. An impenetrable wall was considered to be along

the bottom of the domain, while the top of the domain was treated as either a wall or another outflow boundary.

To date, the best approach for handling inflow and outflow boundary conditions is based on the characteristics of the one-dimensional form of the governing equations [58]. Essentially, the sign of the characteristics at the boundary determine the direction in which information is physically propagating. For example, at the inflow boundary positive characteristics indicate that information is propagating into the domain, while negative characteristics indicate information is propagating out of the domain. Therefore, the boundary conditions that are used must adhere to this physical mechanism. The inflow boundary is straightforward: all characteristics are positive for supersonic inflow, therefore information is only propagating into the domain. Numerically, this requires that all of the flow variables be specified at the inflow boundary.

The right outflow boundary is less trivial. For a supersonic outflow, all characteristics point out of the domain, while for a subsonic outflow four characteristics point out and one points in. Physically this means that information is only propagating out of the domain in a supersonic outflow, but both out of and into the domain for a subsonic outflow. Numerically, a supersonic outflow requires that the variables at the outflow boundary be entirely specified using quantities from within the domain. The numerical condition for a subsonic outflow is very difficult, however, because it requires information from outside the domain that is unknown (at least for the cases studied in this research). Poinso and Lele [58] suggest treating the outgoing waves with the characteristic method of Thompson [76], and estimating the magnitude of the incoming waves with a simple equation based upon pressure. The following development is based upon this approach.

For the general, two-dimensional reactive (or inert) Euler equations, there is no characteristic form directly analogous to a one-dimensional system (for example, Eqs.

(5.19 or 5.21)). However, a reasonably close approximation suitable for determining characteristically-based boundary conditions is possible. The vector conservative form of the conservation equations (Eqs. (3.46)) may be written in the following form

$$\mathbf{M}\frac{\partial\mathbf{p}}{\partial\tau} + \mathbf{G}\frac{\partial\mathbf{p}}{\partial\xi} + \mathbf{Z}\frac{\partial\mathbf{p}}{\partial\eta} = \hat{\mathbf{w}}, \quad (5.118)$$

where $\mathbf{p} = [\rho, u, v, p, Y_2]^T$ is the vector of primitive variables, and $\mathbf{M} = \partial\hat{\mathbf{q}}/\partial\mathbf{p}$, $\mathbf{G} = \partial\hat{\mathbf{f}}/\partial\mathbf{p}$, and $\mathbf{Z} = \partial\hat{\mathbf{g}}/\partial\mathbf{p}$ are Jacobian matrices. The conservation equations in terms of the primitive variables are then simply

$$\frac{\partial\mathbf{p}}{\partial\tau} + \mathbf{H}\frac{\partial\mathbf{p}}{\partial\xi} + \mathbf{K}\frac{\partial\mathbf{p}}{\partial\eta} = \mathbf{b}, \quad (5.119)$$

where $\mathbf{H} = \mathbf{M}^{-1}\mathbf{G}$, $\mathbf{K} = \mathbf{M}^{-1}\mathbf{Z}$, and $\mathbf{b} = \mathbf{M}^{-1}\hat{\mathbf{w}}$. For the outflow boundary condition desired, the ξ direction is of interest. Since

$$\frac{\partial\hat{\mathbf{f}}}{\partial\xi} = \mathbf{G}\frac{\partial\mathbf{p}}{\partial\xi}, \quad (5.120)$$

and $\mathbf{G} = \mathbf{M}\mathbf{H}$, the conservative form of the equations (Eqs. (3.46)) may be written

$$\frac{\partial\hat{\mathbf{q}}}{\partial\tau} + \mathbf{M}\mathbf{H}\frac{\partial\mathbf{p}}{\partial\xi} + \frac{\partial\hat{\mathbf{g}}}{\partial\eta} = \hat{\mathbf{w}}. \quad (5.121)$$

The characteristic information is obtained by considering the eigenvalue problem

$$\mathbf{H}\mathbf{Y} = \mathbf{Y}\boldsymbol{\zeta}, \quad (5.122)$$

where, as before, \mathbf{Y} is a right eigenvector array, and $\boldsymbol{\zeta}$ is the diagonal array of eigenvalues. In fact, $\boldsymbol{\zeta}$ has the same elements as $\boldsymbol{\lambda}$, except they are in a different order.

Using this in Eqs. (5.121) results in

$$\frac{\partial\hat{\mathbf{q}}}{\partial\tau} + \mathbf{M}\mathbf{Y}(\boldsymbol{\zeta}\mathbf{Y}^{-1}\frac{\partial\mathbf{p}}{\partial\xi}) + \frac{\partial\hat{\mathbf{g}}}{\partial\eta} = \hat{\mathbf{w}}. \quad (5.123)$$

Defining

$$\mathcal{P} = \boldsymbol{\zeta}\mathbf{Y}^{-1}\frac{\partial\mathbf{p}}{\partial\xi}, \quad (5.124)$$

and examining the form of its terms more closely

$$\mathcal{P}_i = \zeta_i \sum_{m=1}^5 \mathbf{Y}_{i,m}^{-1} \frac{\partial p_m}{\partial \xi}, \quad (5.125)$$

it is apparent that each eigenvalue multiplies the sum of the ξ derivatives and their coefficients (note: there are no derivatives in the components of \mathbf{Y}^{-1}). The outflow boundary variables are then calculated as follows: for each $\zeta_i > 0$, \mathcal{P}_i is calculated from Eq. (5.125) using upwind differencing for the derivatives; for the one possible $\zeta_i < 0$, \mathcal{P}_i is estimated by

$$\mathcal{P}_i = K(p - p_\infty), \quad (5.126)$$

where p_∞ is a constant far-field pressure, and K is a constant. For the cases studied here, $p_\infty = p_0$, the freestream pressure. The \mathcal{P}_i are then used in Eqs. (5.123), which are solved using the same technique as the points on the interior of the domain. For the method employed here, Eqs. (5.102) was solved using second-order upwind finite differencing for the spatial term, and explicit Euler time differencing. Equations (5.103 and 5.104) were solved as described previously. If the top of the computational domain (see Fig. 5.8) is considered an outflow boundary, $\partial \hat{\mathbf{g}} / \partial \eta$ is replaced by $\mathbf{MK}(\partial \mathbf{p} / \partial \eta)$ in Eqs. (3.46) and the same type of analysis applied.

The remaining boundary conditions to be determined lie along the top and bottom of the domain. For the case of a solid wall at this boundary, Fig. (5.9) illustrates one of the computational cells. The standard wall condition for inviscid equations is flow tangency, *i.e.* there is no flow through the wall. In the curvilinear coordinate system this results in $V^c = 0$ at the wall. The numerical method requires the flux at the wall, which would be

$$\hat{\mathbf{g}}_{i,j-1/2} = J^{-1} \begin{bmatrix} \rho V^c \\ \rho u V^c + \eta_x p \\ \rho v V^c + \eta_y p \\ \rho H V^c - \eta_t p \\ \rho Y_2 V^c \end{bmatrix}_{i,j-1/2} = J^{-1} \begin{bmatrix} 0 \\ \eta_x p \\ \eta_y p \\ 0 \\ 0 \end{bmatrix}_{i,j-1/2}, \quad (5.127)$$

from the definition of $\hat{\mathbf{g}}$ (recall that $\eta_t = 0$). Unfortunately, the value of p is only

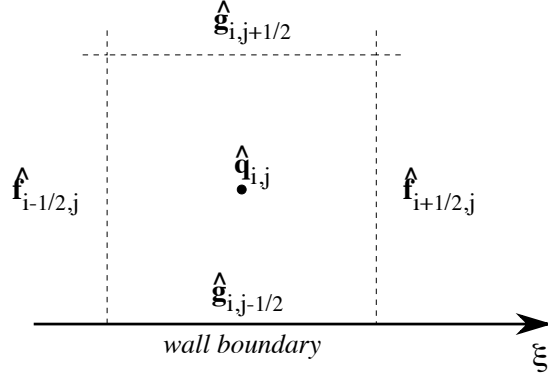


Figure 5.9: Schematic of wall boundary computational cell.

determined at the cell center. The most straightforward form of the boundary condition would be to take $p_{i,j-1/2} = p_{i,j}$. This, however, leads to severe overshoots near shocks and rarefactions for the cell-centered flow variables along the wall.

A better, more stable approach utilizing one-dimensional Riemann invariants was suggested by Dadone and Grossman [23]. Consider the one-dimensional, nonreactive form of Eqs. (5.119)

$$\frac{\partial \mathbf{p}}{\partial \tau} + \mathbf{K} \frac{\partial \mathbf{p}}{\partial \eta} = 0. \quad (5.128)$$

Using the eigenvalue problem (in this case $\boldsymbol{\kappa}$ has the same elements as $\boldsymbol{\mu}$, except in a different order)

$$\mathbf{K}\mathbf{X} = \mathbf{X}\boldsymbol{\kappa}, \quad (5.129)$$

this can be written in characteristic form

$$\mathbf{X}^{-1} \frac{\partial \mathbf{p}}{\partial \tau} + \boldsymbol{\kappa} \mathbf{X}^{-1} \frac{\partial \mathbf{p}}{\partial \eta} = 0. \quad (5.130)$$

The fifth equation in this system is

$$\eta_x \frac{\partial u}{\partial \tau} + \eta_y \frac{\partial v}{\partial \tau} - \frac{\sqrt{\eta_x^2 + \eta_y^2}}{\rho c} \frac{\partial p}{\partial \tau} + (V^c - c\sqrt{\eta_x^2 + \eta_y^2}) \left[\eta_x \frac{\partial u}{\partial \eta} + \eta_y \frac{\partial v}{\partial \eta} - \frac{\sqrt{\eta_x^2 + \eta_y^2}}{\rho c} \frac{\partial p}{\partial \eta} \right] = 0. \quad (5.131)$$

Along $d\eta/d\tau = V^c - c\sqrt{\eta_x^2 + \eta_y^2}$ this reduces to

$$\eta_x \frac{du}{d\tau} + \eta_y \frac{dv}{d\tau} - \frac{\sqrt{\eta_x^2 + \eta_y^2}}{\rho c} \frac{dp}{d\tau} = 0, \quad (5.132)$$

or

$$\eta_x du + \eta_u dv - \frac{\sqrt{\eta_x^2 + \eta_y^2}}{\rho c} dp = 0, \quad (5.133)$$

which can be further simplified to

$$dV^c - \frac{\sqrt{\eta_x^2 + \eta_y^2}}{\rho c} dp = 0. \quad (5.134)$$

This is one of the Riemann invariants; a second Riemann invariant along $d\eta/d\tau = V^c + c\sqrt{\eta_x^2 + \eta_y^2}$ can be obtained from the fourth equation of Eqs. (5.130):

$$dV^c + \frac{\sqrt{\eta_x^2 + \eta_y^2}}{\rho c} dp = 0. \quad (5.135)$$

At this point it is assumed the “downward” travelling characteristic from the cell center ($d\eta/d\tau = V^c - c\sqrt{\eta_x^2 + \eta_y^2} < 0$ since V^c is small near the wall) intersects with the wall at $(i, j - 1/2)$. Thus, Eq. (5.134) holds, and can be approximated by

$$(V_{i,j}^c - V_{i,j-1/2}^c) - \left[\frac{\sqrt{\eta_x^2 + \eta_y^2}}{\rho c} \right]_{i,j} (p_{i,j} - p_{i,j-1/2}) = 0. \quad (5.136)$$

This can be solved to estimate the pressure at the wall

$$p_{i,j-1/2} = p_{i,j} - \left[\frac{V^c \rho c}{\sqrt{\eta_x^2 + \eta_y^2}} \right]_{i,j}, \quad (5.137)$$

since all quantities at (i, j) are known. For the case in which the top boundary of the computational domain is considered a wall, Eq. (5.135) would hold and the estimated wall pressure would be

$$p_{i,j+1/2} = p_{i,j} + \left[\frac{V^c \rho c}{\sqrt{\eta_x^2 + \eta_y^2}} \right]_{i,j}. \quad (5.138)$$

Chapter 6

VERIFICATION OF NUMERICAL ALGORITHM

This chapter will present the results obtained for various one- and two-dimensional test problems used to verify the accuracy of the numerical algorithm. The following test cases were considered: the Riemann problem, steady and unsteady ZND detonations, inert supersonic flow over a wedge, and the straight wall, curved oblique detonation problem of Powers and Stewart [63]. Of the four test cases considered, only the unsteady ZND detonation has no known analytical solution; it has been studied extensively in the literature, however, so detailed numerical solutions are available.

6.1 Riemann Problem

Figures 6.1, 6.2, and 6.3 show comparisons of the the first- and second-order Roe scheme solutions with the exact Riemann solution. All solutions here and in the rest of this section used a common equally spaced grid of 200 points and $CFL = 0.4$. All of the numerical studies here and in the remainder of the dissertation were performed on IBM RS/6000 workstations. Because of the contact discontinuity and the shock, the density solution is the hardest for the scheme to resolve. The first-order scheme can be seen to be very dissipative, smearing the contact discontinuity rather severely, and to a lesser degree, the shock. It does not, however, show any evidence of unphysical oscillations near either discontinuity. The solutions for velocity and pressure show

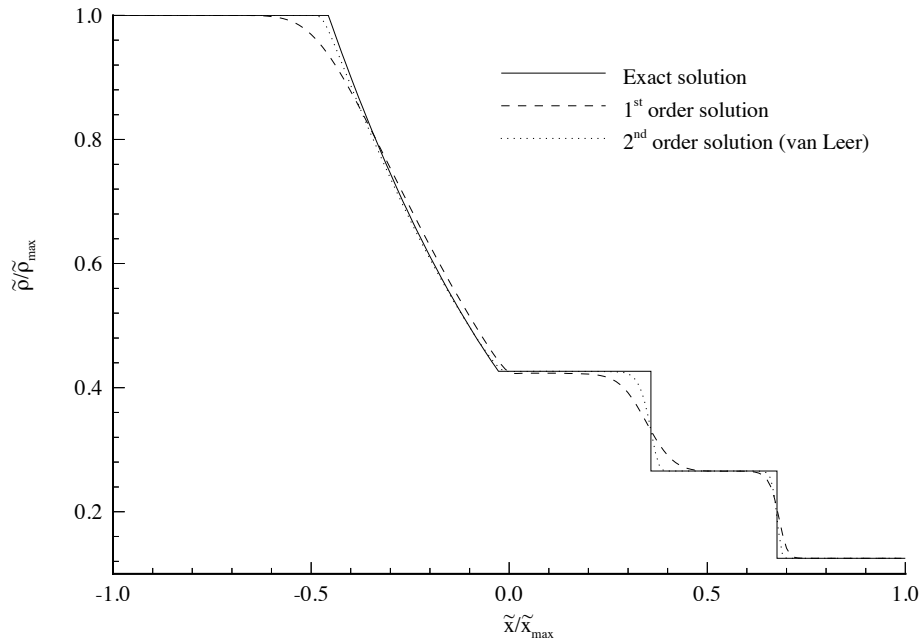


Figure 6.1: Comparison of Roe scheme solutions with the exact density solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).

the same behavior for the first-order scheme.

As expected, the second-order solution (using the van Leer limiter) is much more accurate. The contact discontinuity and shock both exhibit much steeper profiles. Though it is not apparent from this line plot, the shock was resolved in approximately 3 points and the contact discontinuity in 6 points. For the first-order scheme these figures were approximately 13 and 30, respectively. Similar improvements are seen in the velocity and pressure plots. The limiting functions work very well; there are no apparent oscillations in the second-order solution. Overall, the slight increase in computational effort required to obtain the second-order results is easily offset by the significant improvement in the solution.

For completeness, Fig. 6.4 shows the solutions obtained for a Riemann problem with much larger initial pressure and density ratios. In this case there is an expansion through a sonic point, and a correction must be added to both the first- and second-

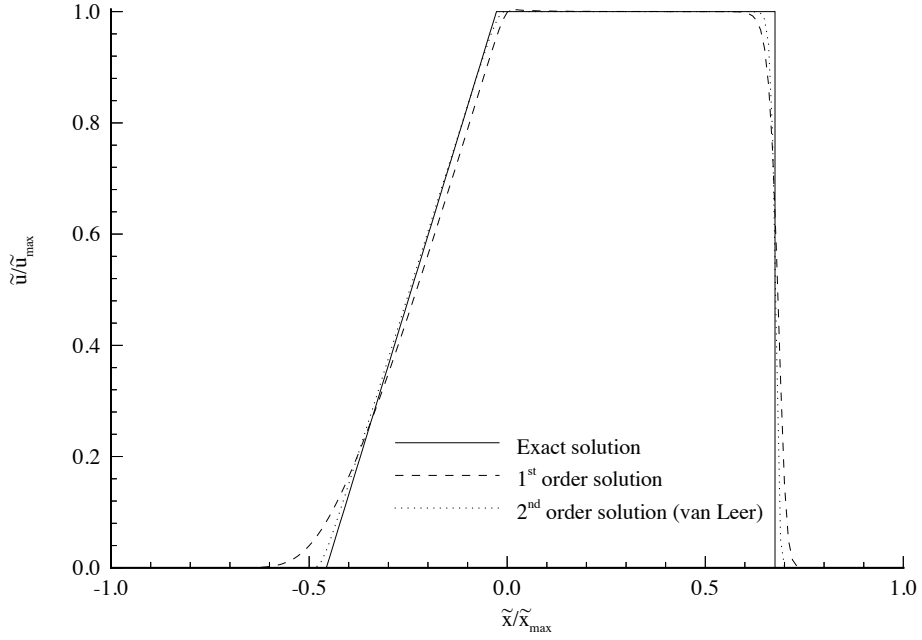


Figure 6.2: Comparison of Roe scheme solutions with the exact velocity solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).

order Roe schemes. The following correction suggested by Harten and Hyman [36] was used:

$$|\bar{\lambda}_m|_{\text{mod}} = \begin{cases} |\bar{\lambda}_m|_{i+1/2} & \text{if } |\bar{\lambda}_m|_{i+1/2} \geq \varepsilon \\ \varepsilon & \text{if } |\bar{\lambda}_m|_{i+1/2} < \varepsilon \end{cases} \quad (6.1)$$

where

$$\varepsilon = \max\{0, [(\bar{\lambda}_m)_{i+1/2} - (\lambda_m)_i][(\lambda_m)_{i+1} - (\bar{\lambda}_m)_{i+1/2}]\}. \quad (6.2)$$

This is described as introducing a local expansion fan in the approximate Riemann solution when a rarefaction through a sonic point is detected [40]. As can be seen in the figure, the correction works very well; there is no detrimental effect on either the first- or second-order solution.

Figure 6.5 shows a comparison of the second-order scheme using the van Leer and Superbee limiters. Three different values of the parameter ψ were used with the Superbee limiter. Through the expansion region and the shock, both limiters

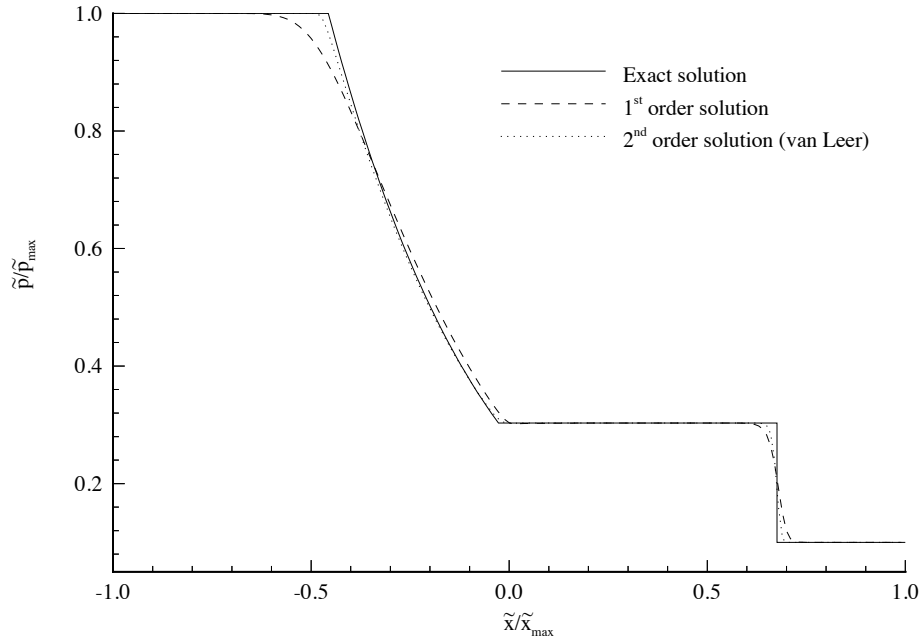


Figure 6.3: Comparison of Roe scheme solutions with the exact pressure solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).

resulted in nearly identical solutions regardless of the value chosen for ψ . Through the contact discontinuity and at the beginning of the expansion region the solutions differ slightly. Figure 6.6 shows an expanded view of the contact discontinuity and shock of Fig. 6.5. The solution obtained using the Superbee limiter with $\psi = 2.0$ is the best, and the solution for $\psi = 1.0$ is the worst; the van Leer limited solution falls in between. Overall, the solutions obtained with both limiters are quite good. In terms of complexity (see Eqs. (5.92) and (5.93)), the Superbee limiter has a much higher operation count than the van Leer limiter. Since the limiter is used in the innermost loop of the program, a significantly higher operation count could result in significantly longer run times for the simulation, particularly for large two-dimensional problems. For this reason, and the fact that the van Leer limited solution is only slightly worse than the best Superbee limited solution, the van Leer limiter was used for the remainder of the study.

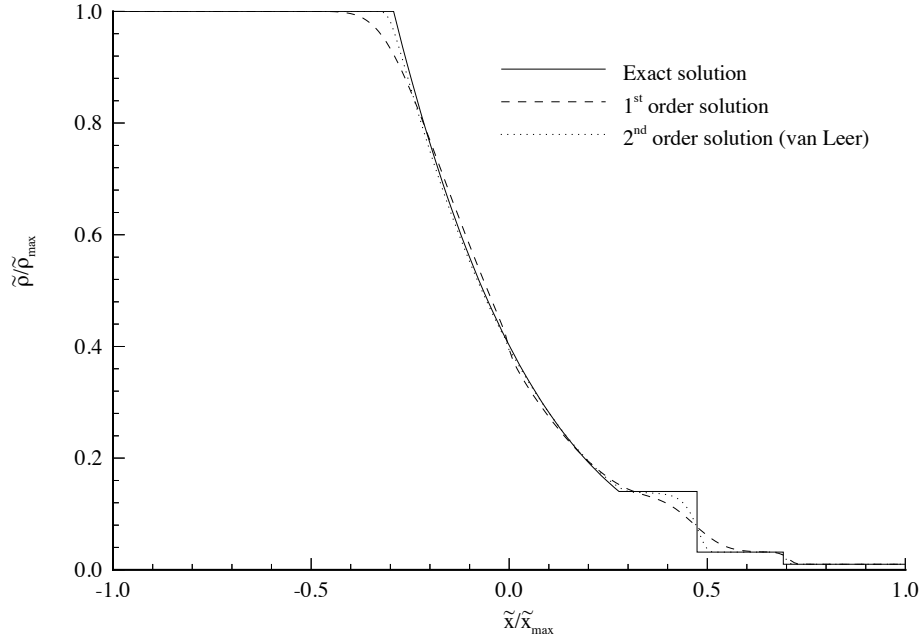


Figure 6.4: Comparison of Roe scheme solutions with the exact density solution of the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 100$, $\tilde{\rho}_l/\tilde{\rho}_r = 100$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0039$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).

Since an exact Riemann solution was known, a measurement was made of the actual spatial order of accuracy of the scheme. The error of the numerical solution is proportional to the truncation error of the scheme

$$e_1 \propto (\Delta\xi)^n, \quad (6.3)$$

where e_1 is the fractional error defined in Eq. (4.49), and $n = 2$ for a second-order scheme. The actual value of n can be determined by considering the log of this equation

$$\log e_1 \propto n \log \Delta\xi, \quad (6.4)$$

which indicates that the slope of a line on a plot of $\log e_1$ vs $\log \Delta\xi$ is the actual order of accuracy of the scheme. Figure 6.7 shows the results obtained for the (nominally) first- and second-order Roe schemes as $\Delta\xi$ was made increasingly smaller. In this case $\Delta\xi$ is proportional to the inverse of the number of grid points ($1/N$), since the

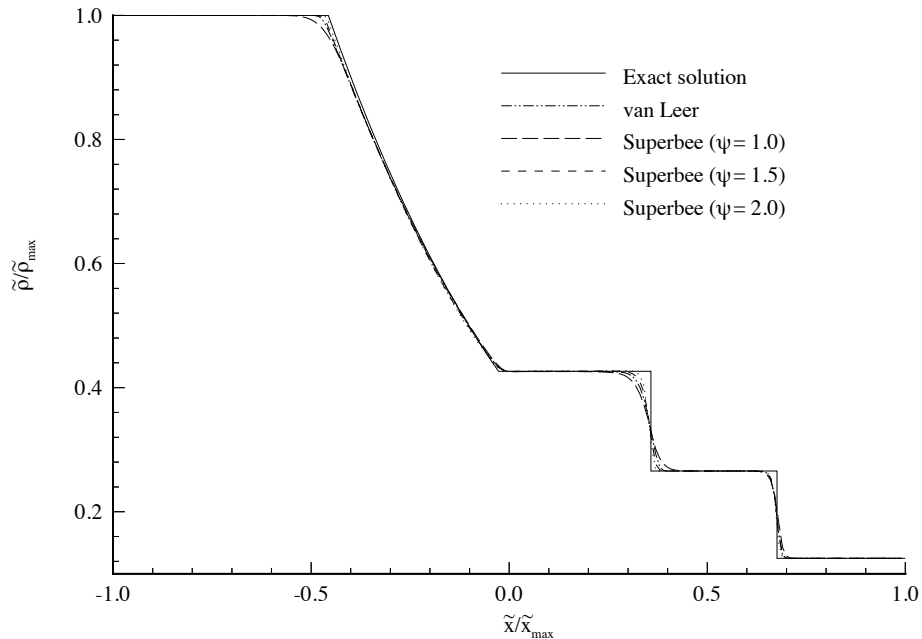


Figure 6.5: Comparison of 2nd order Roe scheme solutions using different limiters with the exact density solution for the Riemann problem ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m).

grid spacing is uniform. The slope of the least-squares fitted lines through the data are indicated on the graph. Obviously, neither the first nor the second-order solution lives up to expectations. The much lower than expected accuracy of the second-order scheme is likely the result of limiting the scheme to the first-order algorithm near discontinuities. The error at the discontinuity then overwhelms the error in the remainder of the solution. This type of behavior was noted by Woodward and Colella [86] when they considered a number of different schemes applied to a one-dimensional, non-linear problem. In fact, the value of n for their best second-order (Godunov-based) scheme did not pass 1 either.

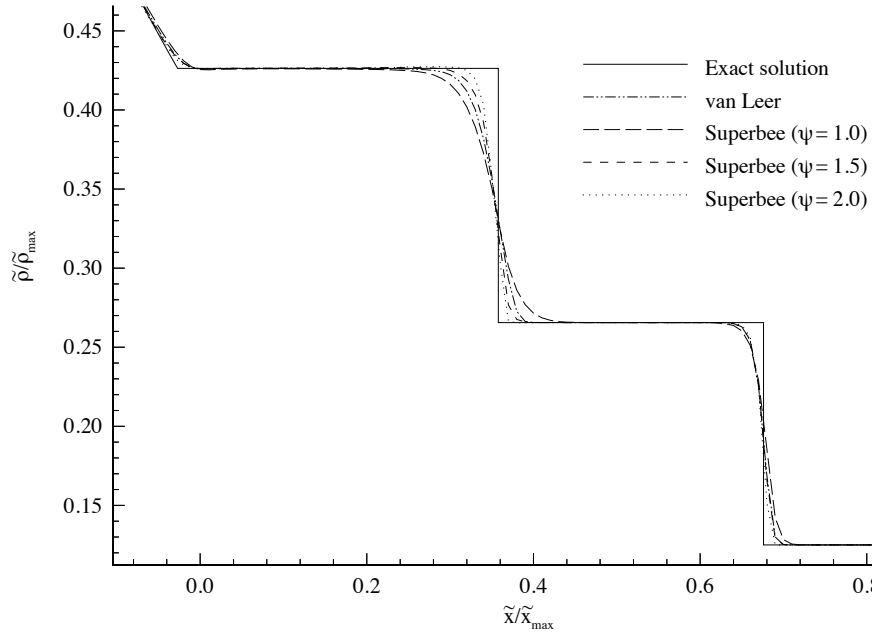


Figure 6.6: Enlarged view of the contact and shock discontinuities in the comparison of the 2nd order Roe scheme solutions. ($\tilde{p}_l/\tilde{p}_r = 10$, $\tilde{\rho}_l/\tilde{\rho}_r = 8$, $\tilde{u}_l = \tilde{u}_r = 0$ m/s, $\gamma = 1.4$, $\tilde{t} = 0.0061$ s, $\tilde{x}_{diaphragm} = 0$ m, $\tilde{x}_{max} = 5$ m)

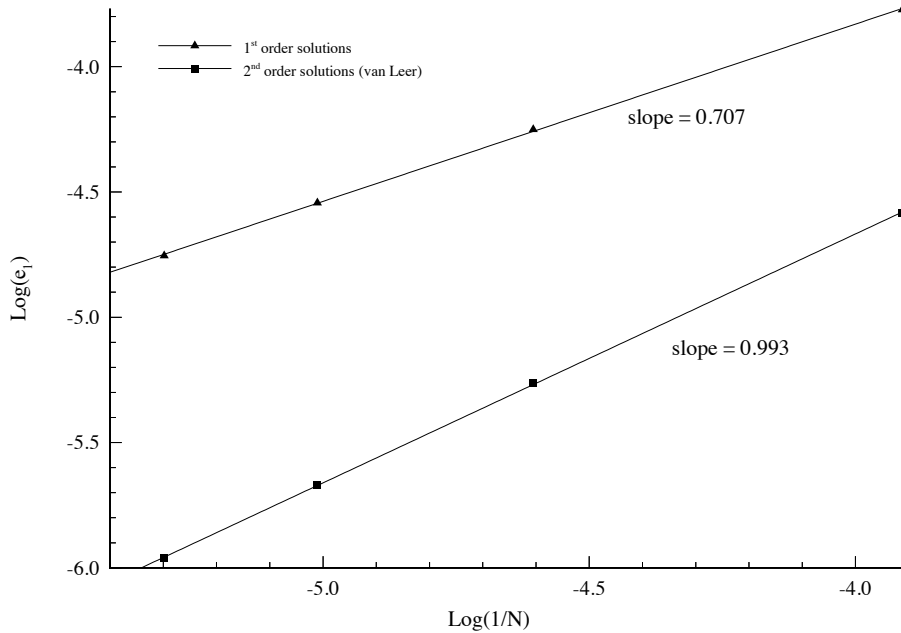


Figure 6.7: Measured order of accuracy for the first- and second-order Roe schemes using the exact Riemann solution as a basis.

6.2 One-dimensional Detonations

As previously discussed, one-dimensional detonations predicted by the model of Ch. 3 are known to be unstable for certain values of heat release and activation energy. The linear stability boundaries for these detonations are well known [47], and a number of researchers have numerically determined the unsteady solutions [28, 9]. Bourlioux, *et al.* [9] performed a detailed study of such unsteady detonations, and showed that they provide a stringent test of a numerical algorithm’s capability to accurately determine unsteady phenomena in reacting systems. In particular, because the linear stability characteristics of these detonations are known, it is straightforward to determine whether a given numerical solution is correctly predicting steady or unsteady behavior.

The second-order algorithm was tested by starting with the exact “steady” ZND solution of Ch. 4. An equally spaced one-dimensional grid was constructed, and the desired exact solution written on the left end of the grid; the remaining cell centers were initialized to the quiescent conditions ahead of the detonation. The calculations were then started, and the detonation would travel from the left side of the grid towards the right side. During the calculation the leftmost boundary fluxes were held constant at the end state conditions of the exact steady solution. The rightmost boundary fluxes were set to the quiescent values; the calculation was always terminated before the detonation reached this boundary, so it had no effect upon the solution.

6.2.1 Unsteady Solutions

The linear stability results indicate that for a nondimensional activation energy $\bar{E}_a = \tilde{E}_a \tilde{\rho}_0 / \tilde{p}_0 = 50$, nondimensional heat release $\bar{q} = 50$, and $\gamma = 1.2$, ZND detonations with an overdrive ratio $f > 1.73$ are stable, while those with $f < 1.73$ are unstable. Bourlioux, *et al.* reported detailed results for $f = 1.6$ and 1.8 using a higher-order

Godunov method with a front tracking, adaptive mesh algorithm. Their predictions of unsteady peak pressure for $f = 1.6$ compare favorably with those obtained using the second-order Roe scheme (see Fig. 6.8). The peak pressure is simply the maximum

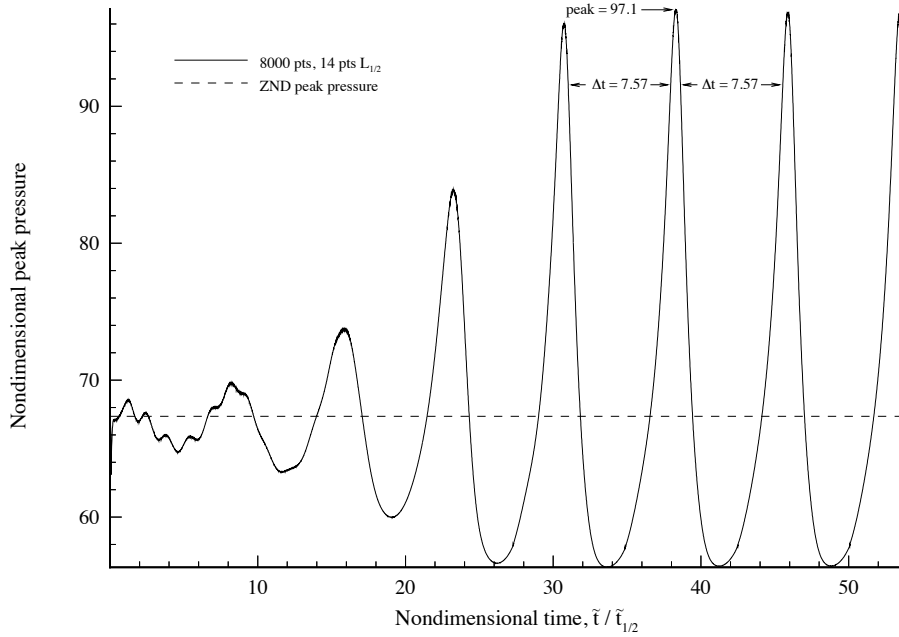


Figure 6.8: Unsteady detonation peak pressures ($f = 1.6$, CFL = 0.4, 14 points $\tilde{L}_{1/2}$).

pressure of the detonation at a given instant, and is a good indicator of the time-dependent nature of the detonation. The solution exhibits regular oscillations in the peak pressure, and thus the detonation is indeed unsteady. For reference, the peak pressure of the exact ZND solution is also shown; the unsteady detonation exhibits peaks more than 50% higher than the steady solution. The time scales were chosen for direct comparison with [9]; $\tilde{t}_{1/2}$ is the dimensional time required for a fluid particle to cross the dimensional half reaction zone length ($\tilde{L}_{1/2}$). The half reaction zone length is defined as the distance between the detonation front and the point at which the reaction is halfway to completion. An important criteria in these studies is the number of computational points within the half reaction zone length; the more points

there are, the more resolved the reaction zone structure is. In this case, there were 14 points per $\tilde{L}_{1/2}$, and 8000 total computational points.

For this case Bourlioux, *et al.* obtained a nearly identical plot with maximum peak pressures in the range of 100.6-102.7, and periods of oscillation in the range 7.4-7.5. As indicated in the figure, the peak values are nearly 100, and the periods are very close at 7.57. Figure 6.9 shows the results of doubling the number of grid points.

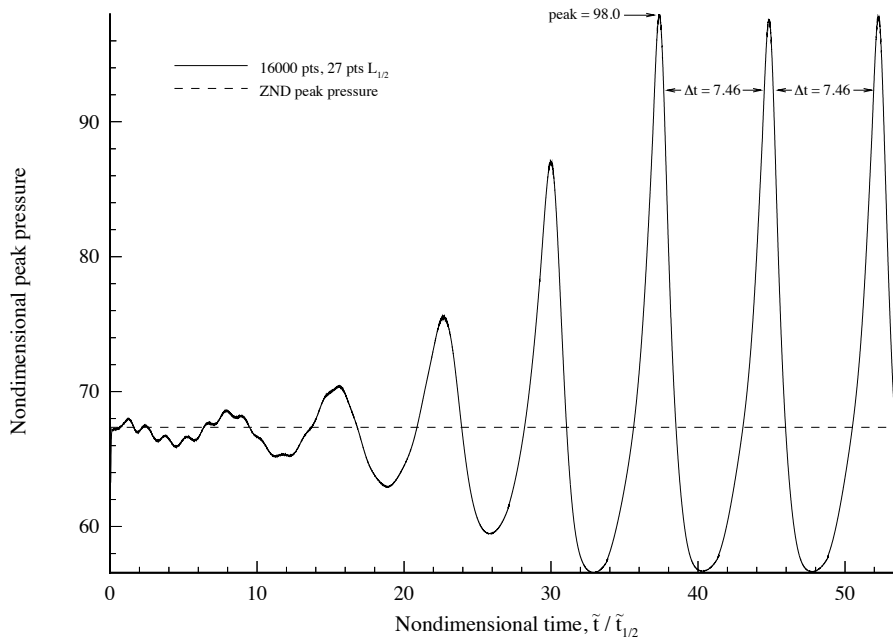


Figure 6.9: Unsteady detonation peak pressures ($f = 1.6$, $CFL = 0.4$, 27 points $\tilde{L}_{1/2}$).

Overall, the plot appears nearly identical to Fig. 6.8, except that the peak has moved slightly higher to 98.0, and the period has decreased slightly to 7.46. Thus, even better agreement with Bourlioux, *et al.* was obtained. Fourteen and 27 points per $\tilde{L}_{1/2}$ is not unreasonable; in comparison, Bourlioux, *et al.* used 20 points per $\tilde{L}_{1/2}$.

Figures 6.8 and 6.9 do exhibit one apparent artifact of the scheme. This is particularly noticeable in the transition region of each plot ($\tilde{t}/\tilde{t}_{1/2} < 20$), where portions of the peak pressure curve appear somewhat thicker. A closer inspection revealed these

to be slight changes in the peak pressure as the detonation front moved from cell to cell during the simulation. Reducing the CFL number lessened this effect somewhat, but did not seem to eliminate it entirely (see Fig. 6.10). As will be shown in the

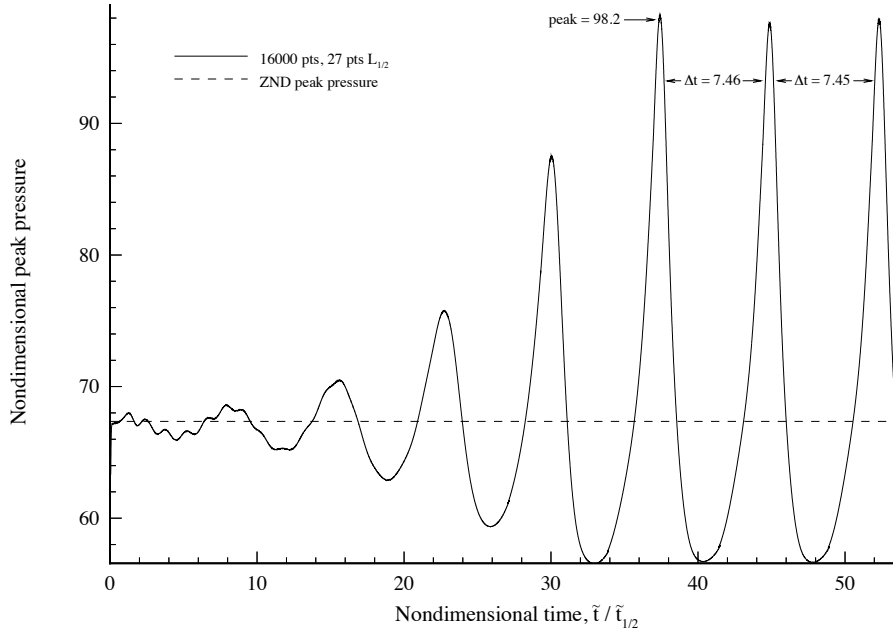


Figure 6.10: Unsteady detonation peak pressures ($f = 1.6$, $CFL = 0.35$, 27 points $\tilde{L}_{1/2}$).

following discussion, however, this artifact did not seem to have any effect on the schemes ability to correctly predict stable and unstable detonation solutions.

In order to more fully evaluate the scheme, a case closer to the stability boundary was studied. Figures 6.11 and 6.12 show the results for $f = 1.72$, which is slightly within the unstable regime. For 5 points per $\tilde{L}_{1/2}$ the solution oscillates regularly with a very slightly decreasing amplitude as time progresses. Doubling the number of points results in the solution oscillating with increasing magnitude as time progresses. Thus, the correct unstable behavior is predicted even very close to the stability boundary.

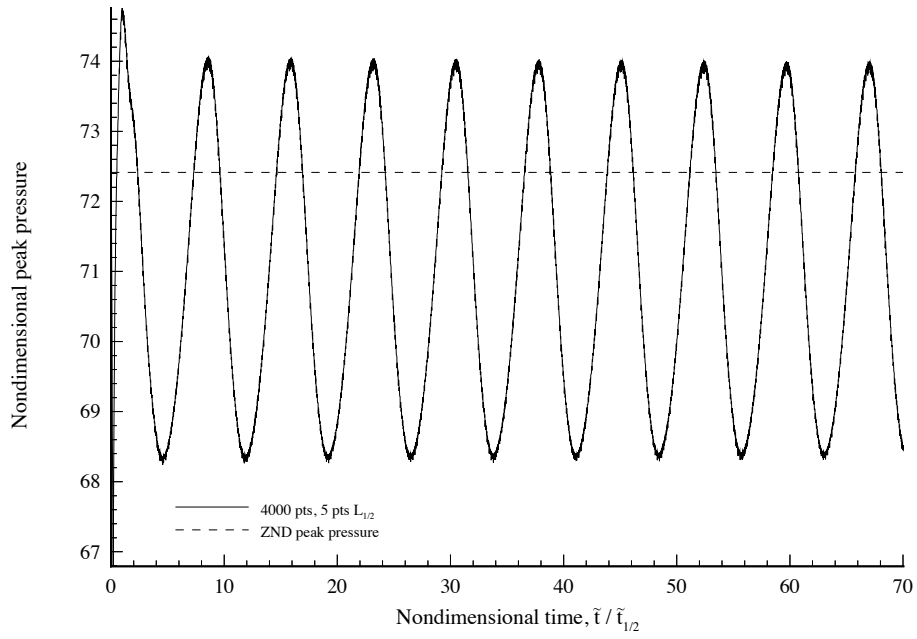


Figure 6.11: Unsteady detonation peak pressures ($f = 1.72$, CFL = 0.4, 5 points $\tilde{L}_{1/2}$).

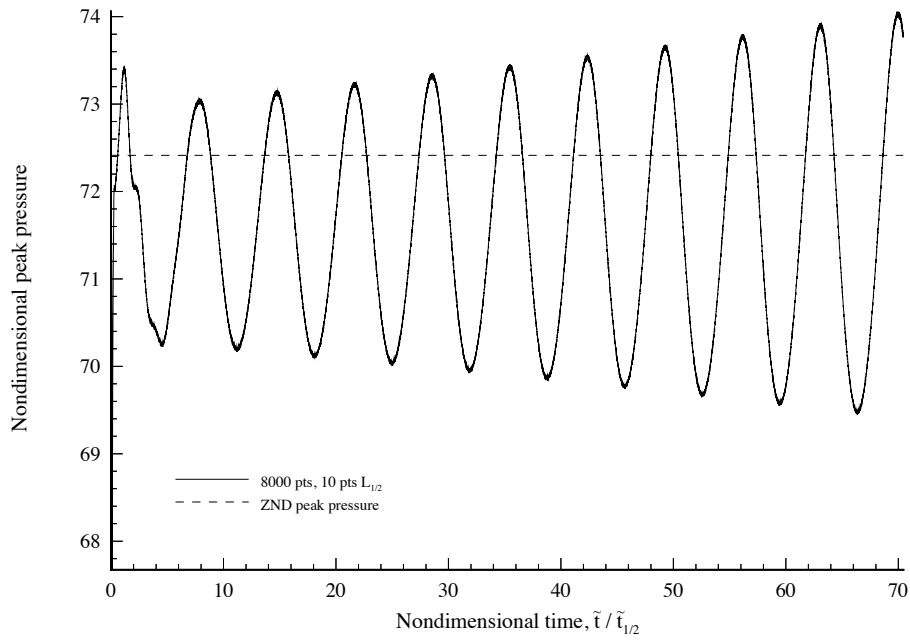


Figure 6.12: Unsteady detonation peak pressures ($f = 1.72$, CFL = 0.4, 10 points $\tilde{L}_{1/2}$).

6.2.2 Steady Solutions

Figure 6.13 shows a solution obtained in the steady regime ($f = 1.8$) using 9 and 18 points per $\tilde{L}_{1/2}$. As expected, the initial transients are damped, and the solutions

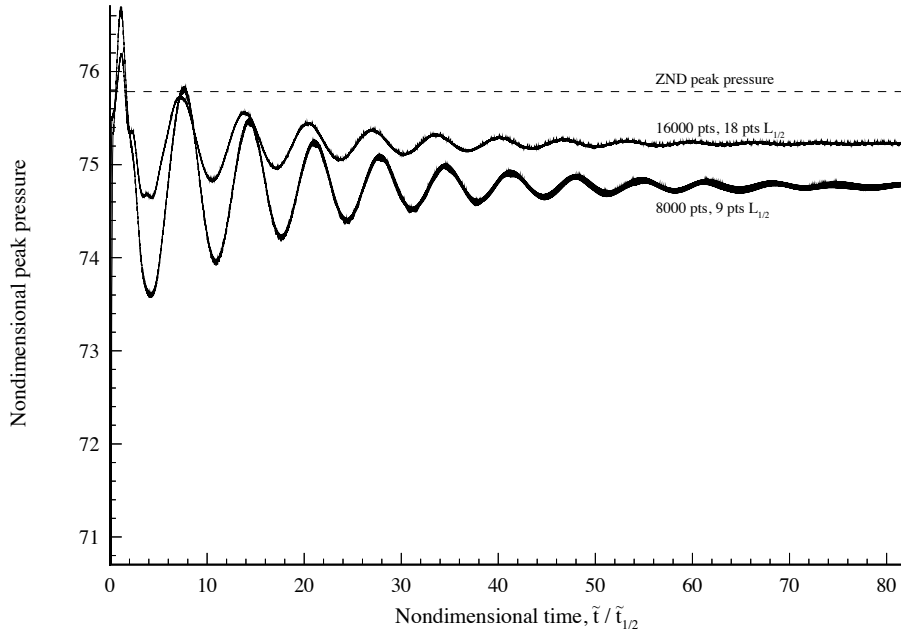


Figure 6.13: Steady detonation peak pressures ($f = 1.8$, CFL = 0.4).

asymptotically approach a steady peak pressure. For linearly stable cases the steady peak pressure should equal that of the exact ZND solution; increasing the number of points in the half reaction zone length resulted in the solutions more closely approaching this value. As before, a case closer to the stability boundary was considered. Figure 6.14 shows the results obtained slightly within the stable regime ($f = 1.74$). In this case using 5 and 20 points per $\tilde{L}_{1/2}$ led to solutions that are slowly approaching a steady value close to the expected value.

Finally, as was done using the exact Riemann solution, the order of accuracy of the scheme was determined using the exact ZND solution at $f = 1.8$ (see Fig. 6.15). In this case only the portions of the solutions near the wave front were compared due to the initial transients in the numerical solution. As before, the indicated order of

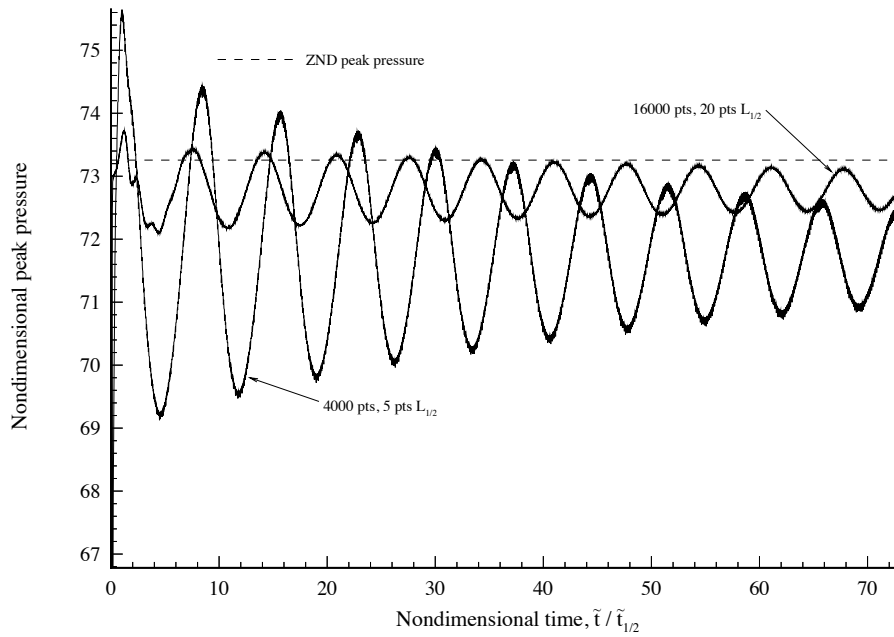


Figure 6.14: Steady detonation peak pressures ($f = 1.74$, $CFL = 0.4$).

accuracy is around 1; in this case the fit to the points is not nearly as good as in Fig. 6.7. The poor fit is likely a result of the ambiguity in “lining up” the final numerical solution with the exact solution for the comparison.

Overall, when an adequate number of points within the half reaction zone length is used, the second-order Roe scheme is capable of obtaining the correct steady and unsteady features of a one-dimensional reactive flow.

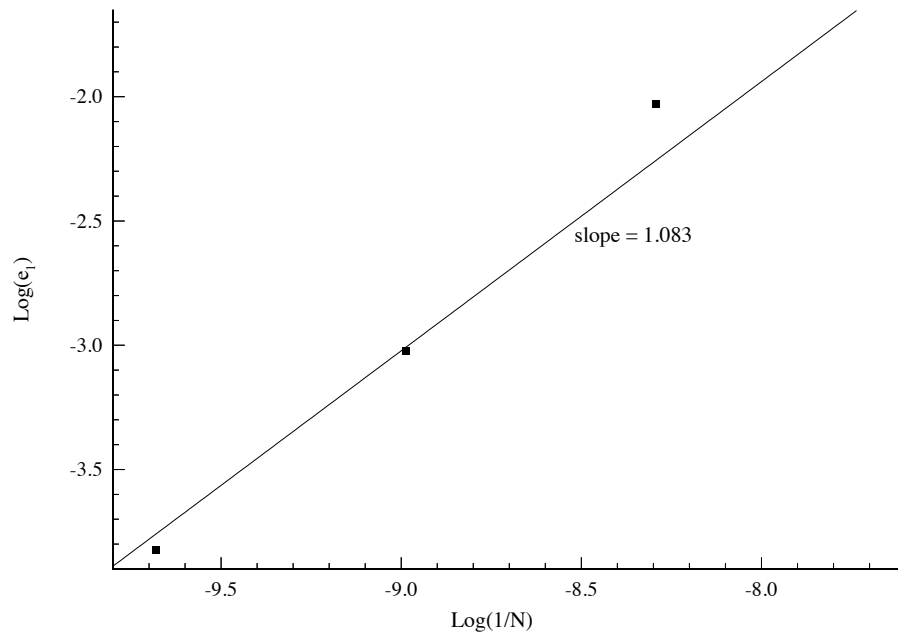


Figure 6.15: Measured order of accuracy for the second-order Roe scheme using the exact ZND solution as a basis ($f = 1.8$).

6.3 Inert Supersonic Flow over a Wedge

The numerical method was tested in two dimensions by considering inert supersonic flow over a wedge. This flow is characterized by oblique shock waves whose strength and angle can be determined exactly using a Rankine-Hugoniot analysis. A program was written to generate smooth, somewhat orthogonal, clustered computational grids over arbitrary shapes. Figure 6.16 indicates the computational mesh generated for this problem: the wedge angle is $\theta = 20^\circ$; there are 49 cell centers in each direction; and the mesh has been clustered somewhat near the wedge surface. Following common

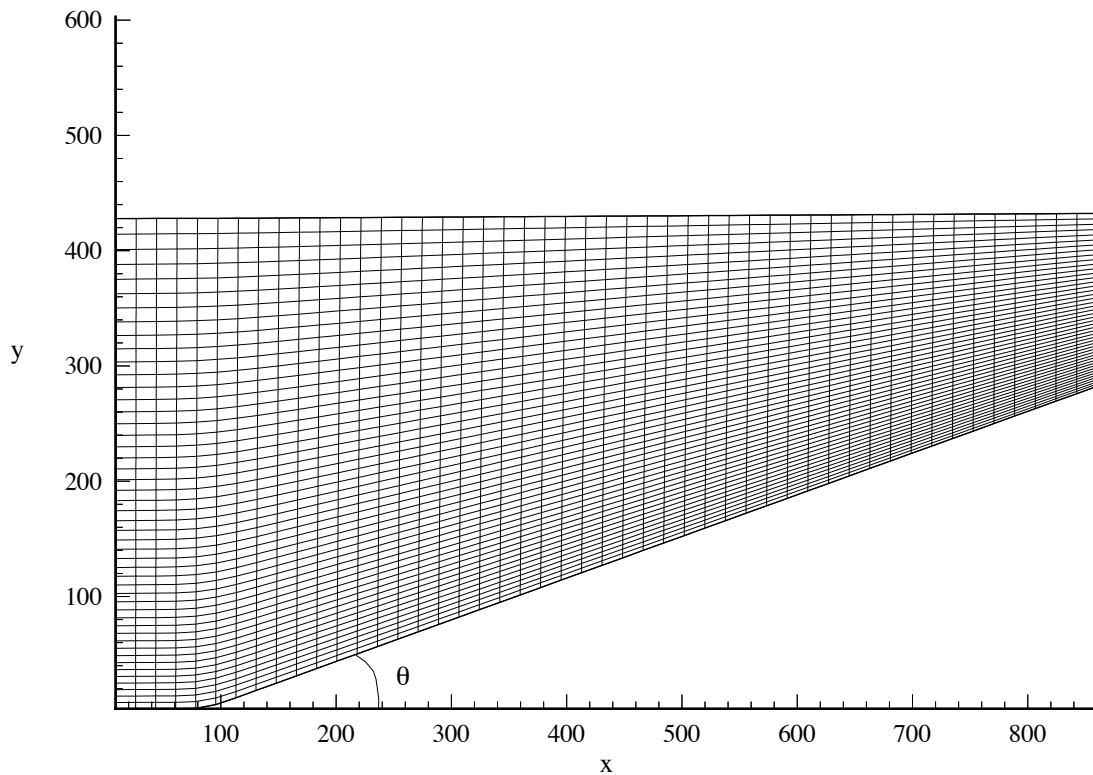


Figure 6.16: Computational grid of cell centers used for 20° wedge.

convention, the grid points in the x direction are numbered $i = 1, 2, \dots, 49$, while those in the y direction are numbered $j = 1, 2, \dots, 49$. This grid was used for all of the following cases with $CFL = 0.4$.

Figure 6.17 indicates the pressure and velocity vector solution obtained for a freestream Mach number of $\mathcal{M}_0 = 10$. In this case the shock exits through the rear of the domain, so the outflow condition at the top of the domain does not affect the solution. As expected, the plots indicate a sharp pressure and velocity discontinuity at the shock. The measured shock angle is $25^\circ \pm 1^\circ$, which compares well with the exact value of 25.8° . The effect of the rear outflow condition can be seen in the slight tilting of the pressure contours at the exit of the domain. A few extraneous contours in the region after the shock indicate the pressure is not quite constant at the post-shock pressure. The tangency of the velocity vectors at the wall indicates the slip condition is being enforced by the wall boundary condition. Figure 6.18 shows a comparison between the numerical and exact density solutions along various $j = \text{constant}$ grid lines. In this and the following cases, the exact solution was written on the same computation grid as the numerical solution. This is a more realistic portrayal of the actual “best” solution that can be obtained for a given grid. Along each gridline the numerical solution captures the shock in only 4 to 5 points, which is not much worse than the scheme in one dimension. The worst agreement appears in the $j = 1$ and $j = 3$ gridlines, where $j = 1$ is the row of cell centers nearest the wall. Along $j = 1$ the solution initially overshoots the exact solution, and then follows it more closely farther downstream. Along $j = 3$ the solution undershoots the exact solution, and then remains below it downstream. The solutions farther from the wall do not seem to have this problem, though they do drop below the exact solution downstream of the shock. The overshoot is likely a result of the discontinuity in the grid at the wedge tip. The solutions at each grid line also exhibit slight oscillations, indicating that the extension to two dimensions has not entirely preserved the TVD nature of the one-dimensional scheme. Figure 6.19 is the same type of plot for the v component of the velocity. In this case the agreement is even better; the v component shows no over- or undershoots, but does oscillate slightly. Though not shown, the behavior of

the u component solution is nearly identical to that of the v component. Lastly, Fig. 6.20 compares the pressure solutions. In this case there is a slight overshoot along the $j = 1$ gridline, but otherwise the agreement between the numerical and exact solutions is very good.

To get an idea of how well this scheme converged to a steady solution, Fig. 6.21 shows the density and u velocity component residuals as the solution progressed. The residuals are defined as follows

$$\Upsilon_{res} = \frac{\sum_{i=1}^{i_{max}} \sum_{j=1}^{j_{max}} |\Upsilon_{i,j}^{n+2} - \Upsilon_{i,j}^n|}{N}, \quad (6.5)$$

where Υ represents the variable of interest, and N is the total number of cells in the grid. Thus, the residual is basically the average difference between the variable at the old and new time step (the value at $n + 2$ is used due to the splitting method of Eqs. (5.117)). Ideally, the residuals would decrease to a machine zero of 10^{-15} for a steady problem. Realistically, this does not always happen. As the figure indicates, the residuals decreased about two orders of magnitude before leveling out. At this point the solution was no longer changing. Slightly different grids resulted in different levels of convergence, but the overall solution would change very little.

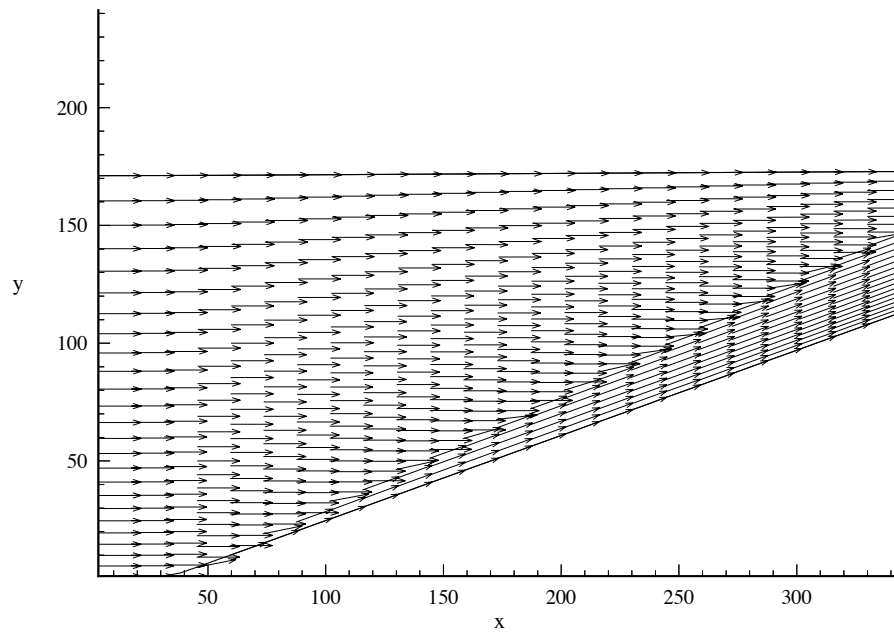
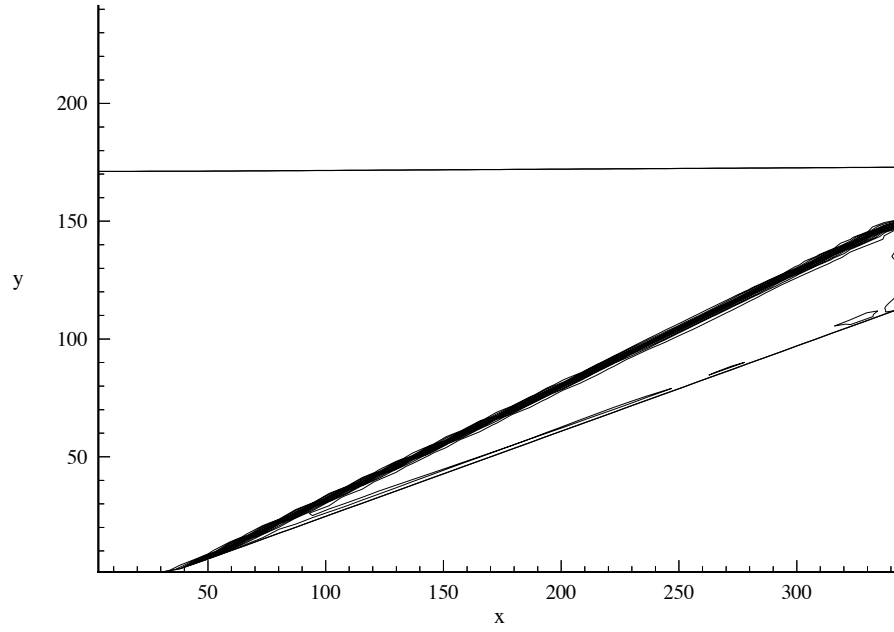


Figure 6.17: Pressure contours and velocity vectors for $\mathcal{M}_0 = 10$.

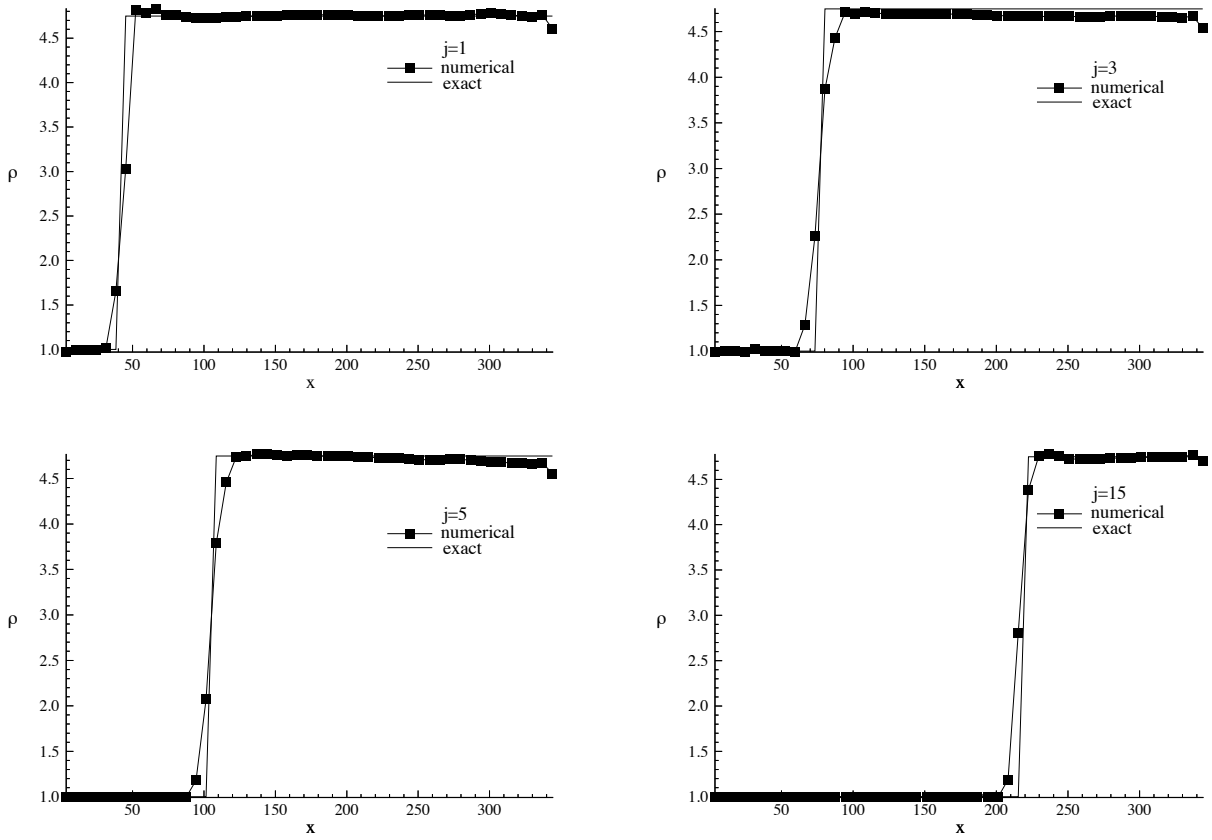


Figure 6.18: Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 10$.

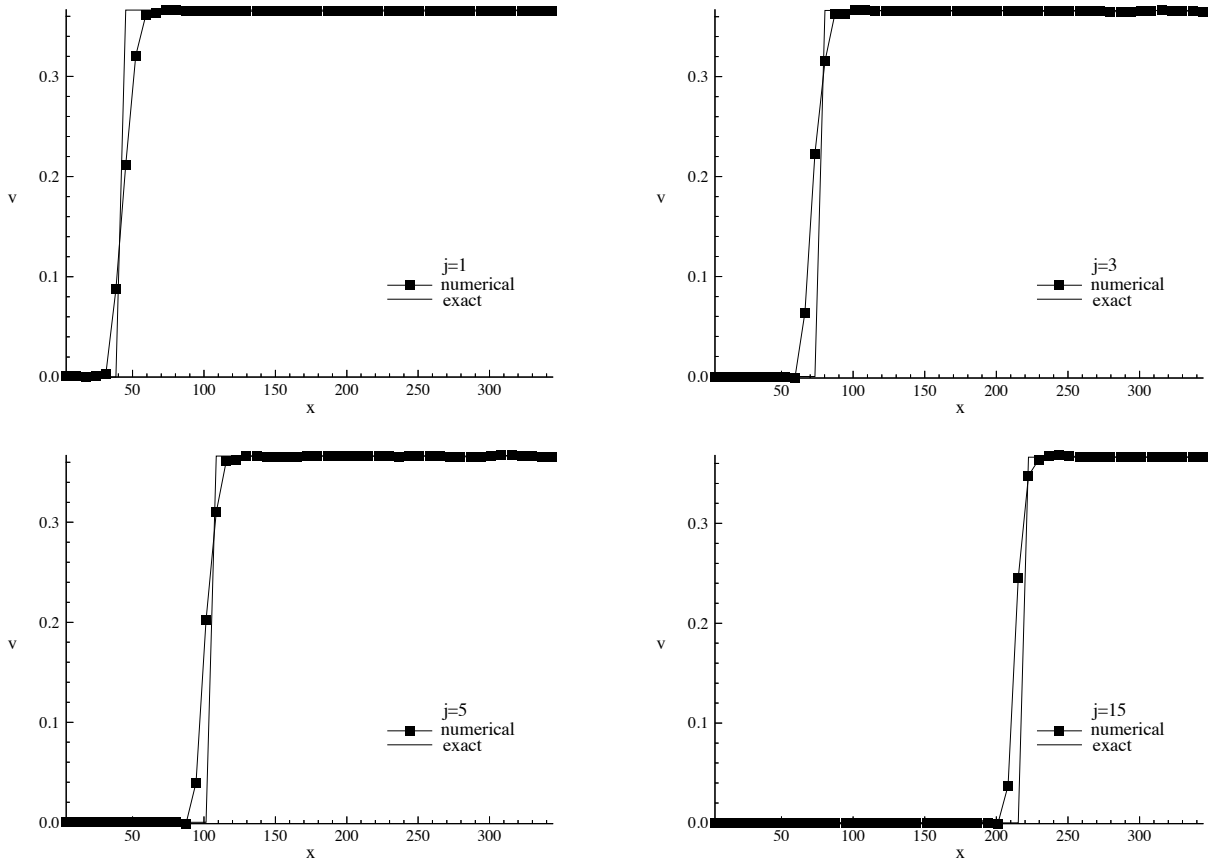


Figure 6.19: Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 10$.

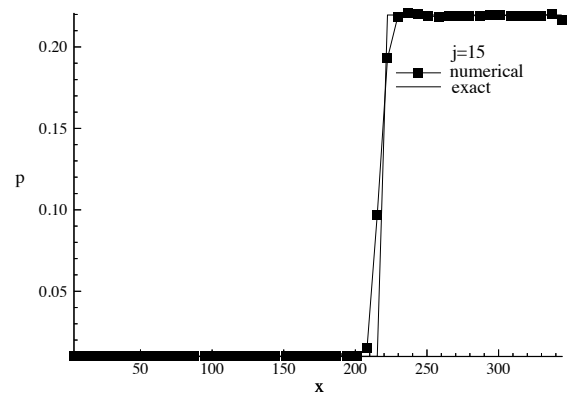
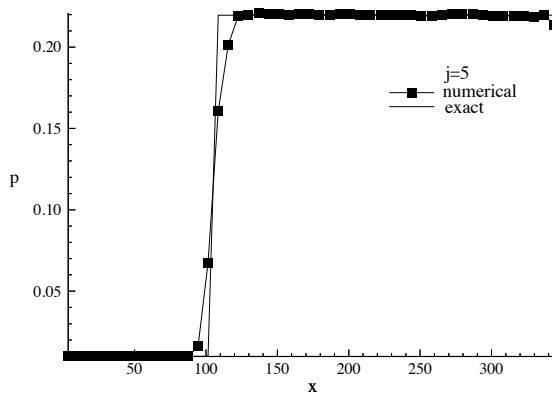
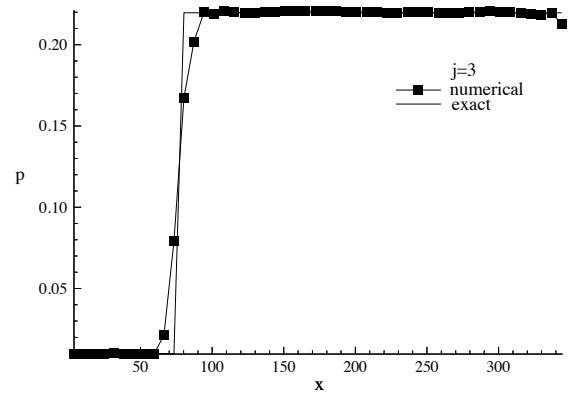
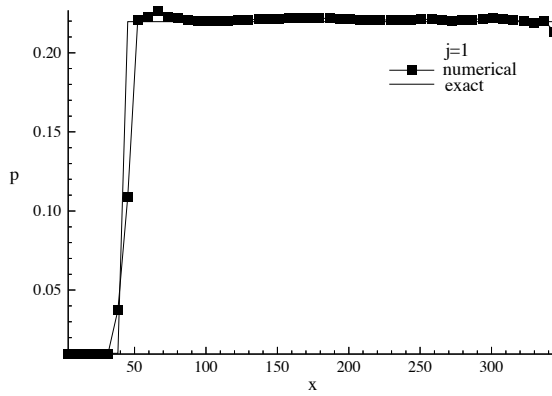


Figure 6.20: Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 10$.

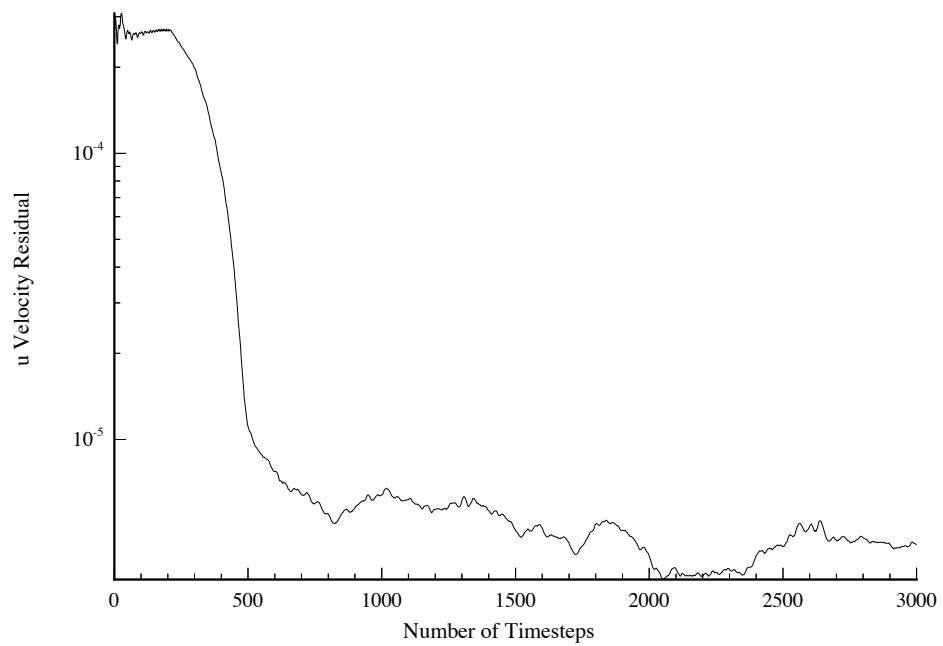
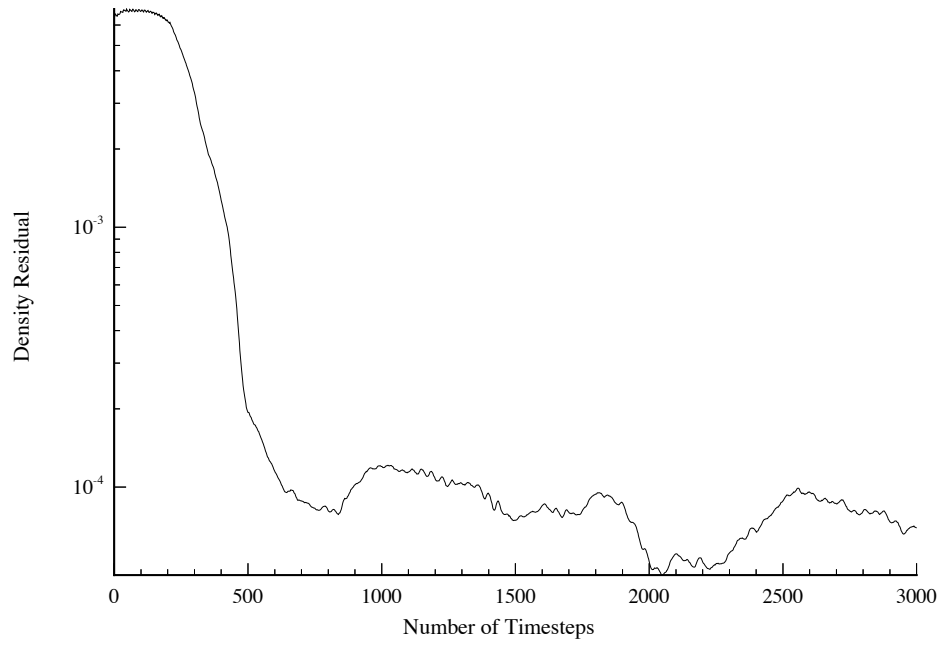


Figure 6.21: Residual histories of density and u component of velocity for $\mathcal{M}_0 = 10$.

In order to evaluate the subsonic outflow condition discussed in Ch. 5, the free-stream Mach number was lowered such that the shock would exit through the top of the computational domain ($\mathcal{M}_0 = 2$). In this case, one characteristic is entering the top of the domain from outside, and Eq. 5.126 is needed to estimate its magnitude. The value of the constant K in the equation will be determined by the solution that best matches the exact solution. Ideally, there should be no reflections from the top of the domain, and the flow state following the shock should be constant and equal to the exact solution. Figure 6.22 shows pressure and velocity solutions obtained for $K = 0.0$. This is Thompson's [76] nonreflective boundary condition; the magnitude of any incoming waves is set equal to zero. As is very apparent in both figures, this leads to large reflections from the boundary. In fact, a shock appears to be propagating upstream into the domain, which is obviously opposite to the desired behavior. The inadequacy of this value of K is not entirely unexpected, however, since it ignores the actual physical propagation of information into the domain from above

Figure 6.23 shows the same information when $K = 0.025$. In this case there is a different type of reflection. The pressure after some distance downstream of the shock is decreasing slowly to an intermediate state between the freestream and shocked pressure. The region between the reflection and the shock is at the correct shocked pressure, however, and the reflection is stationary. The velocity vectors at the boundary appear to have the correct magnitude and direction at the boundary, but it is difficult to determine from this figure. Figures 6.24, 6.25, 6.26, and 6.27, compare the solutions for ρ , p , u , and v with the exact solutions along $j = \text{constant}$ grid lines. In this case the row of cell centers along the top of the domain is shown ($j = 49$). Unlike the case of $\mathcal{M}_0 = 10$, the density nearest the wall is consistently less than the exact solution. Along interior grid lines the density follows the exact solution fairly well to a point, and then begins to decrease to an intermediate density. The point at which the decrease begins moves closer to the shock as the top of the

domain is approached. Along the topmost grid line the density slightly overshoots the exact value, and then immediately begins to decrease. For the pressure solution, the comparison is much better. There is a good region of agreement between the exact and numerical solutions for all but the last grid line. Along $j = 49$ the numerical solution is shocked to the correct value, and then immediately begins to decrease. The u component of velocity follows a pattern similar to the density. It is initially shocked to a value lower than the exact solution along $j = 1$ and for some distance downstream, and then begins to increase. Along interior grid lines the numerical and exact solutions have relatively good agreement for a given distance, and then the numerical solution begins to increase. Along the topmost gridline, the numerical solution is shocked to a value above the exact solution, and immediately begins to increase. Lastly, the v component of velocity is similar to ρ and u in that there are regions of good agreement along the interior $j = \text{constant}$ lines, with discrepancies along the wall and the topmost gridline.

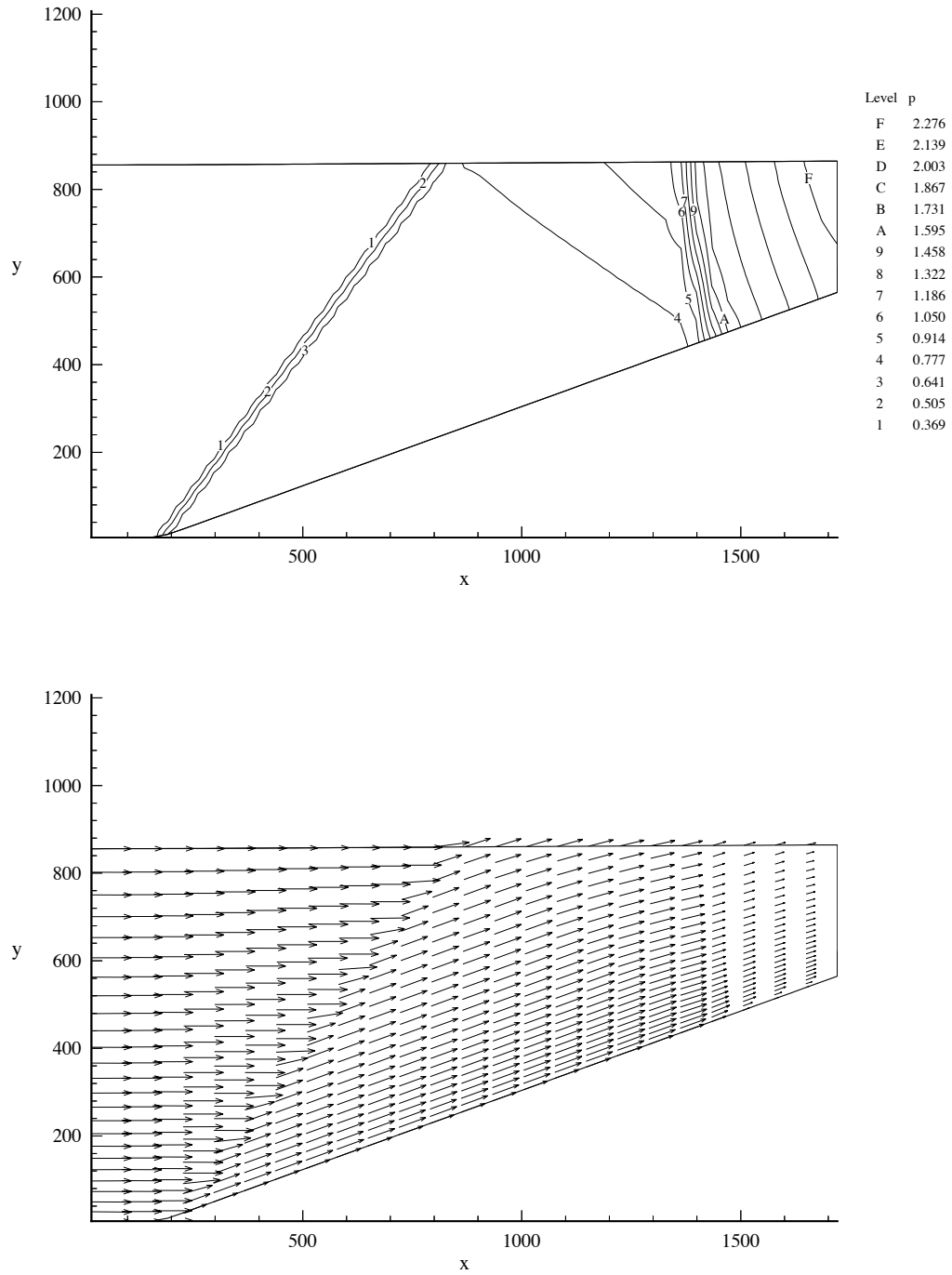


Figure 6.22: Pressure contours and velocity vectors for $\mathcal{M}_0 = 2$ and $K = 0.0$.

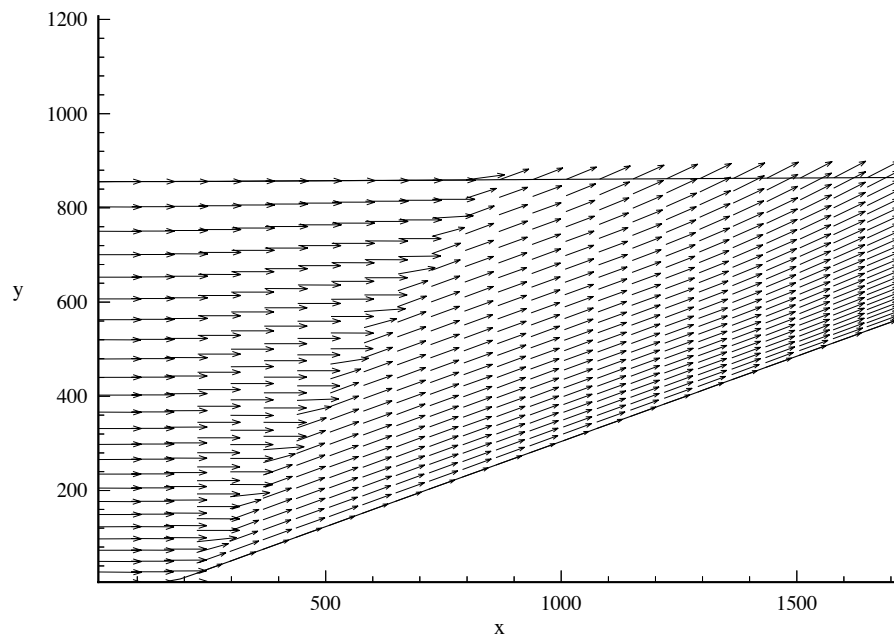
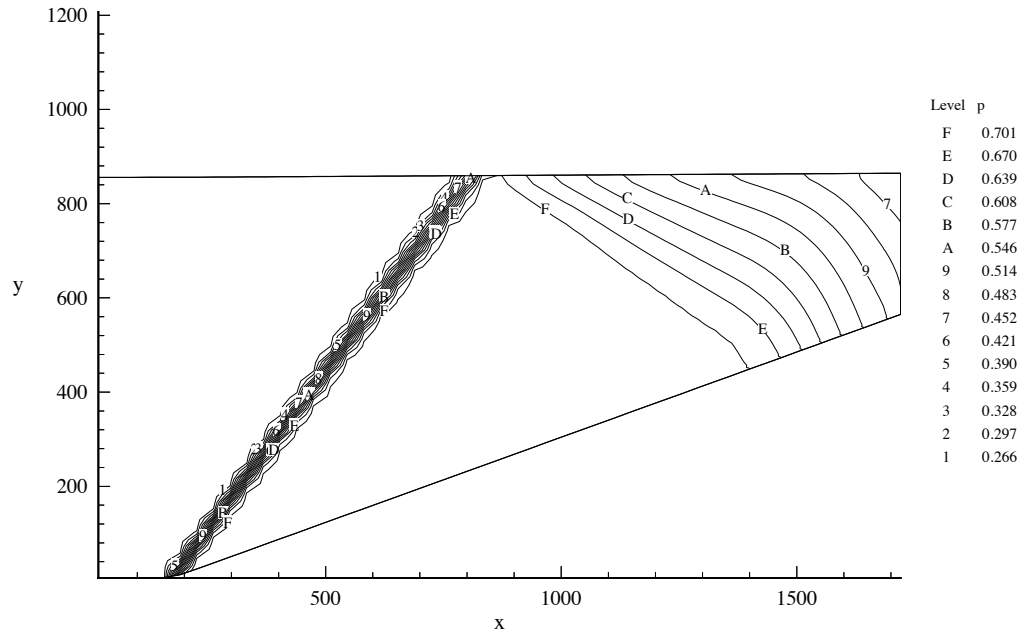


Figure 6.23: Pressure contours and velocity vectors for $\mathcal{M}_0 = 2$ and $K = 0.025$.

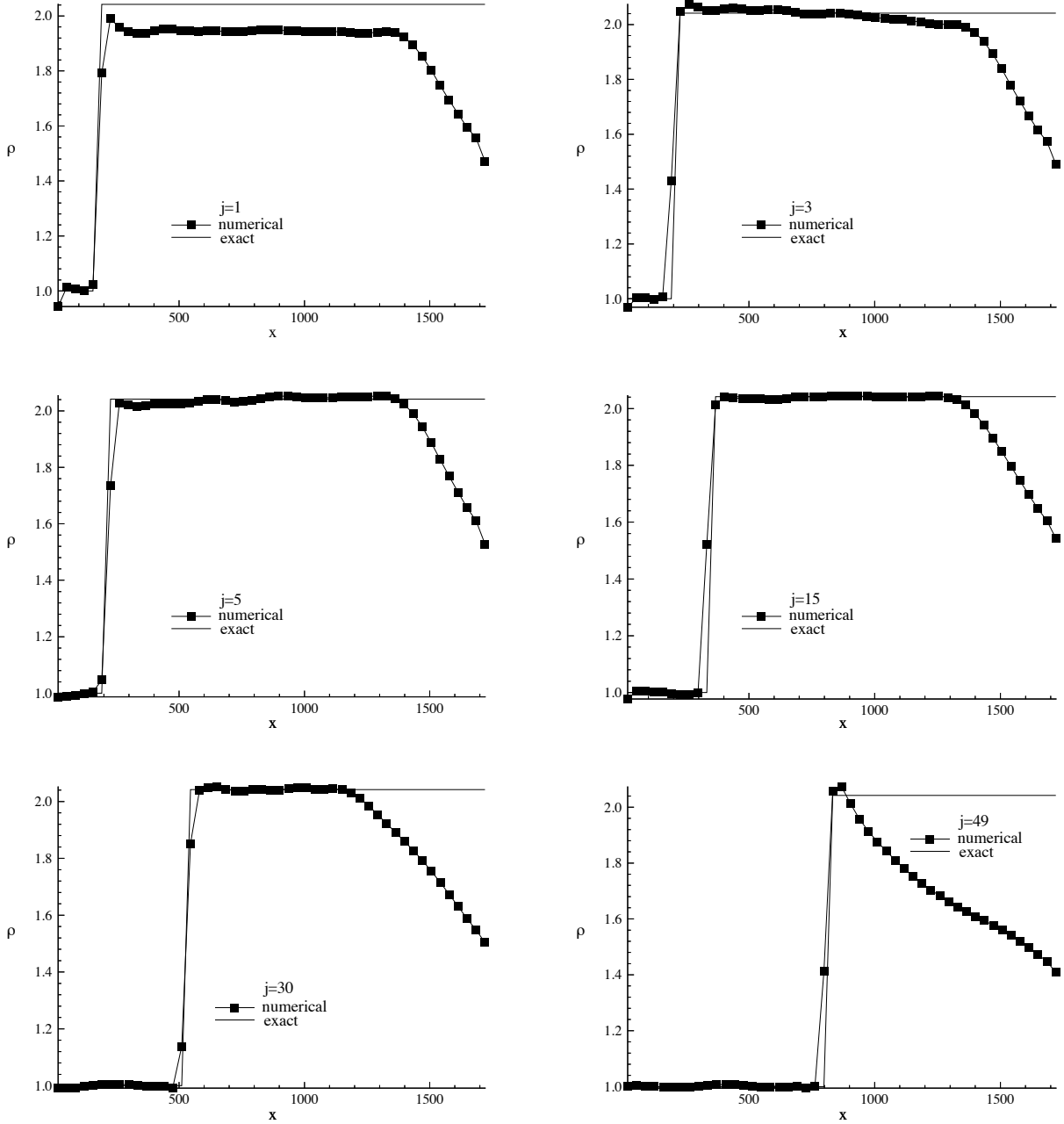


Figure 6.24: Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$.

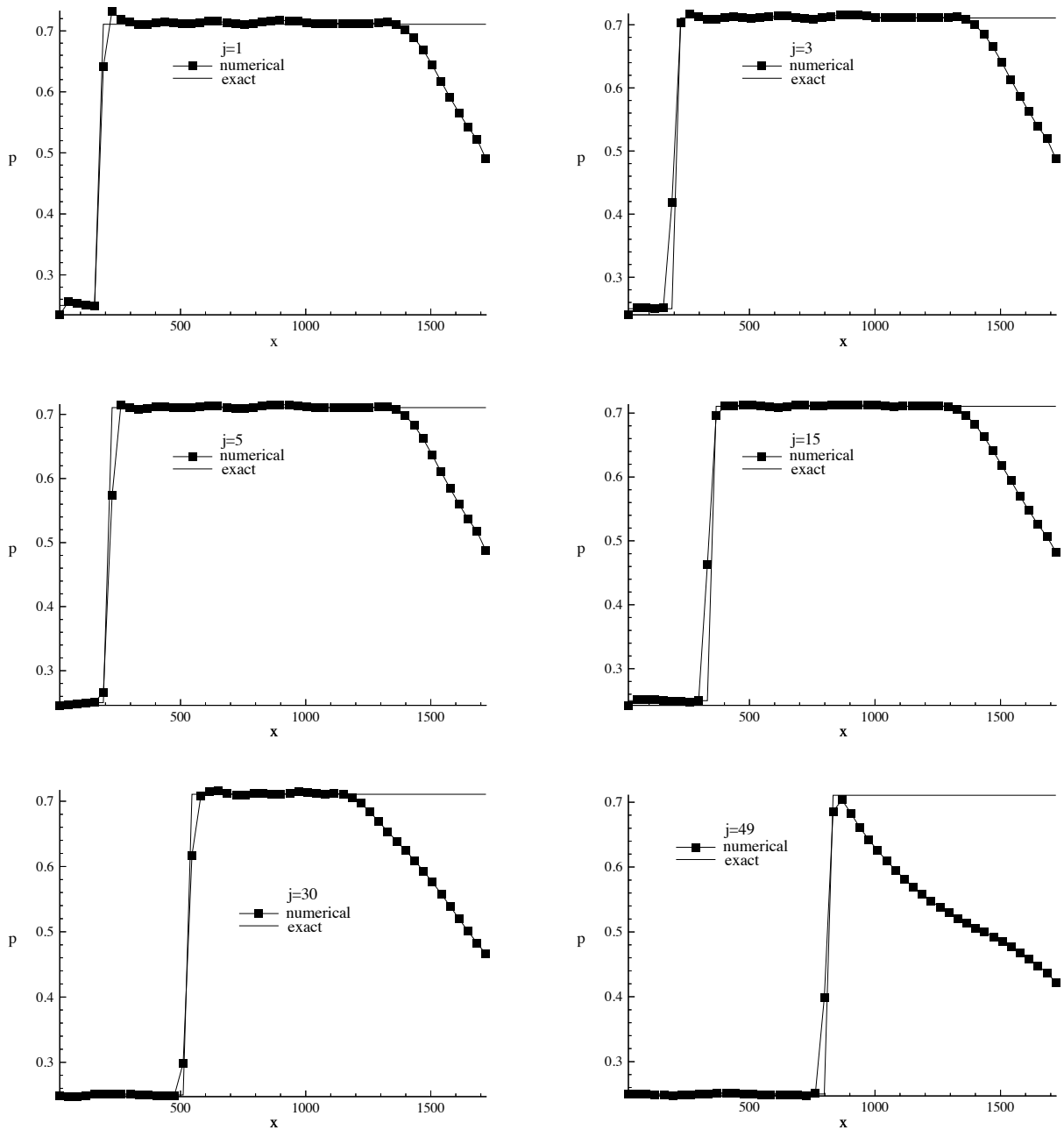


Figure 6.25: Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$.

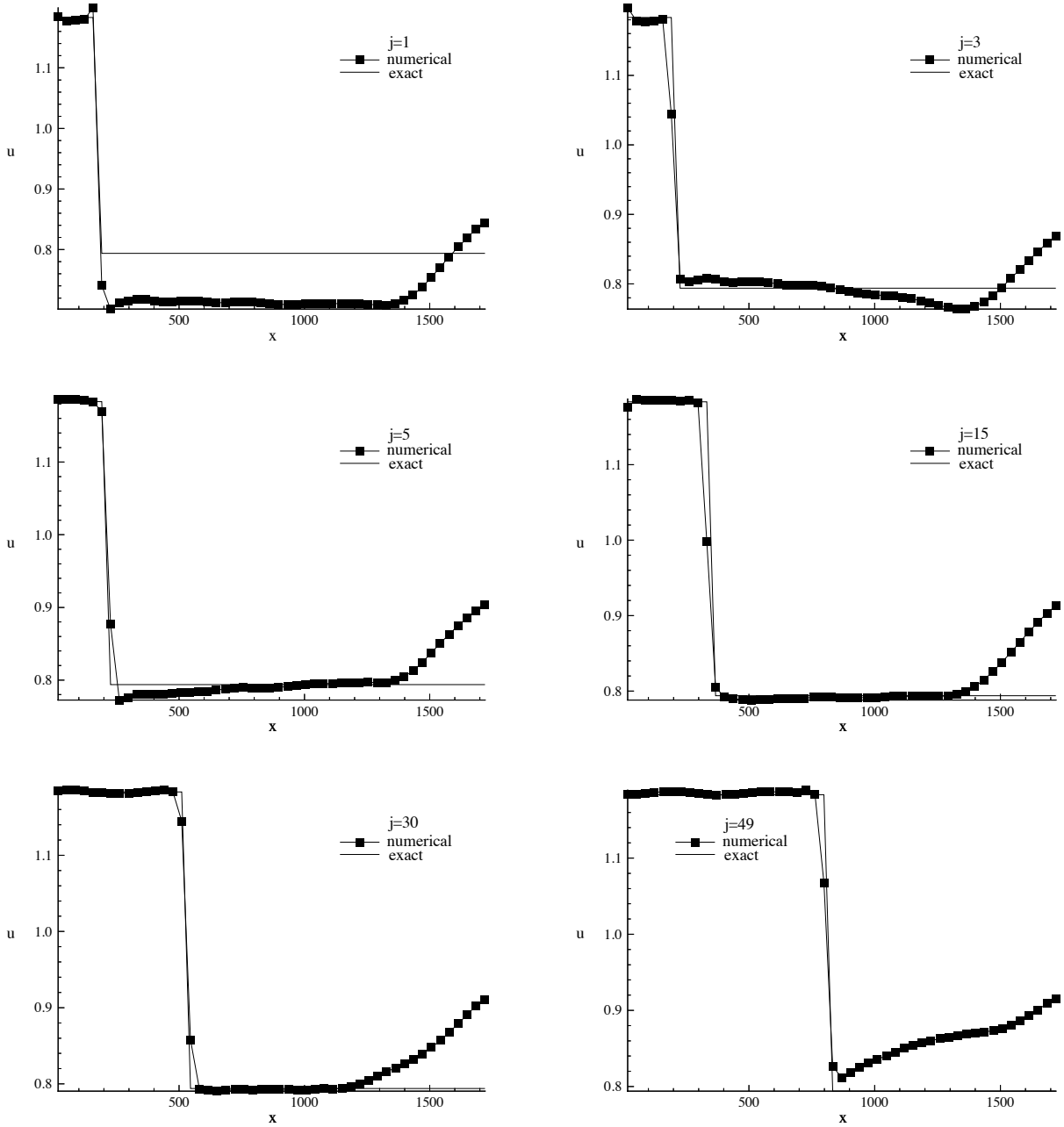


Figure 6.26: Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$.

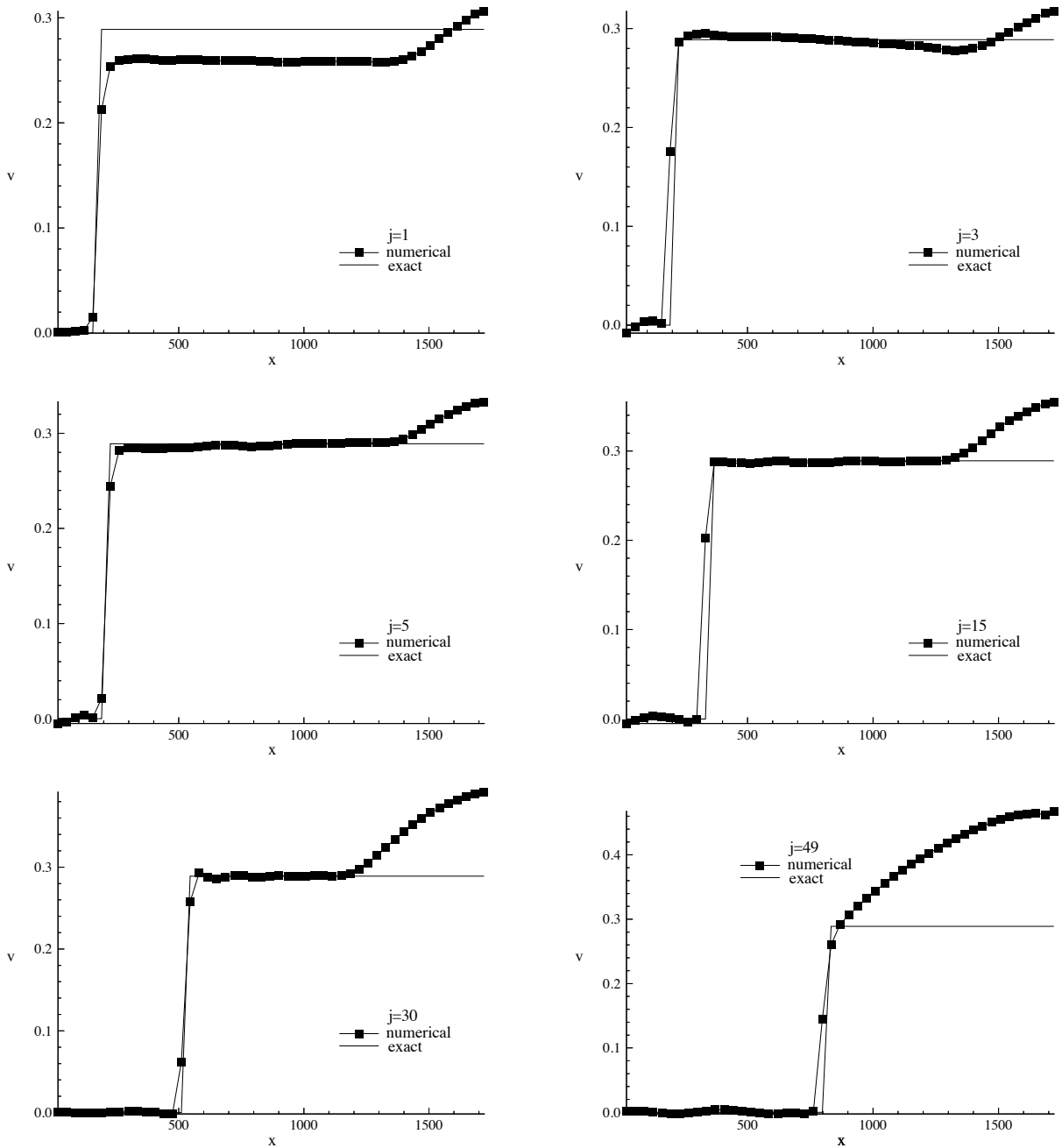


Figure 6.27: Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.025$.

Figures 6.28, 6.29, 6.30, 6.32, and 6.31 show the same types of data obtained for $K = 0.035$. The main change in the solutions from $K = 0.025$ to $K = 0.035$ is that the increase (or decrease, as the case may be) of the variables through the reflection region is larger. For K somewhat less than 0.025, the opposite behavior occurred. However, along the $j = 49$ grid line, the density and pressure solutions both begin to overshoot the exact values at the shock. For a reactive flow, these two variables drive the reaction through the right-hand side of Eq. 3.22. Overshoots in their shocked values at the boundary could result in an increased reaction rate, which would lead to the undesirable situation in which the boundary condition is forcing the reaction to proceed. Based upon this, $K = 0.025$ was chosen as the optimal value to use in the remaining numerical studies.

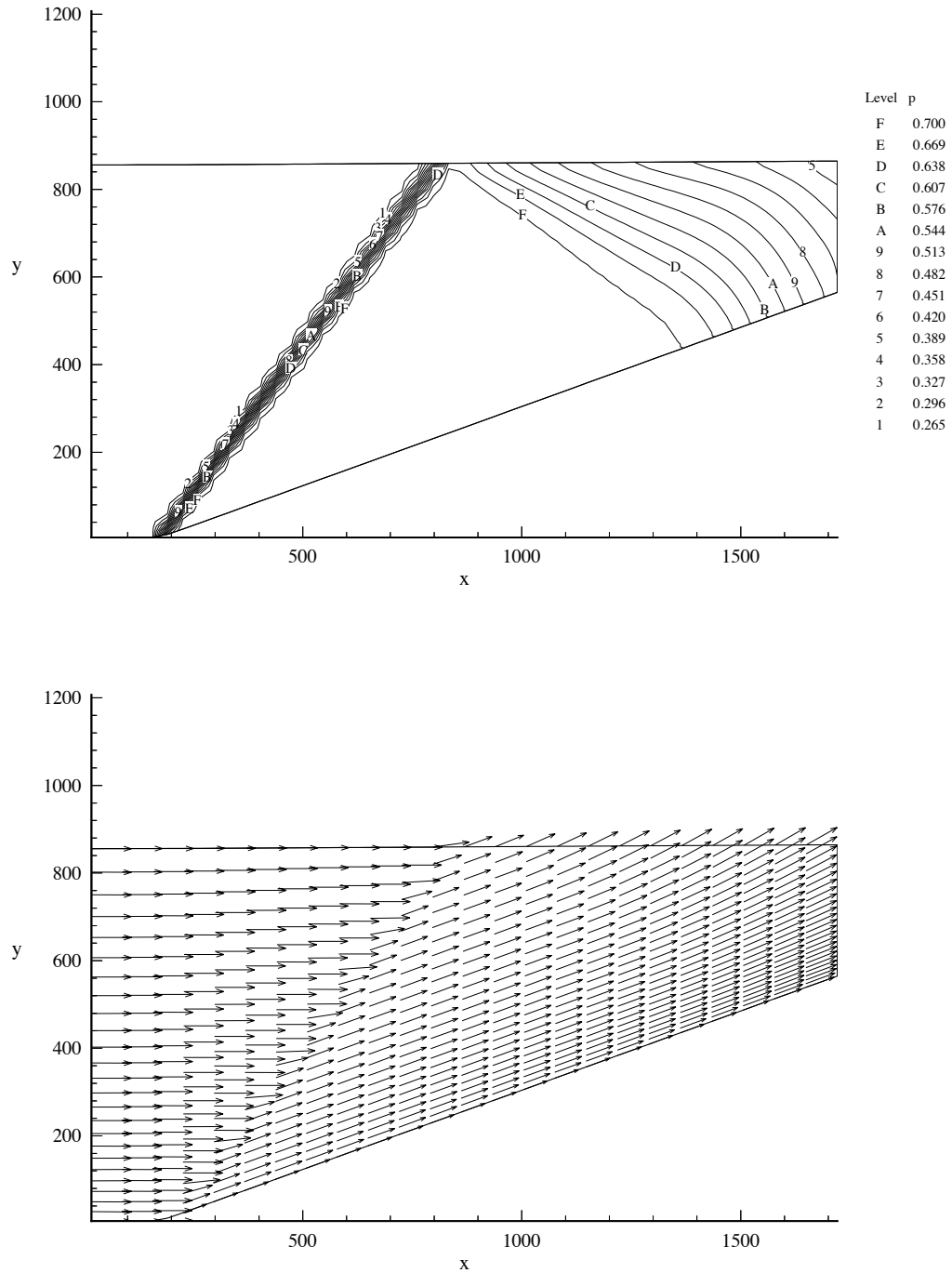


Figure 6.28: Pressure contours and velocity vectors for $\mathcal{M}_0 = 2$ and $K = 0.035$.

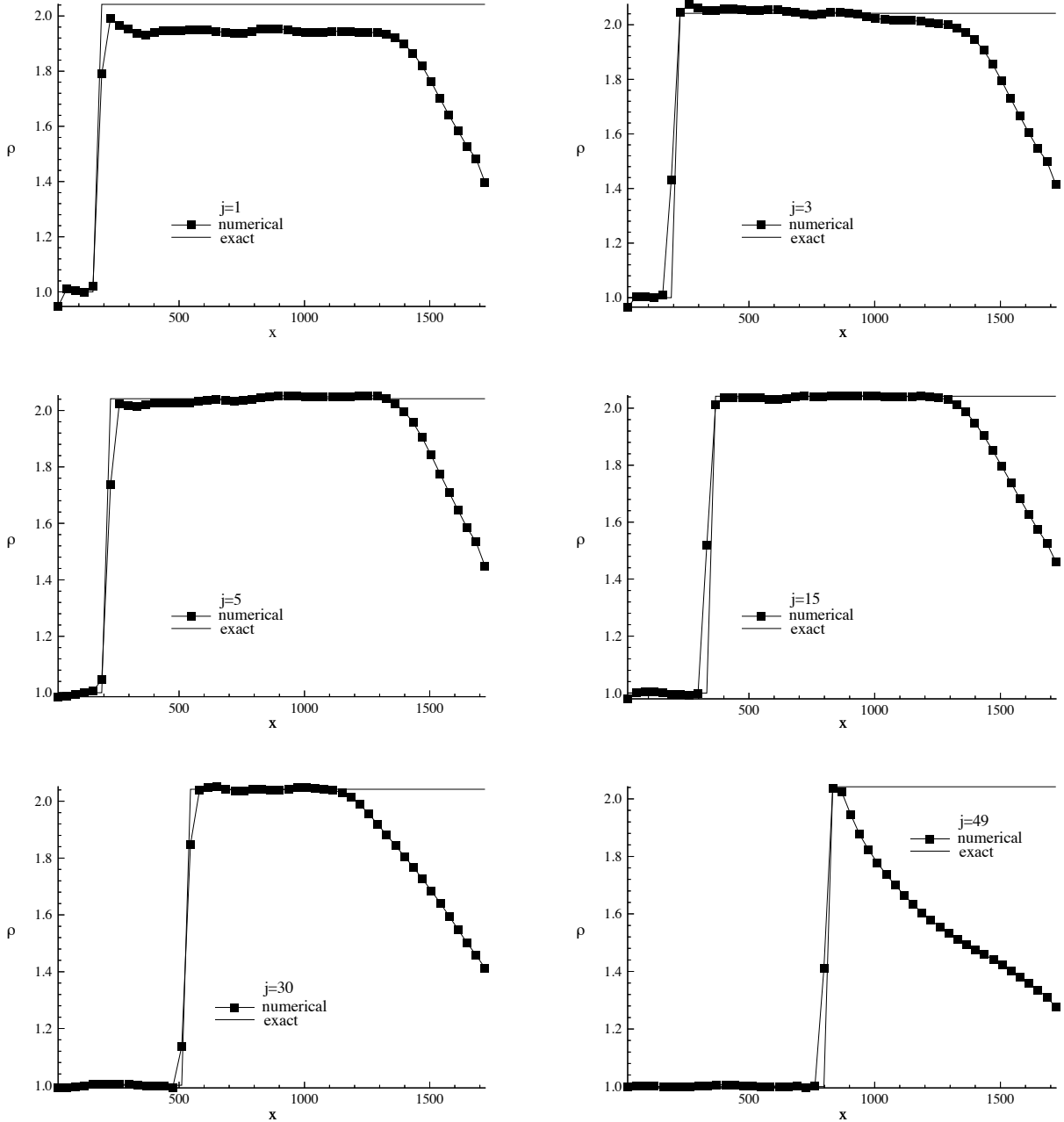


Figure 6.29: Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$.

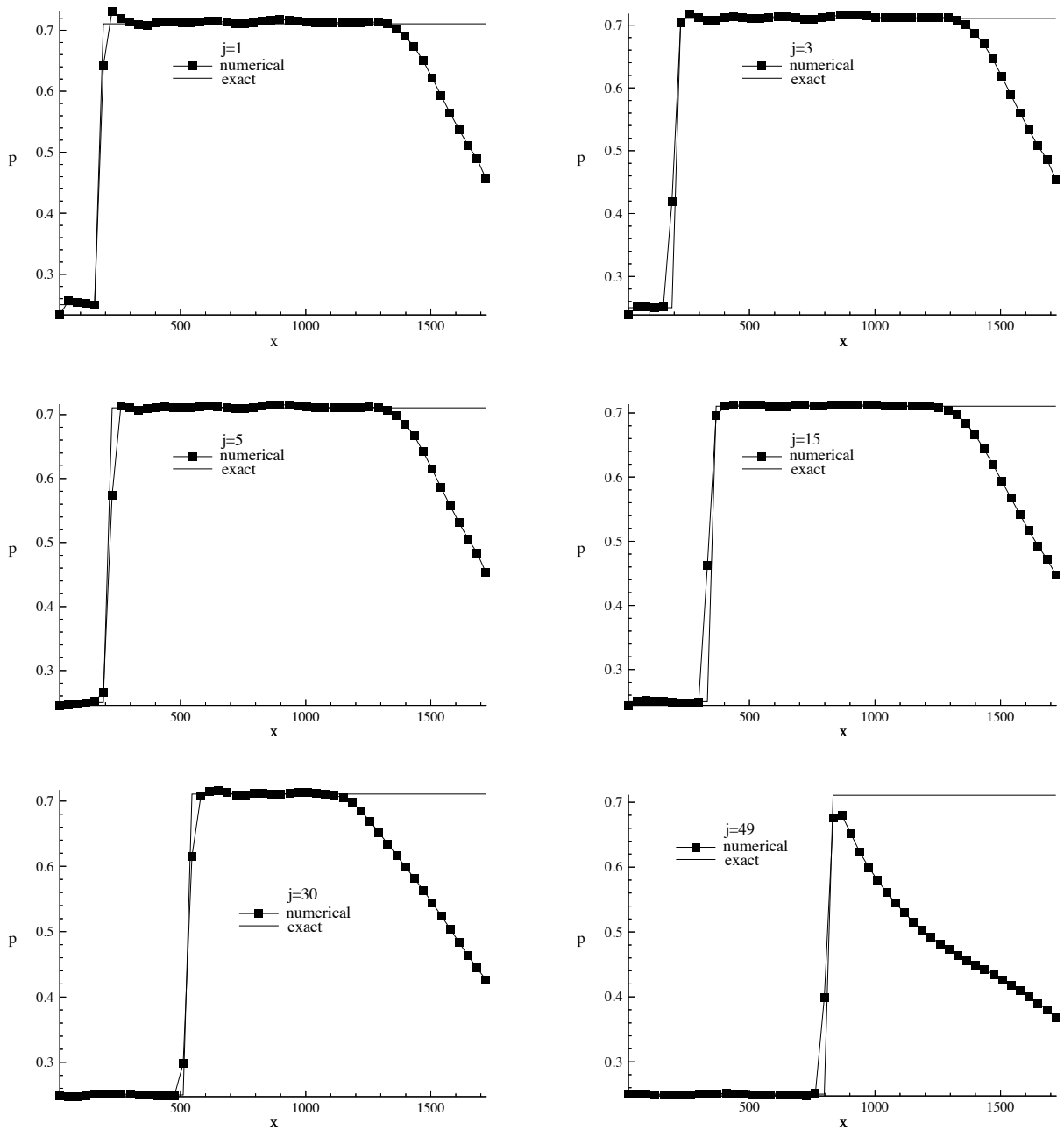


Figure 6.30: Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$.

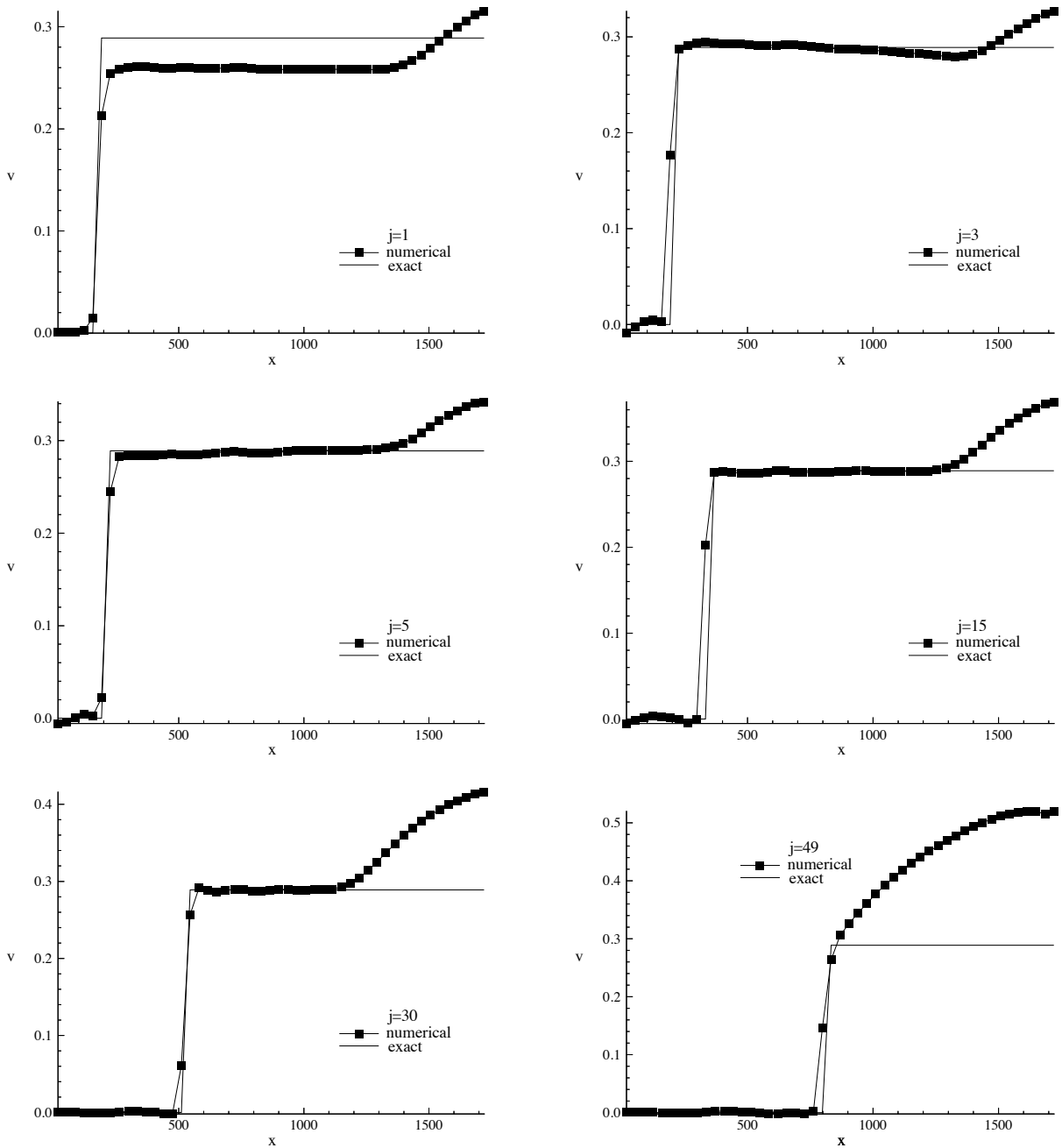


Figure 6.31: Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$.

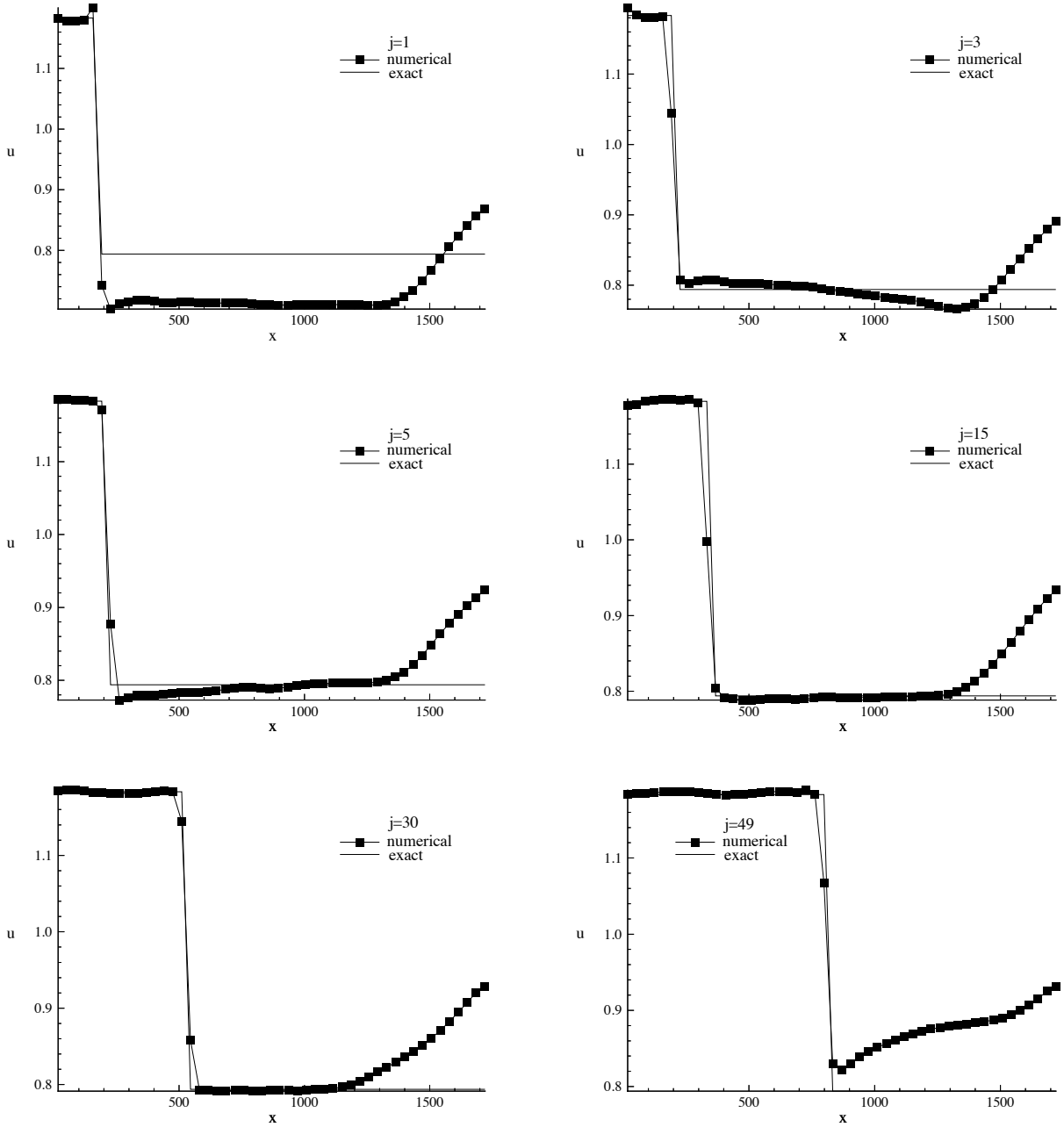


Figure 6.32: Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 2$ and $K = 0.035$.

The chosen value of K was further evaluated by considering a second case in which the shock exits through the top of the computational domain: $\mathcal{M}_0 = 4$. Figure 6.33 shows the pressure contours and velocity vectors obtained for this case. It is immediately apparent in the pressure contours that the reflection due to the boundary condition is much more localized than in the case of $\mathcal{M}_0 = 2$. As before, the velocity vectors would seem to indicate the correct outflow at the boundary. Figures 6.34, 6.35, 6.36, and 6.37 show the comparisons with the exact solution for ρ , p , u and v , respectively. In each case the agreement with the exact solution is very good for all of the $j = \text{constant}$ grid lines except the last. Along $j = 49$ the previously seen behavior of the numerical solutions for ρ , p , and v reappear; the numerical solution for u , however, maintains excellent agreement with the exact solution. Apparently, the higher velocity of the flow at $\mathcal{M}_0 = 4$ acts to reduce the range of influence of the outflow boundary condition.

As was done with the one-dimensional Riemann solutions, the average error over the entire grid of the two-dimensional oblique shock solutions was determined on a number of increasingly finer grids to ascertain the actual spatial order of accuracy of the scheme. The results are shown in Fig. 6.38 for $\mathcal{M}_0 = 4$ and $K = 0.025$. As before, the actual order of accuracy of the second-order scheme is slightly less than one. This indicates that extending the scheme to two dimensions via fractional stepping, as well as using the outflow boundary condition along the top of the domain, did not adversely effect the spatial accuracy of the scheme.

Overall, the extension of the Roe scheme to two dimensions via operator splitting resulted in essentially the correct inert shock solutions.

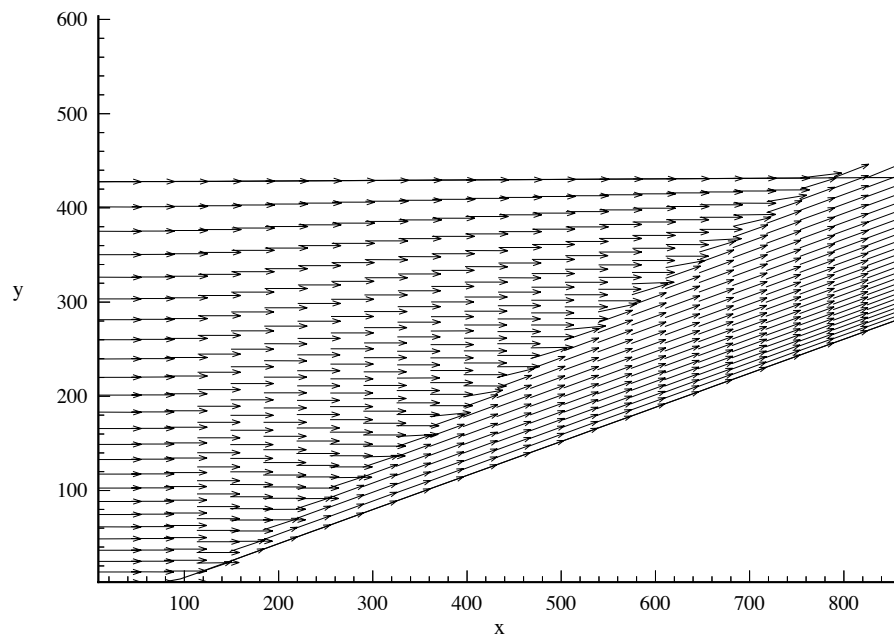
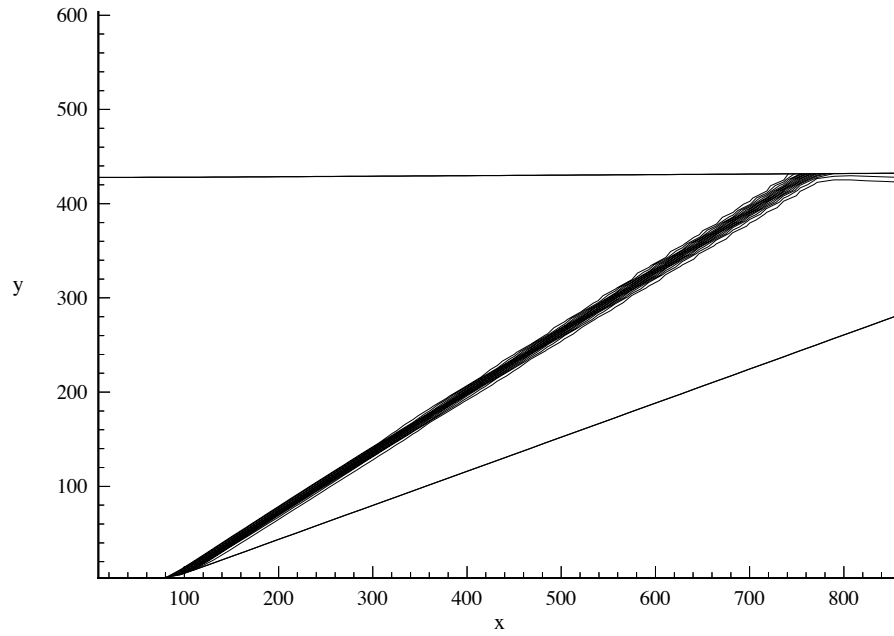


Figure 6.33: Pressure contours and velocity vectors for $\mathcal{M}_0 = 4$ and $K = 0.025$.

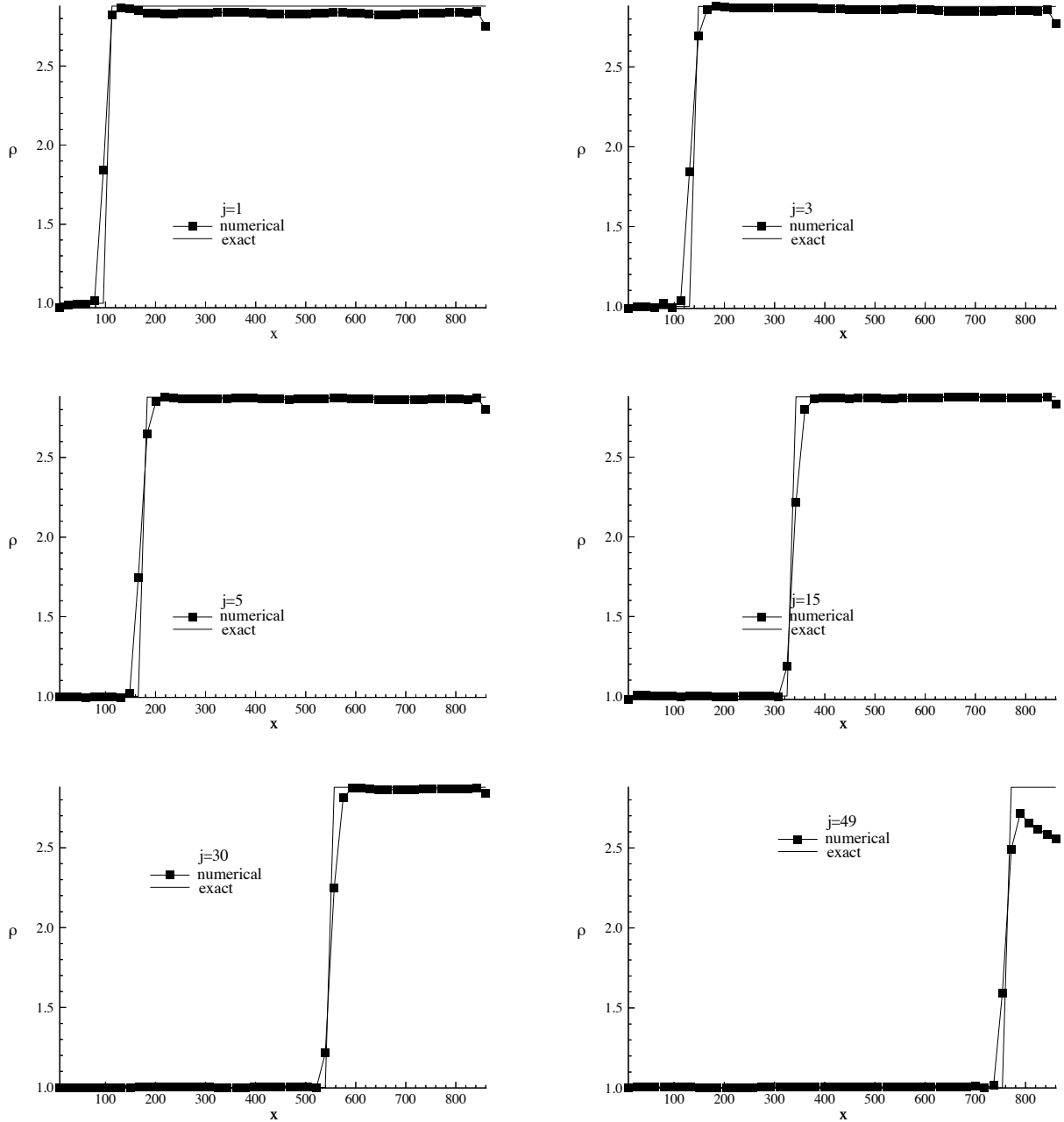


Figure 6.34: Comparison of numerical and exact density solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$.

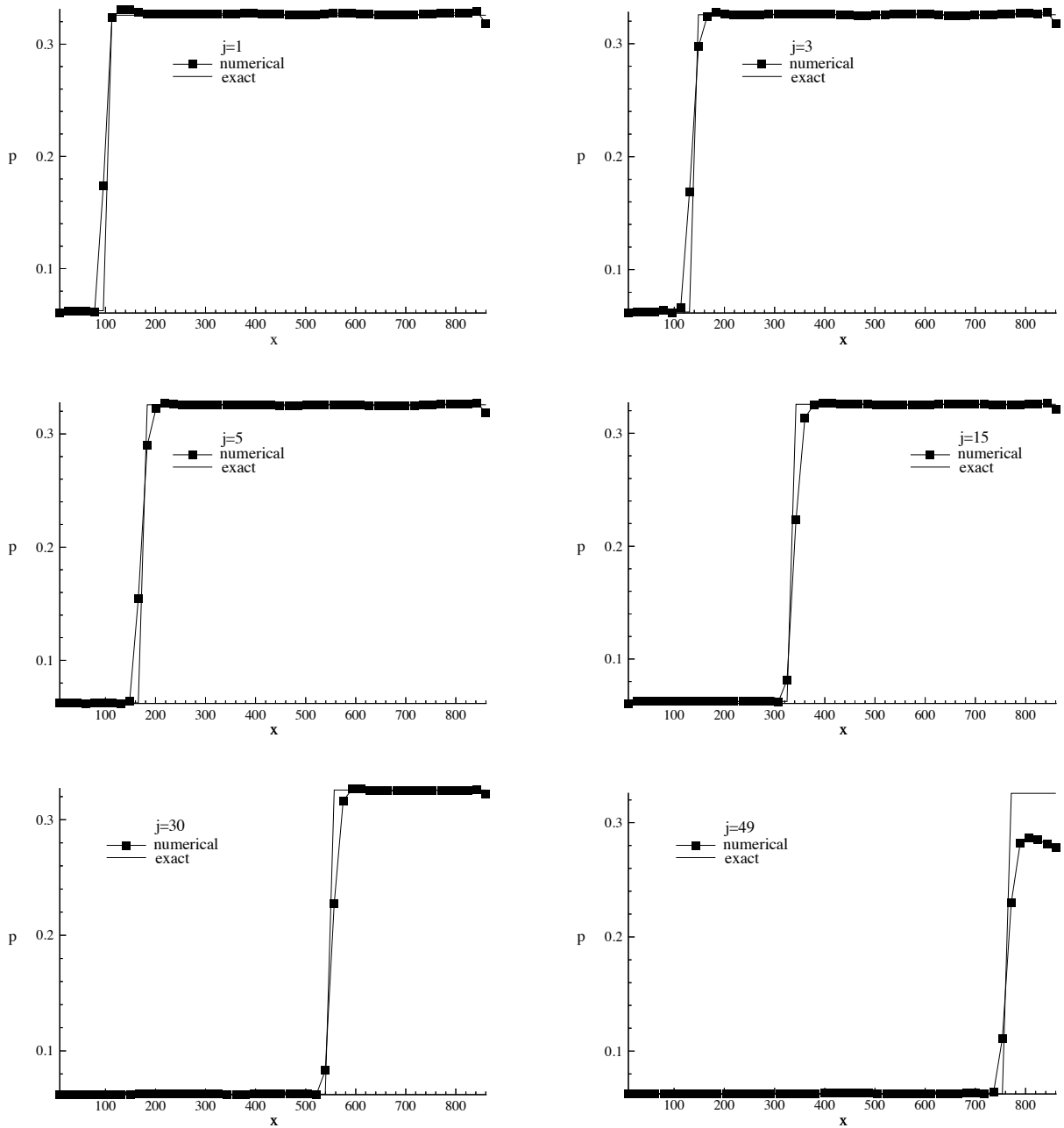


Figure 6.35: Comparison of numerical and exact pressure solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$.

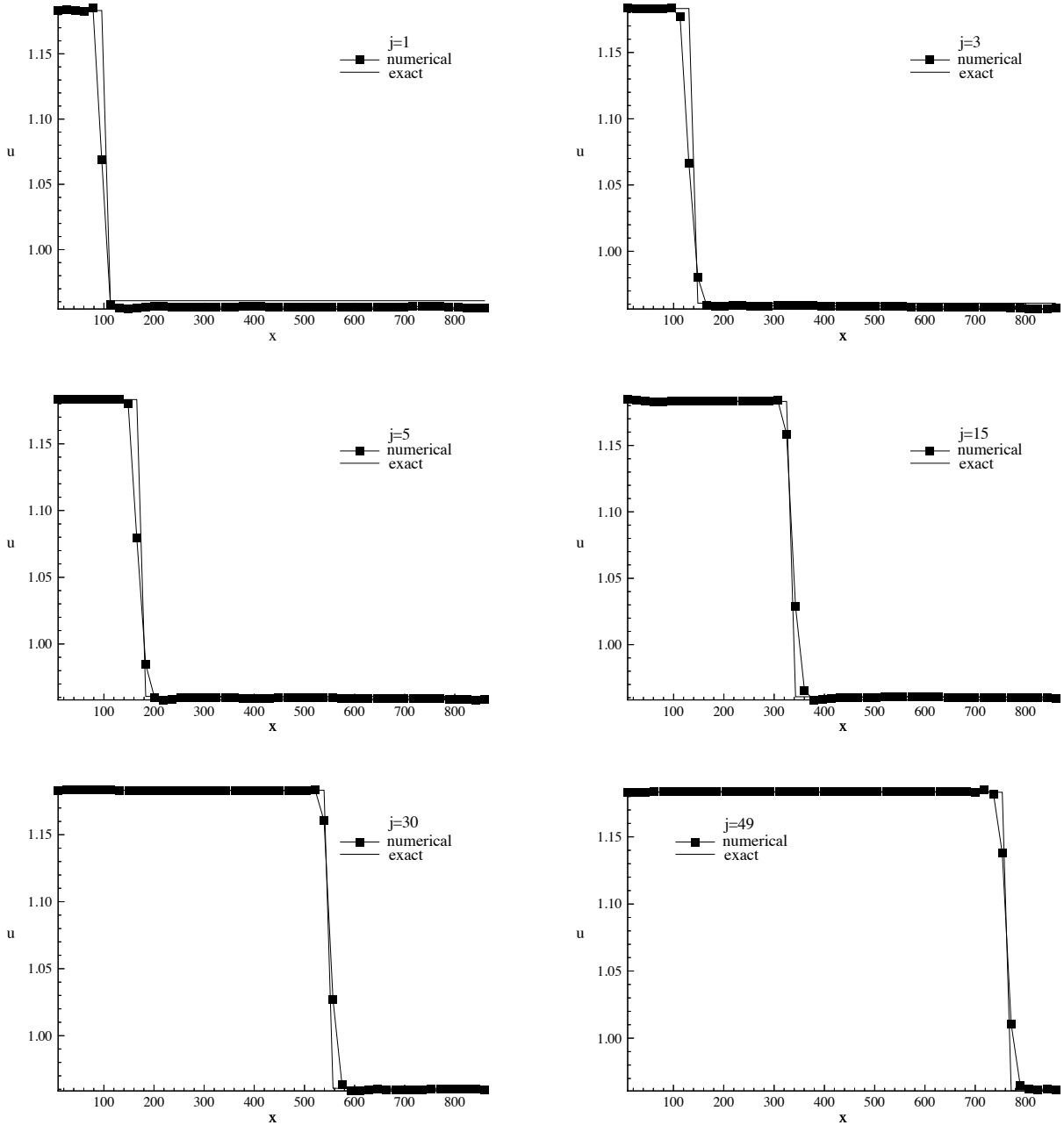


Figure 6.36: Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$.

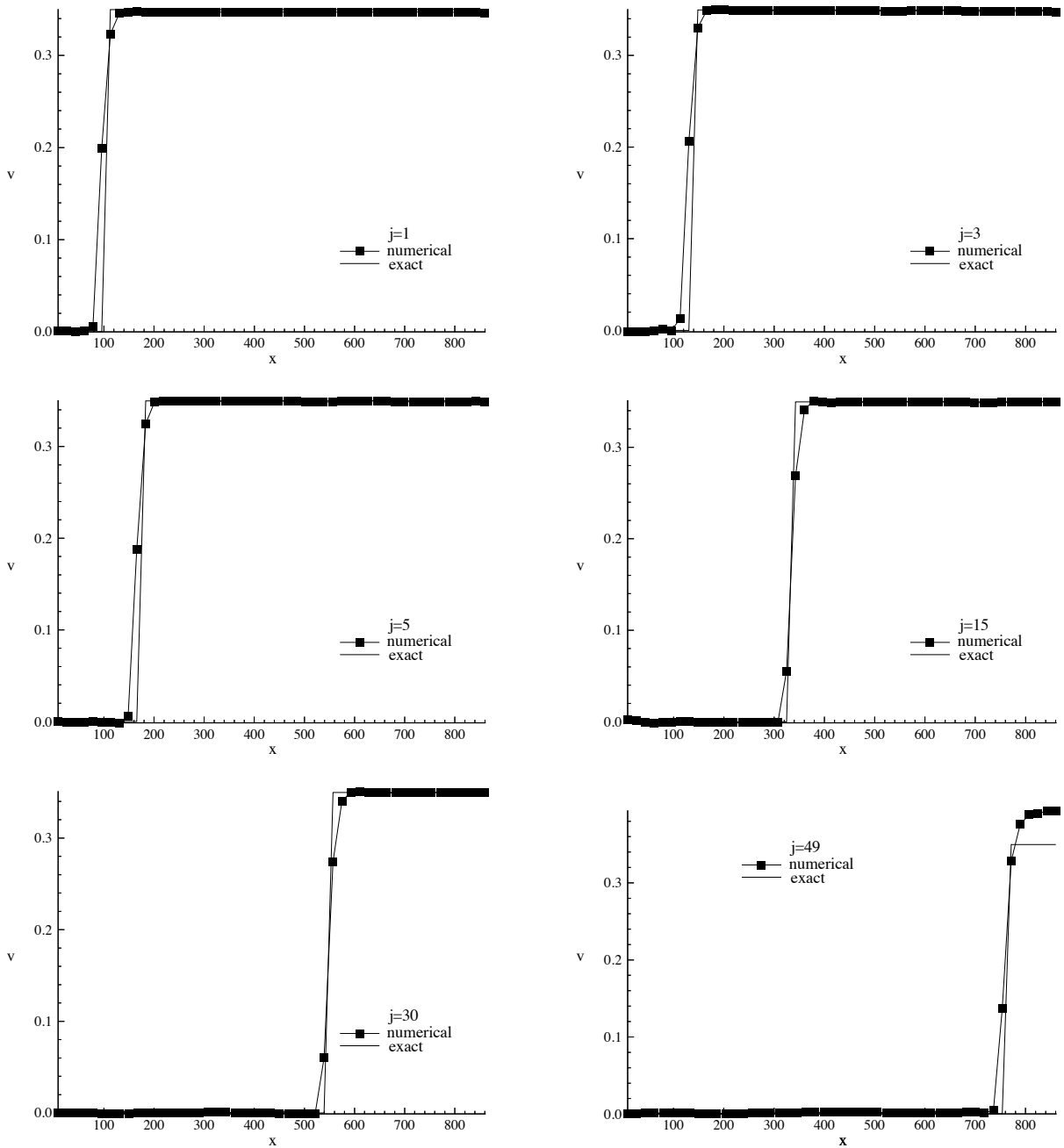


Figure 6.37: Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ gridlines for $\mathcal{M}_0 = 4$ and $K = 0.025$.

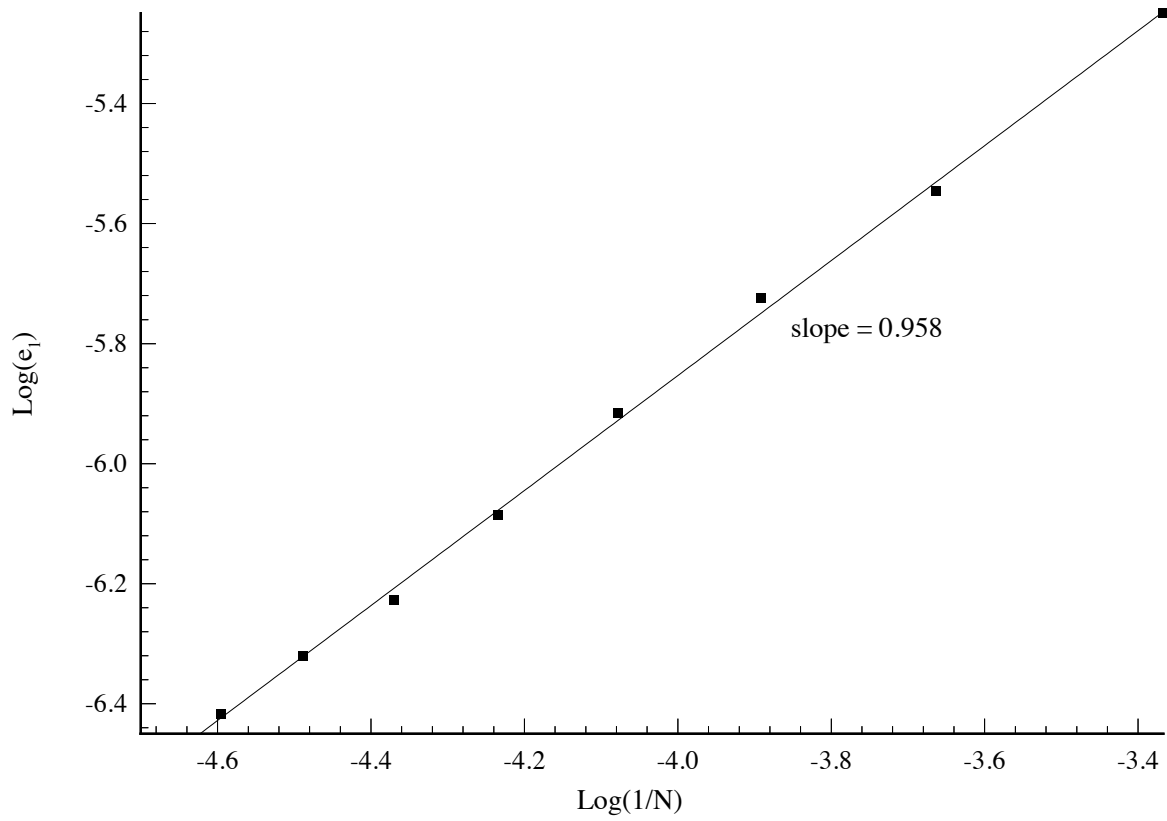


Figure 6.38: Measured order of accuracy for the second-order Roe scheme in two dimensions using the exact oblique shock solution as a basis ($\mathcal{M}_0 = 4$ and $K = 0.025$).

6.4 Straight Wall, Curved Oblique Detonation Problem

The asymptotic, curved oblique detonation solution of Powers and Stewart [63] was used as the final test case for the numerical algorithm. The nature of the asymptotic solution is such that chemical reaction only begins following the lead shock; there is no reaction in the freestream. For a fair comparison, this requires that the same be true for the numerical solution. For reactions modeled with the exponential dependence of Arrhenius kinetics, such as is done here, there is always some level of reaction present at all points in the flow. To eliminate this the Arrhenius kinetics were replaced with a “switch” in the numerical simulation. For densities below a certain value the right-hand side of Eq. 3.22 was set to zero, while for densities equal to or above this value the right-hand side was set to $\rho(1 - Y_2)$. The value of the switch density chosen was somewhat below the density rise due to an inert oblique shock at the given freestream Mach number, in this case $\mathcal{M}_0 = 20$.

Solutions were obtained for a wedge angle of $\theta = 20^\circ$ and heat releases of $\bar{q} = 10$. The grid resolution was 299 by 149 and CFL = 0.4. Figure 6.39 shows the pressure contours for the numerical solution that was obtained. The contour values were chosen so as to match those of the asymptotic solution shown in Fig. 4.12, which is replotted here for ease of comparison (see Fig. 6.40). The numerical solution indicates the correct trends in the pressure solution, with the indicated contours being similar to those seen on the asymptotic solution. Towards the rear outflow boundary the numerical solution appears to degrade somewhat, indicating that the small amplitude oscillations apparent in the inviscid oblique shock solutions are here also. Figure 6.41 shows wedge surface pressure comparisons between the two methods, as well as a comparison of the numerical and exact oblique shock solution. Both numerical solutions show a fairly large overshoot at the wedge tip, and then proceed to follow the asymptotic and exact solutions, respectively, for the remainder of the wedge surface.

Except for the overshoot, the agreement is very good in both cases. The larger than previously observed overshoot is apparently a result of the higher freestream Mach number. Slight oscillations are apparent in the numerical oblique detonation solution toward the end of the wedge. Figure 6.42 shows a comparison of the shock locations determined by the two methods. The asymptotic and numerical shock locations are relatively close to one another, but do not overlap. As discussed in Ch. 4, higher levels of heat release are increasingly inconsistent with the assumptions of the asymptotic solution, so the disagreement between the two solutions at $\bar{q} = 10$ is not unexpected.

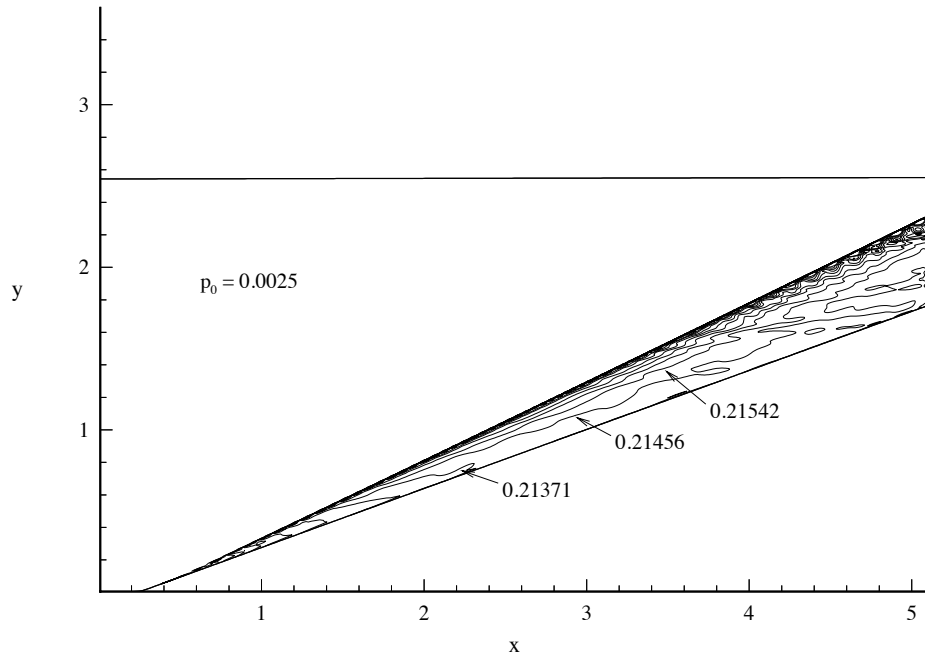


Figure 6.39: Oblique detonation pressure contours from Roe scheme numerical solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).

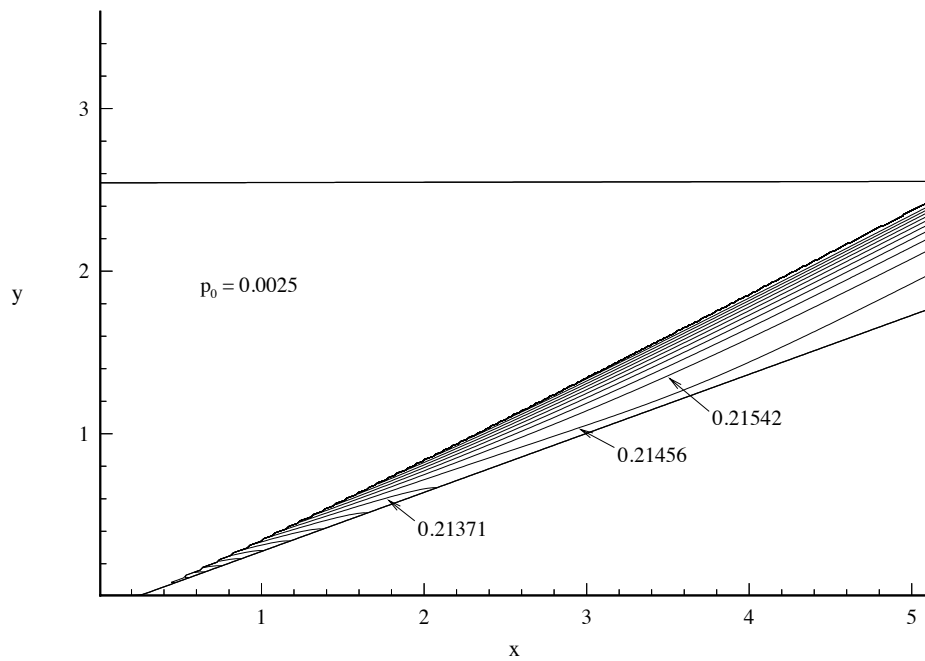


Figure 6.40: Oblique detonation pressure contours from asymptotic solution ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).

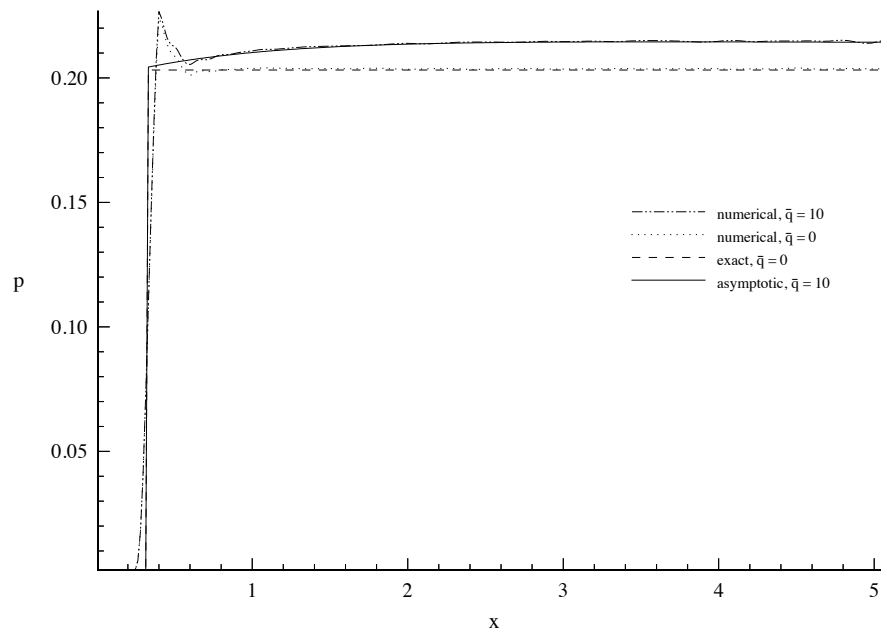


Figure 6.41: Comparison of surface pressures for numerical and asymptotic oblique detonation solutions, and numerical and exact oblique shock solutions ($\mathcal{M}_0 = 20$, $\gamma = 1.4$, $\theta = 20^\circ$).

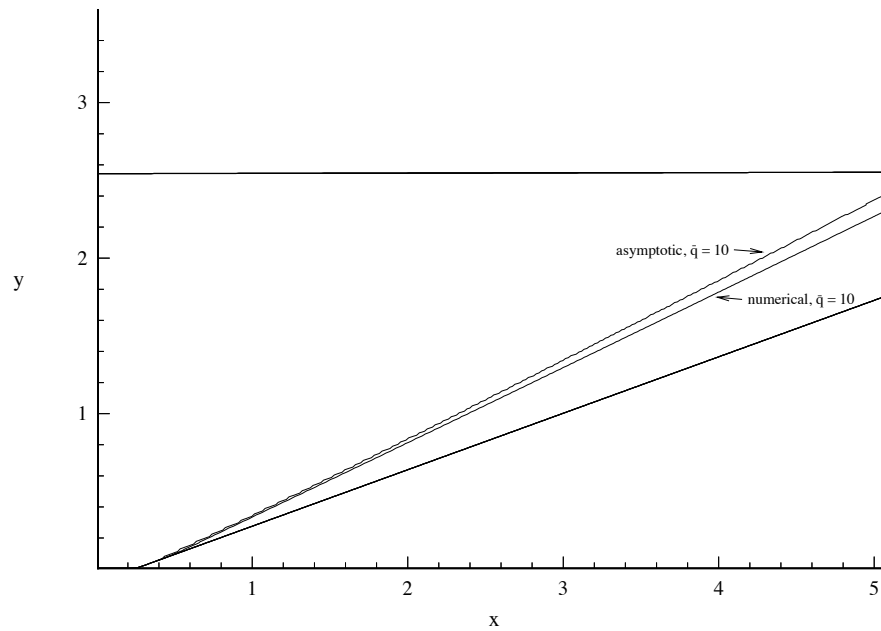


Figure 6.42: Comparison of shock locations for numerical and asymptotic oblique detonation solutions ($\mathcal{M}_0 = 20$, $\bar{q} = 10$, $\gamma = 1.4$, $\theta = 20^\circ$).

Chapter 7

UNSTEADY OBLIQUE DETONATIONS

This chapter will present the results obtained for both steady and unsteady oblique detonations. In order to make more direct comparisons with one-dimensional stability results, only straight shock, curved wall detonations were studied. As discussed in Ch. 4, the exact steady solution for such detonations is simply the steady one-dimensional detonation solution with an additional constant transverse velocity component. As such, the stability threshold of the oblique detonation is expected to be close to the stability threshold of the one-dimensional detonation. As for the one-dimensional detonation results, the parameter varied in this study was the normal overdrive, f_n ; the chapter has been divided into sections describing the results obtained for each value of f_n studied.

The solution procedure used for each case studied was as follows. An exact, weak overdriven oblique detonation solution was determined, along with the wall shape required to support it. This wall shape was entered into the computational mesh program, and smooth grids were generated having grid points concentrated near the wall. Figure 7.1 indicates a typical grid of cell centers used. The exact detonation solution was then written on the grid of cell centers. As indicated by the one-dimensional verification studies of Ch. 6, there is a need to maximize the number of points within the reaction zone of the detonation in order to successfully

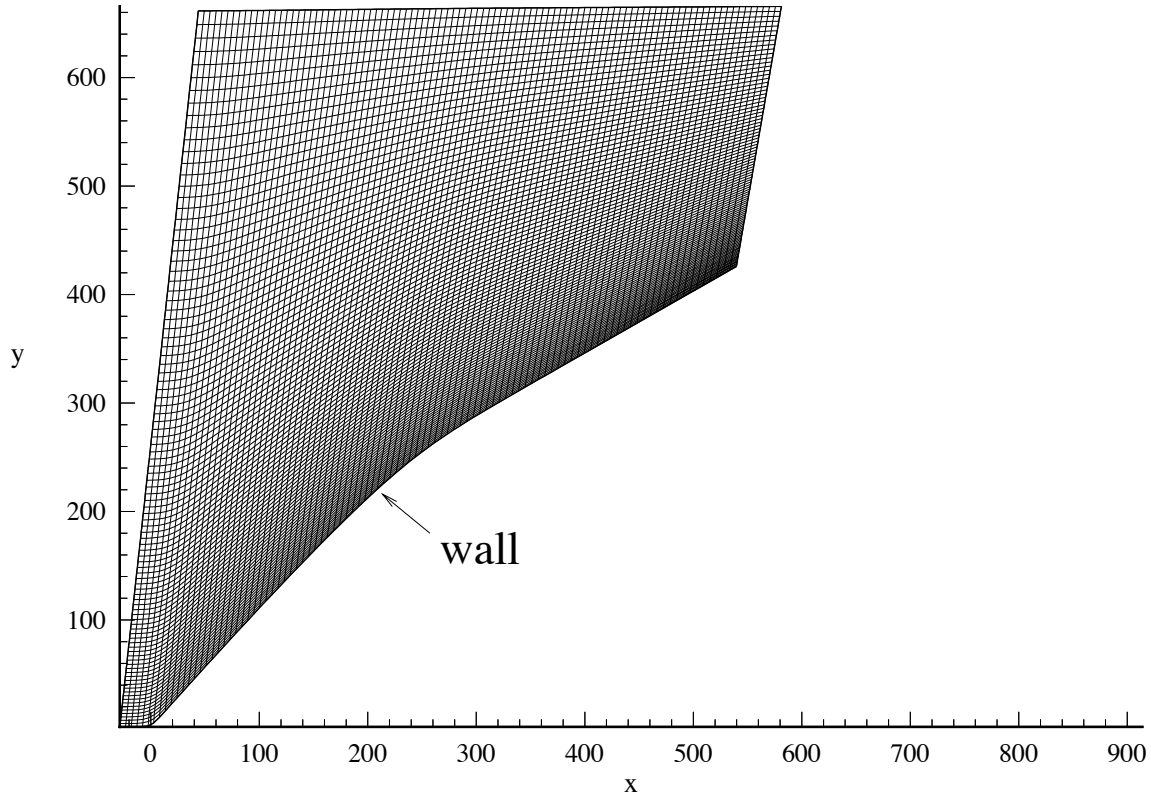


Figure 7.1: Typical grid of cell centers used in oblique detonation studies.

capture detonation instability. For this reason, the grid was generated such that the oblique detonation would exit through the top of the domain and not the rear; in this way it was possible to increase grid resolution within the the reaction zone of the detonation while minimizing the total number of points needed for the computation. An additional parameter used to size the computational grid was the ratio of the initial shock length within the domain to the initial half reaction zone width, $L_{ratio} = L_{sh}/L_{1/2}$. The exact solution on the grid was then used as an initial condition for the simulation, which was run continuously until the solution had reached a steady state or could not continue. Steady state was typically achieved when the residuals of the flow variables had decreased by 2-3 orders of magnitude. During execution,

data files were written every 200 time steps to follow the time-dependent nature of the solution. Complete solutions required anywhere from a $\frac{1}{2}$ day to several days to finish when run on one of the IBM RS/6000 workstations.

In all of the cases studied, the following parameters were held constant: $\bar{E}_a = 50$, $\bar{q} = 50$, $\gamma = 1.2$, $K = 0.025$, and $\text{CFL} = 0.4$ (unless otherwise noted). Additionally, the same constant dimensional tangential velocity of $\tilde{v} = 1379 \text{ m/s}$ was used in all cases to generate the exact solution. Different levels of normal overdrive f_n were thus achieved by varying the freestream Mach number \mathcal{M}_0 . The values of heat release, activation energy, γ , and CFL were chosen to match those used in the unsteady one-dimensional detonation; the tangential velocity was chosen so the exact detonation solutions would be weak and overdriven. A value of $\tilde{B} = 1.0 \times 10^6 \text{ s}^{-1}$ was chosen simply to determine nondimensional values for the coordinate axes and the time; it has no effect upon the solutions obtained. The value of K was chosen based on the results of the verification studies in Ch. 6.

7.1 Normal Overdrive $f_n = 2.0$

This level of overdrive was achieved with $\mathcal{M}_0 = 9.822$. The resulting detonation wave angle was $\beta = 63.5^\circ$, and the wedge had an initial angle of $\theta_{max} = 51.9^\circ$. According to the one-dimensional theory, this case should be well within the stable regime. Figures 7.2, 7.3, and 7.4 show shaded contours of the time-dependent solution obtained on a grid of 99 by 99 cells. There were approximately 10 points in the initial half reaction zone length $L_{1/2}$ of the exact solution and $L_{ratio} = 21.5$. For this and all remaining plots of this type, the convention will be the same as that followed when reading English text; the upper left plot is at the earliest time, the lower right is at the latest time, and time advances from left to right and down. The contours plotted were chosen based upon the interesting phenomena in the solution, and are not necessarily sequential or equally spaced in time.

The detonation front is apparent in each of the plots as the abrupt change from the white freestream value to some shade of gray. As shown in Fig. 4.8, the exact steady solution is characterized by straight contours parallel to the detonation front. It is apparent in the earliest time shown in the figures that the solution has begun to deviate from the exact solution. This small deviation progresses to become a concentrated region of high density and pressure, as well as increased reaction rate, at the detonation front. As the pressure and density increase in this region, it slowly progresses up the length of the detonation, eventually passing out of the domain. After the disturbance exits, the upper half of the detonation front begins to move forward (to the left) slowly, leading to the formation of a second disturbance approximately midway along the length of the front. This disturbance travels down along the detonation front, eventually hitting the wedge surface. The passage of the disturbance results in the detonation moving forward into the freestream until it encounters the inflow boundary. At this point the simulation was stopped, as the inflow boundary condition was not configured to handle such an occurrence. A more detailed description of this general phenomena will be given for the case of $f_n = 1.6$.

Since this unsteady behavior was not expected for this level of overdrive, two more cases were run, one with approximately 12 points per $L_{1/2}$ (124 by 124 cell centers) and the other with 14 points per $L_{1/2}$ (149 by 149 cell centers). In both cases the second, downward travelling disturbance did not form, and the solutions relaxed to a steady state nearly identical to the exact steady solution. Figure 7.5 shows the time history of the residuals for density and the u component of velocity; both residuals decrease approximately three orders of magnitude (note that the curves actually begin at the upper left of each plot, but are masked by the y axes). Figures 7.6, 7.7, and 7.8 show the time-dependent solutions obtained for density, pressure, and product mass fraction on the grid of 149 by 149 cell centers. Similar to the previous less resolved case, an initial disturbance forms and propagates out the top of the domain. In this

case, however, the detonation front returns to its initial straight state following the passage of the disturbance.

Figure 7.9 shows the velocity vectors for the final, steady solution. As expected, these change discontinuously through the detonation front and parallel the wall boundary. Note that the vectors within the freestream only appear to curve because they are plotted at cell centers which lie along curved grid lines. Figure 7.10 compares the final solution contours for density with the exact solution contours using the same contour levels. This quickly illustrates the slight discrepancies in the numerical solution. The numerical contours closely parallel the exact contours, with increasing discrepancies near the top of the domain. Along the wall there appears to be a slight layer in the numerical solution, and in the upper right corner there is an extra contour due to the outflow condition. Figures 7.11, 7.12, 7.13, 7.14, and 7.15 compare the numerical and exact solutions for ρ , u , v , p , and Y_2 , respectively, along $j = \text{constant}$ gridlines. As before, $j = 1$ is the row of cell centers nearest the wall, while in this case $j = 149$ is the last row of cell centers at the top of the domain. It is immediately apparent that there is a large spike in both the density and pressure at the wedge tip along the wall. As in the verification studies, this is a result of the discontinuity in the grid at the wedge tip; the grid was extended a slight distance in front of the wedge tip to allow for some movement of the detonation in this region, which results in a grid discontinuity where the wedge begins. By the $j = 3$ gridline the spike has disappeared, and the numerical solution relatively closely tracks the exact solution. Along $j = 15$ all of the flow variables show excellent agreement with the exact solution. As expected, the agreement along the topmost grid line $j = 149$ has degraded due to the outflow boundary condition; however, the numerical solutions for ρ , u , and Y_2 , are relatively close to the exact solution. Finally, there are slight oscillations in the numerical solution, particularly for the pressure and v component of velocity.

Overall, there appears to be good agreement between the steady numerical solution and the exact solution. As in the verification studies, the top outflow boundary condition and the wall boundary condition induce the largest discrepancies in the numerical solution. The two steady numerical solutions would seem to indicate that at this level of overdrive, the oblique detonation is stable. The instability observed for the least resolved case was apparently the result of an insufficient number of points in the half reaction zone length.

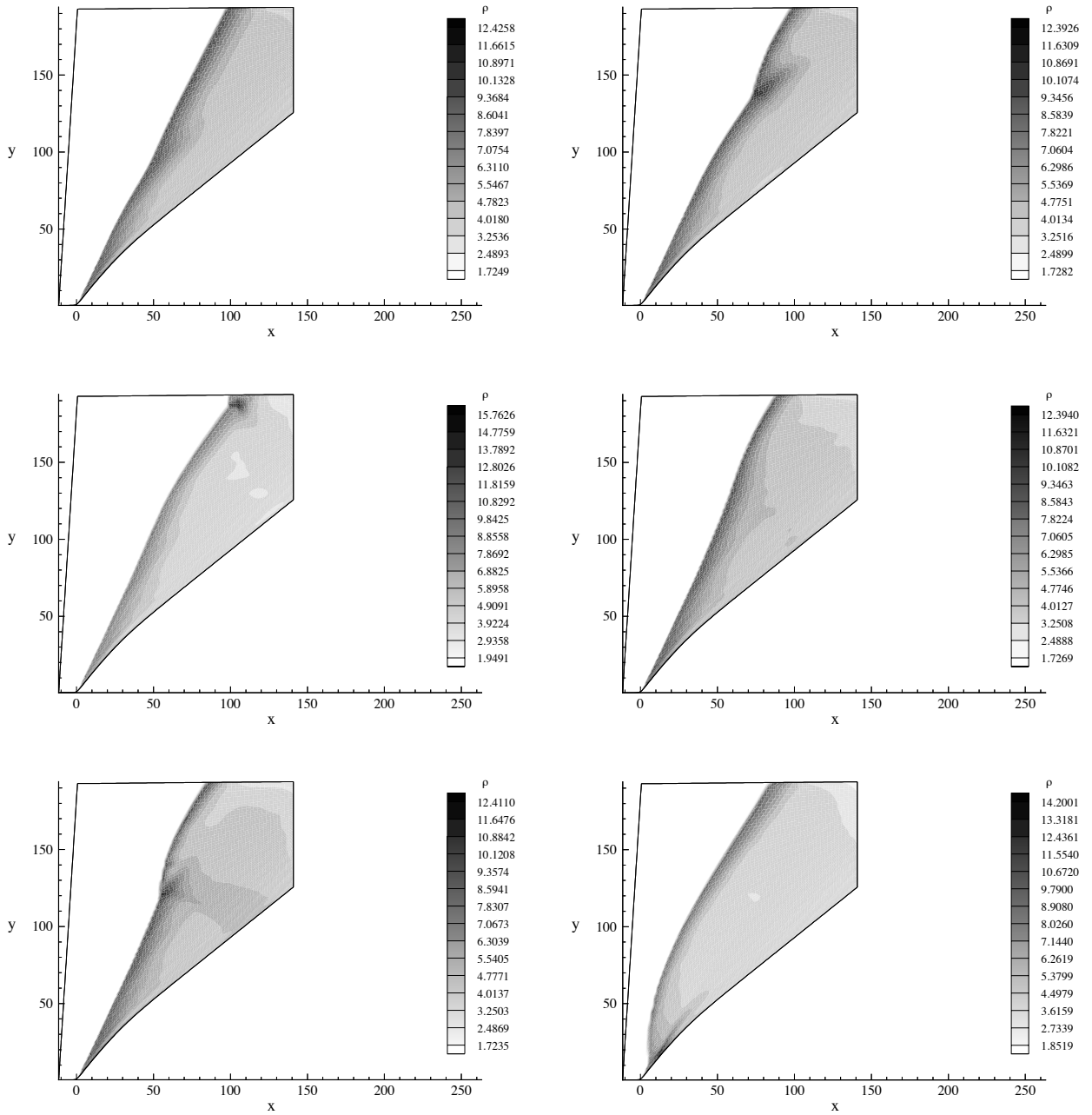


Figure 7.2: Shaded density contours as a function of time ($f_n = 2.0$, 99 by 99 grid, $t = 409.8, 657.5, 1111.6, 2066.9, 2605.0, 3687.0$).

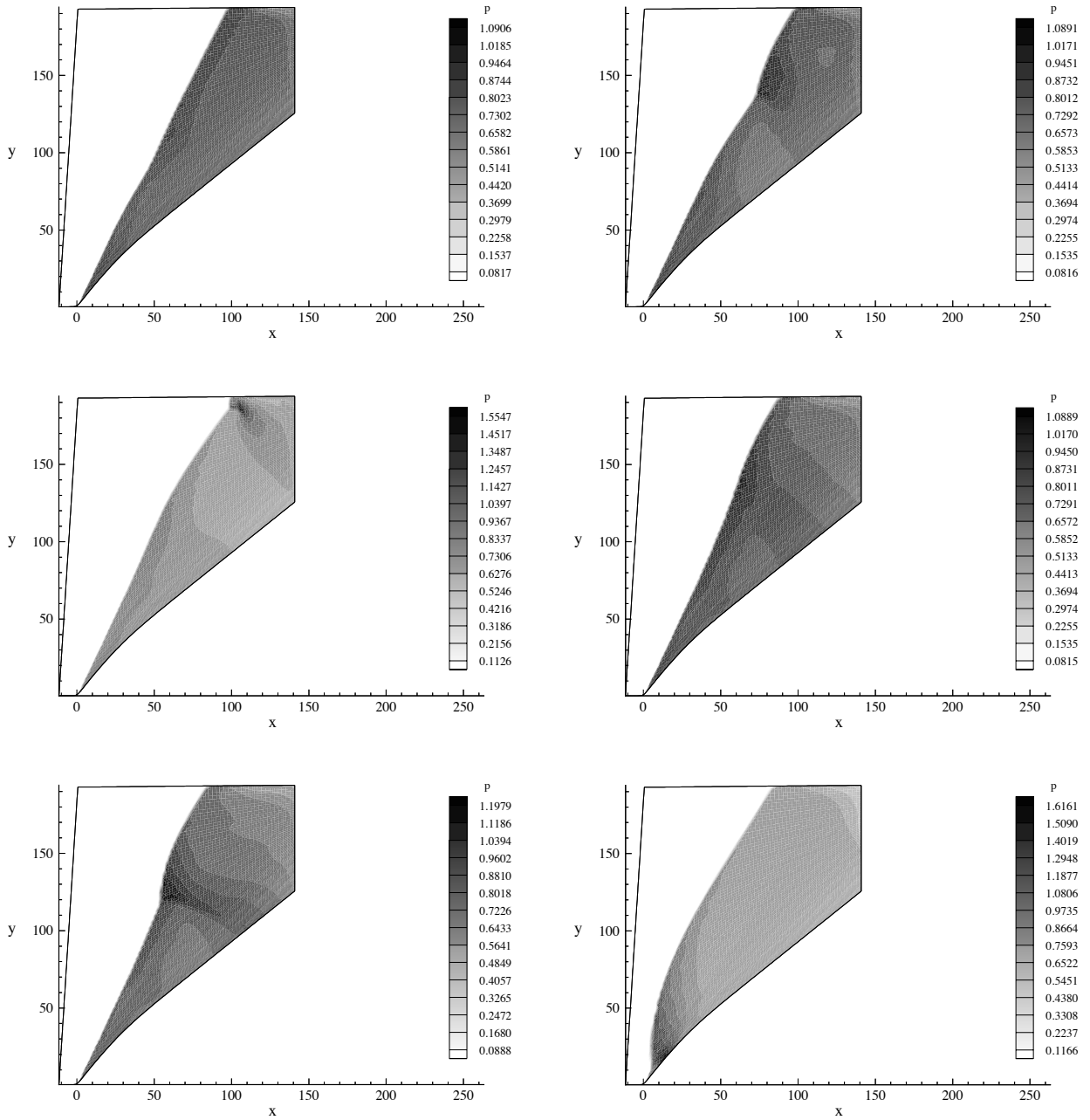


Figure 7.3: Shaded pressure contours as a function of time ($f_n = 2.0$, 99 by 99 grid, $t = 409.8, 657.5, 1111.6, 2066.9, 2605.0, 3687.0$).

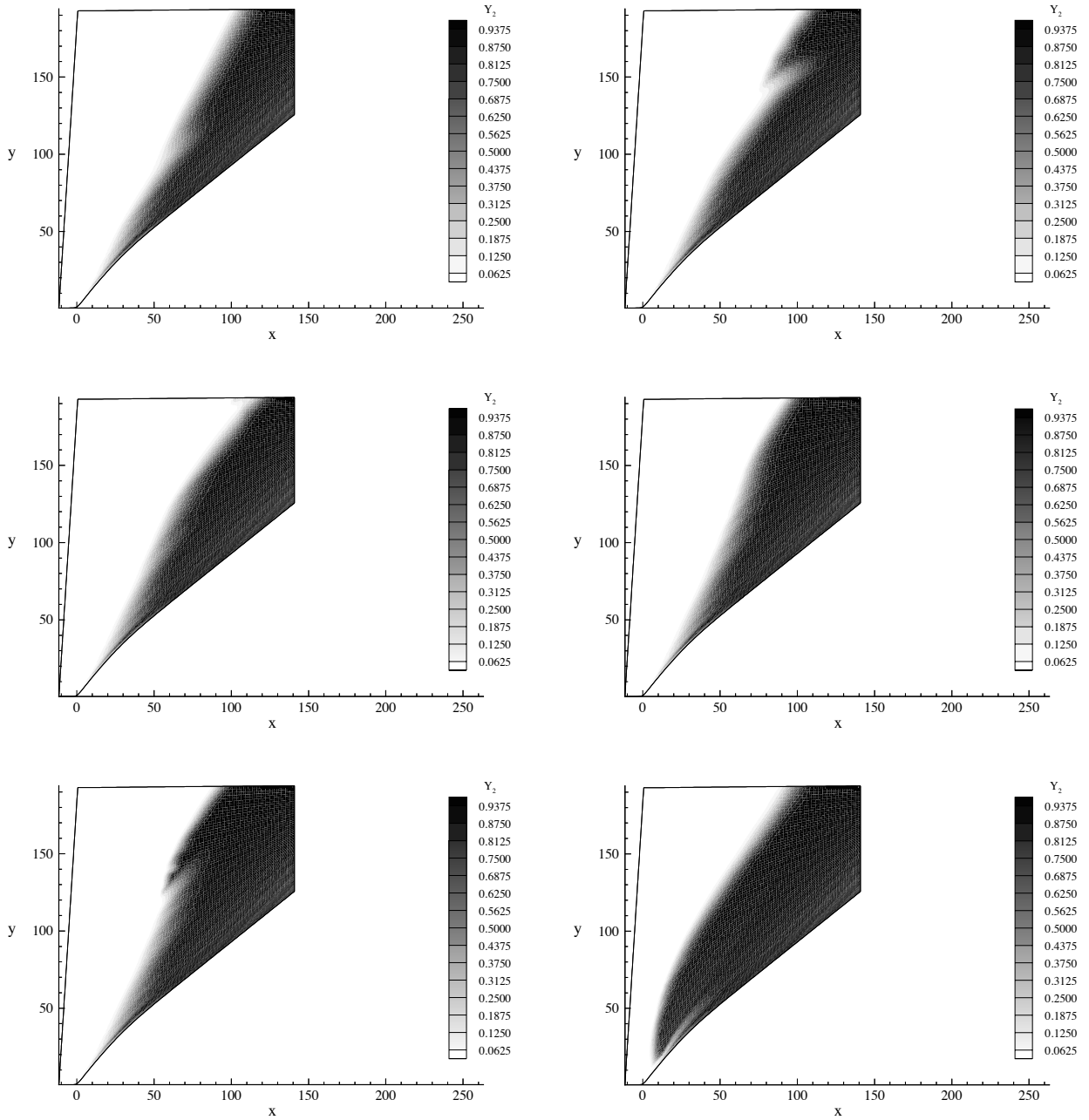


Figure 7.4: Shaded product mass fraction contours as a function of time ($f_n = 2.0$, 99 by 99 grid, $t = 409.8, 657.5, 1111.6, 2066.9, 2605.0, 3687.0$).

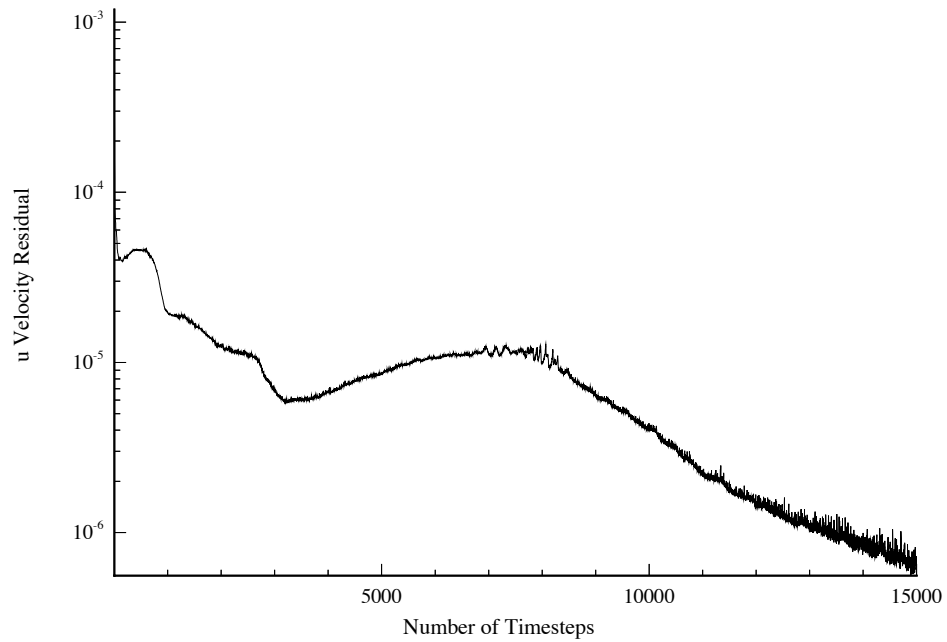
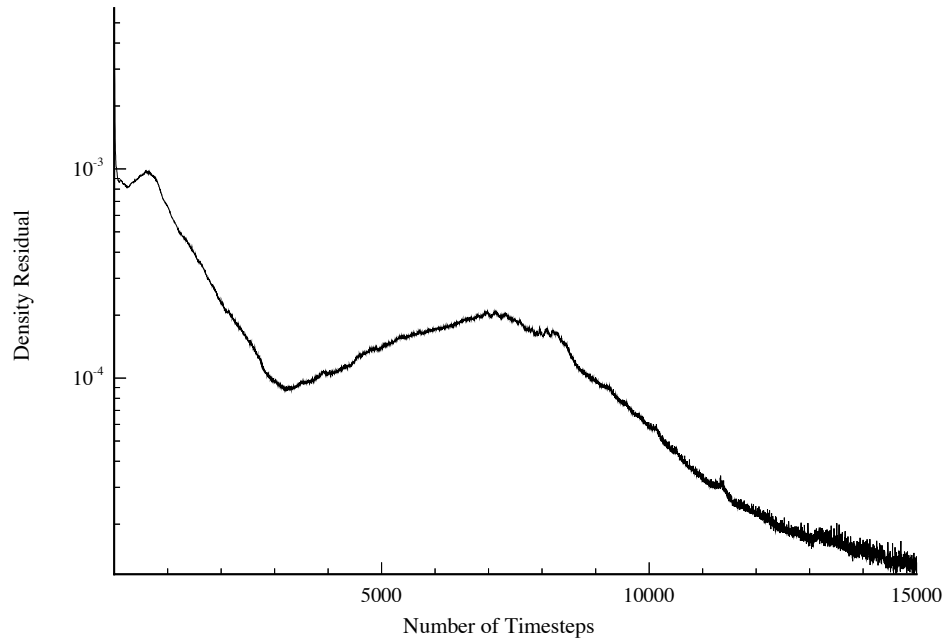


Figure 7.5: Time histories of density and u component of velocity residuals ($f_n = 2.0$, 149 by 149 grid).

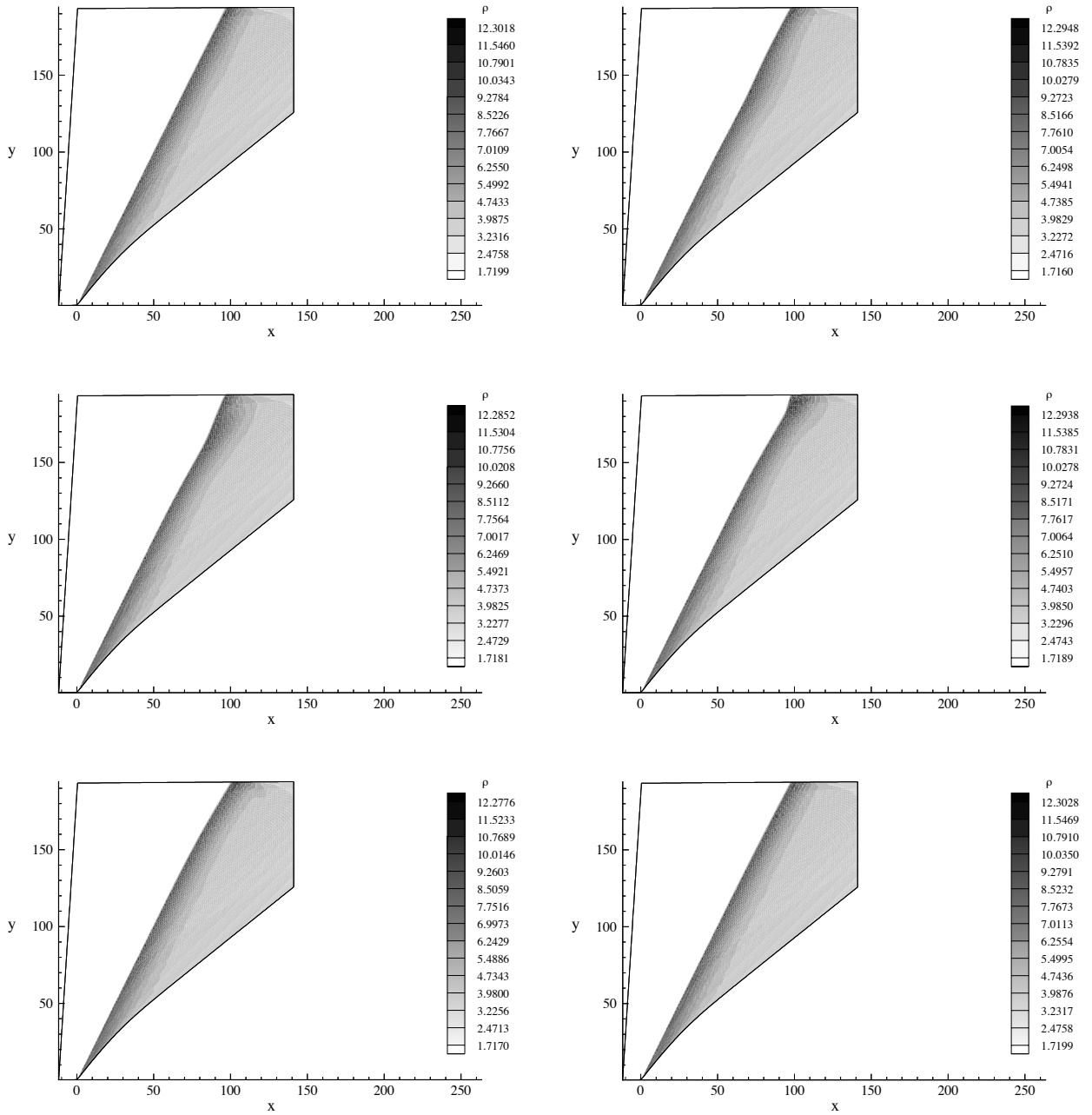


Figure 7.6: Shaded density contours as a function of time ($f_n = 2.0$, 149 by 149 grid, $t = 246.8, 645.2, 845.0, 969.9, 1094.8, 1869.6$).

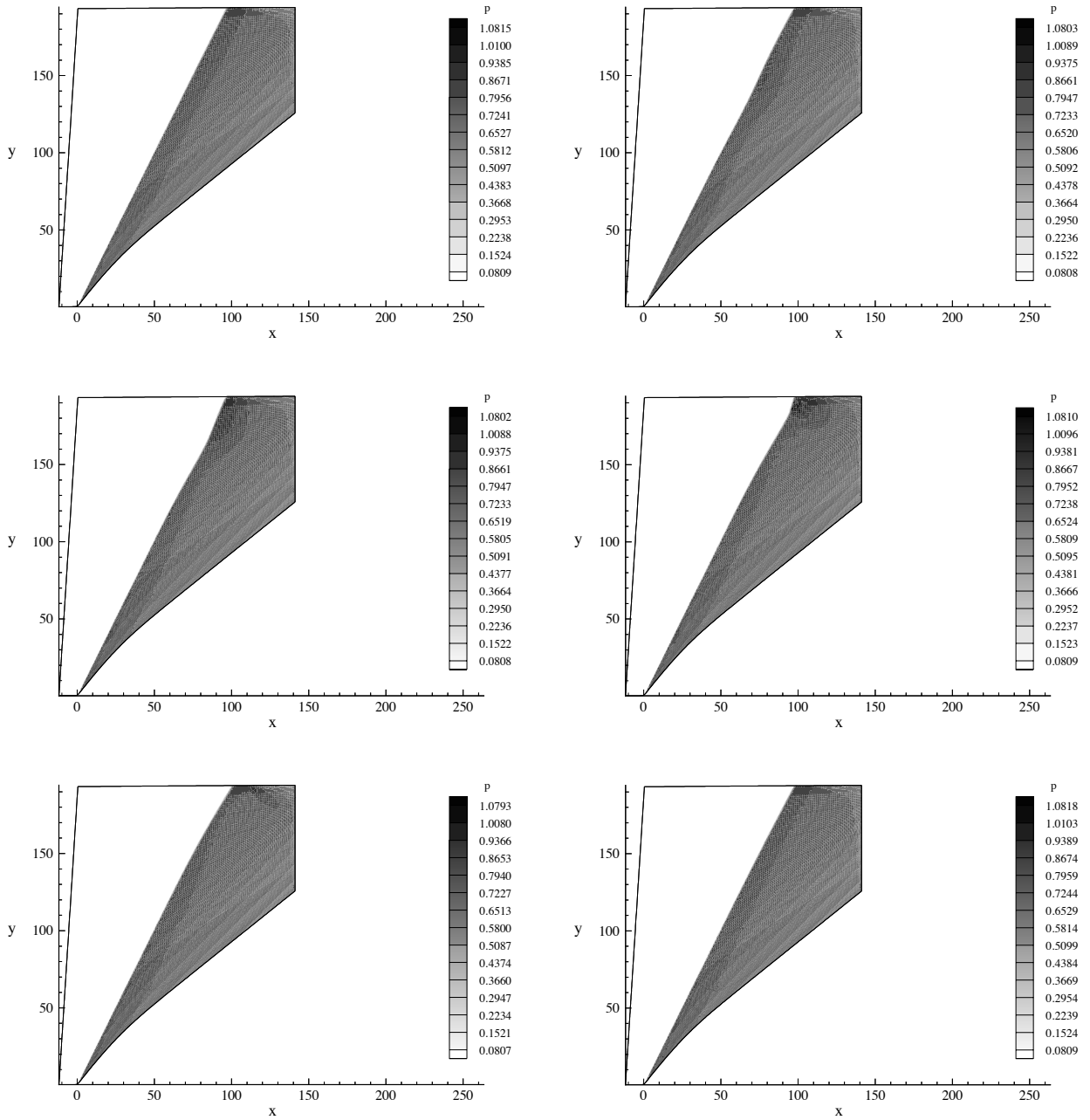


Figure 7.7: Shaded pressure contours as a function of time ($f_n = 2.0$, 149 by 149 grid, $t = 246.8, 645.2, 845.0, 969.9, 1094.8, 1869.6$).

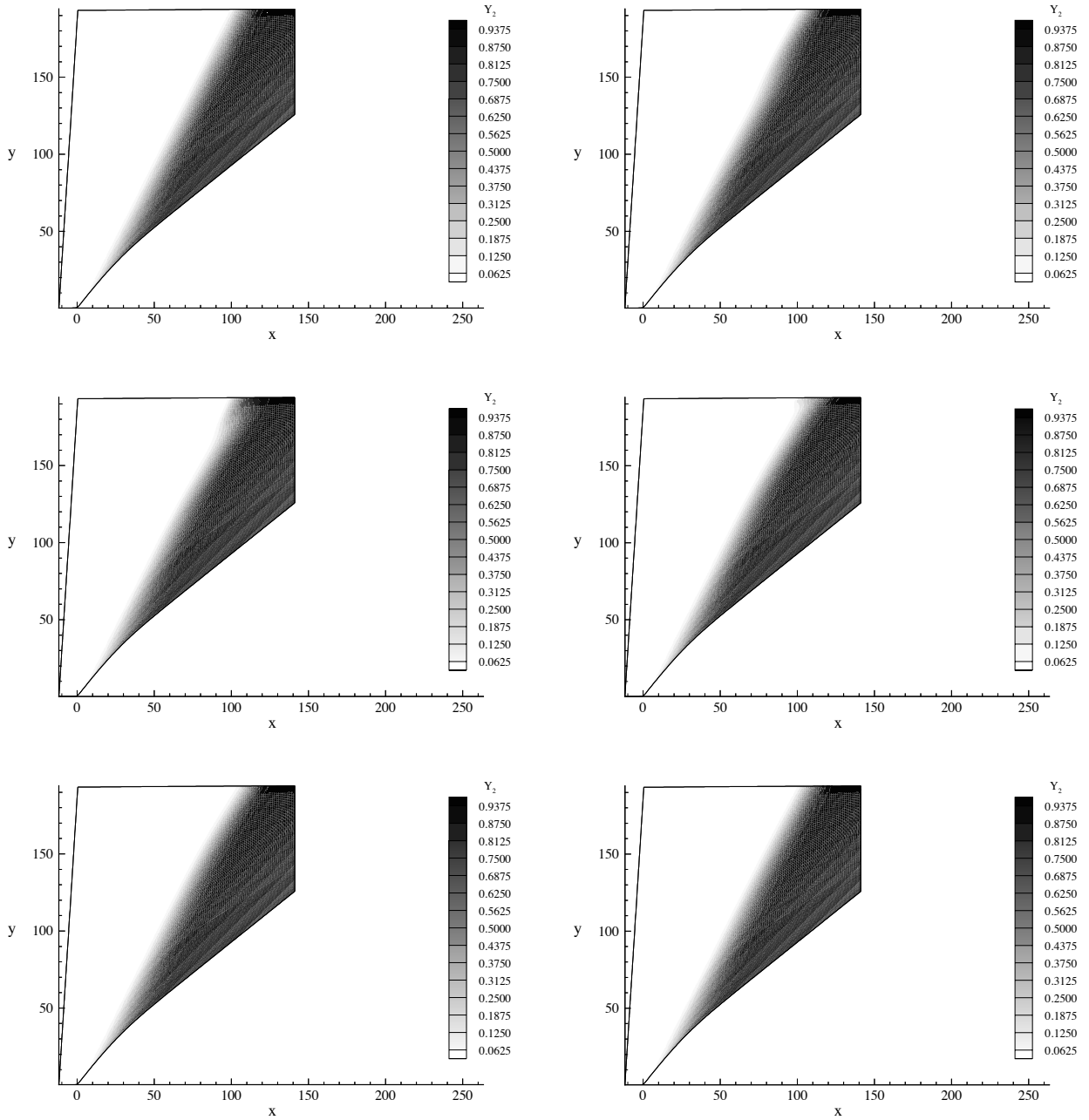


Figure 7.8: Shaded product mass fraction contours as a function of time ($f_n = 2.0$, 149 by 149 grid, $t = 246.8, 645.2, 845.0, 969.9, 1094.8, 1869.6$).

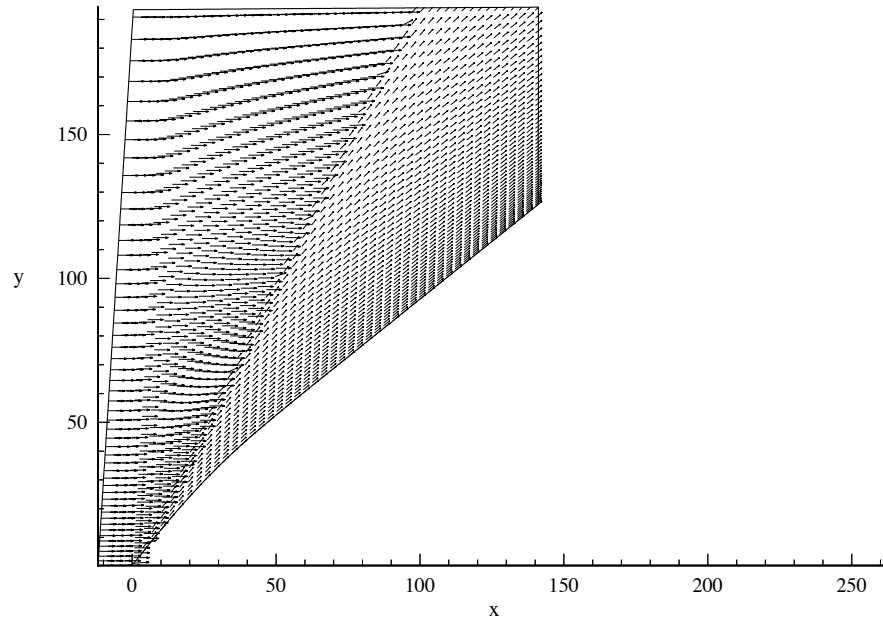


Figure 7.9: Velocity vectors for final, steady solution ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).

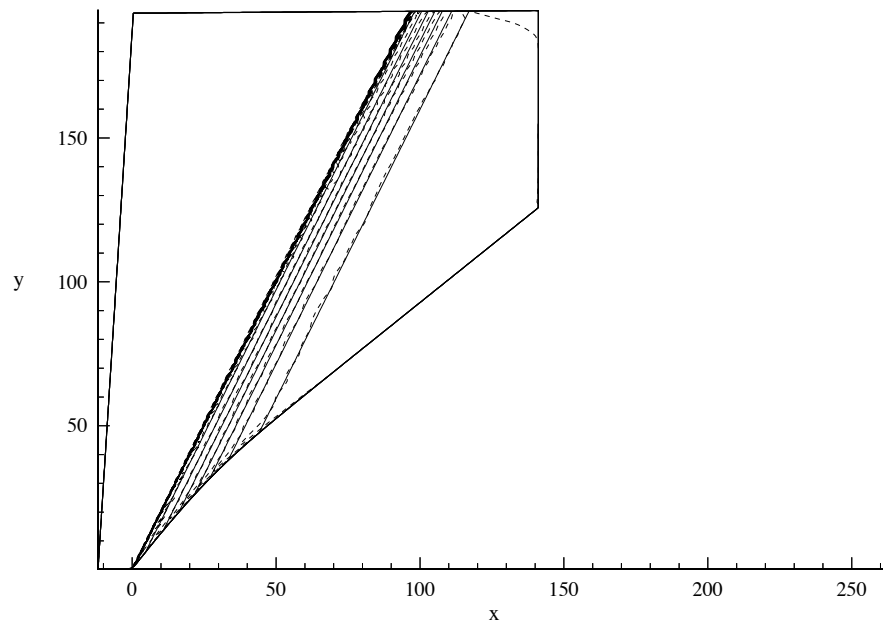


Figure 7.10: Comparison of numerical and exact density contours ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$, dashed = numerical, line = exact).

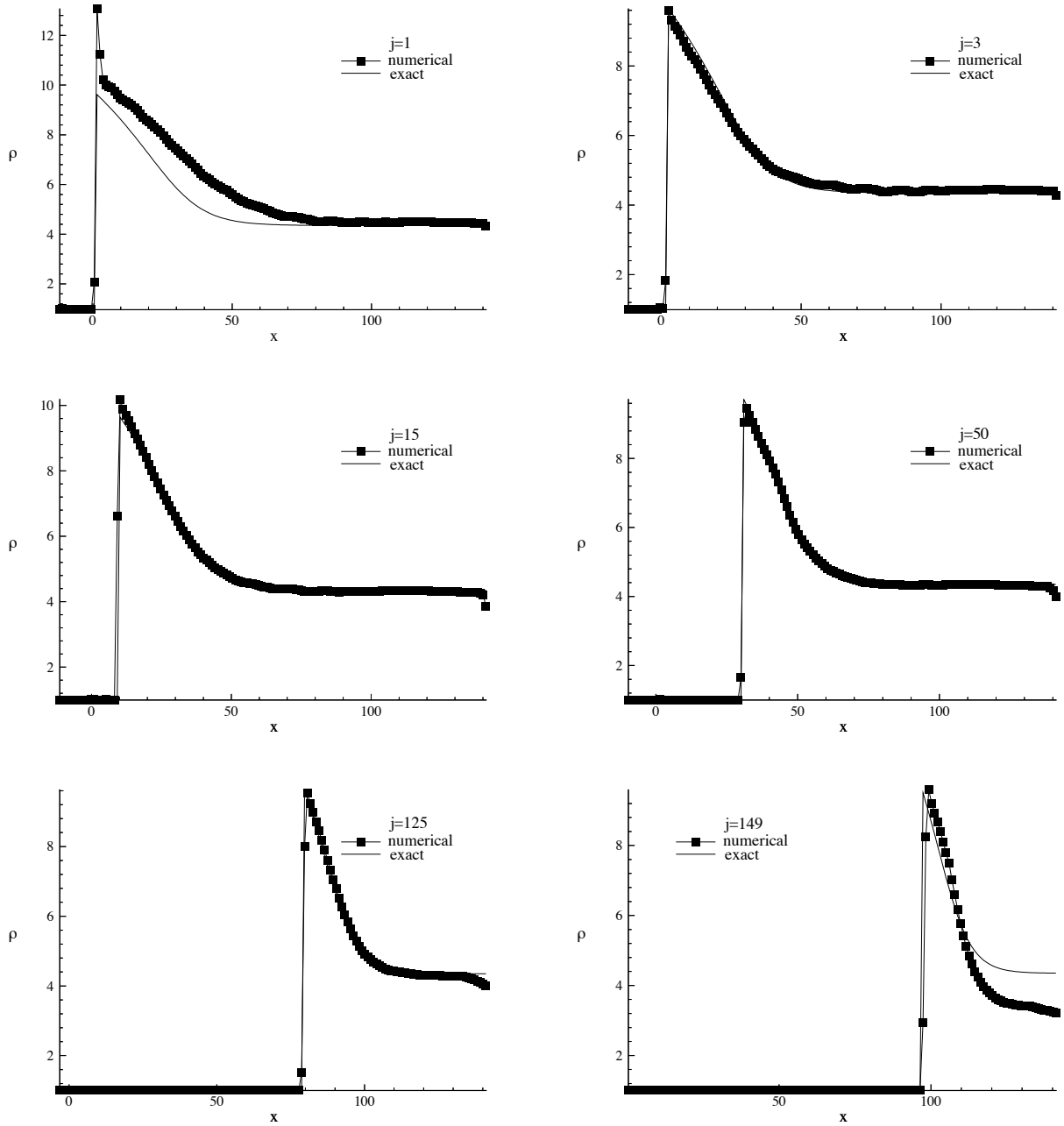


Figure 7.11: Comparison of numerical and exact density solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).

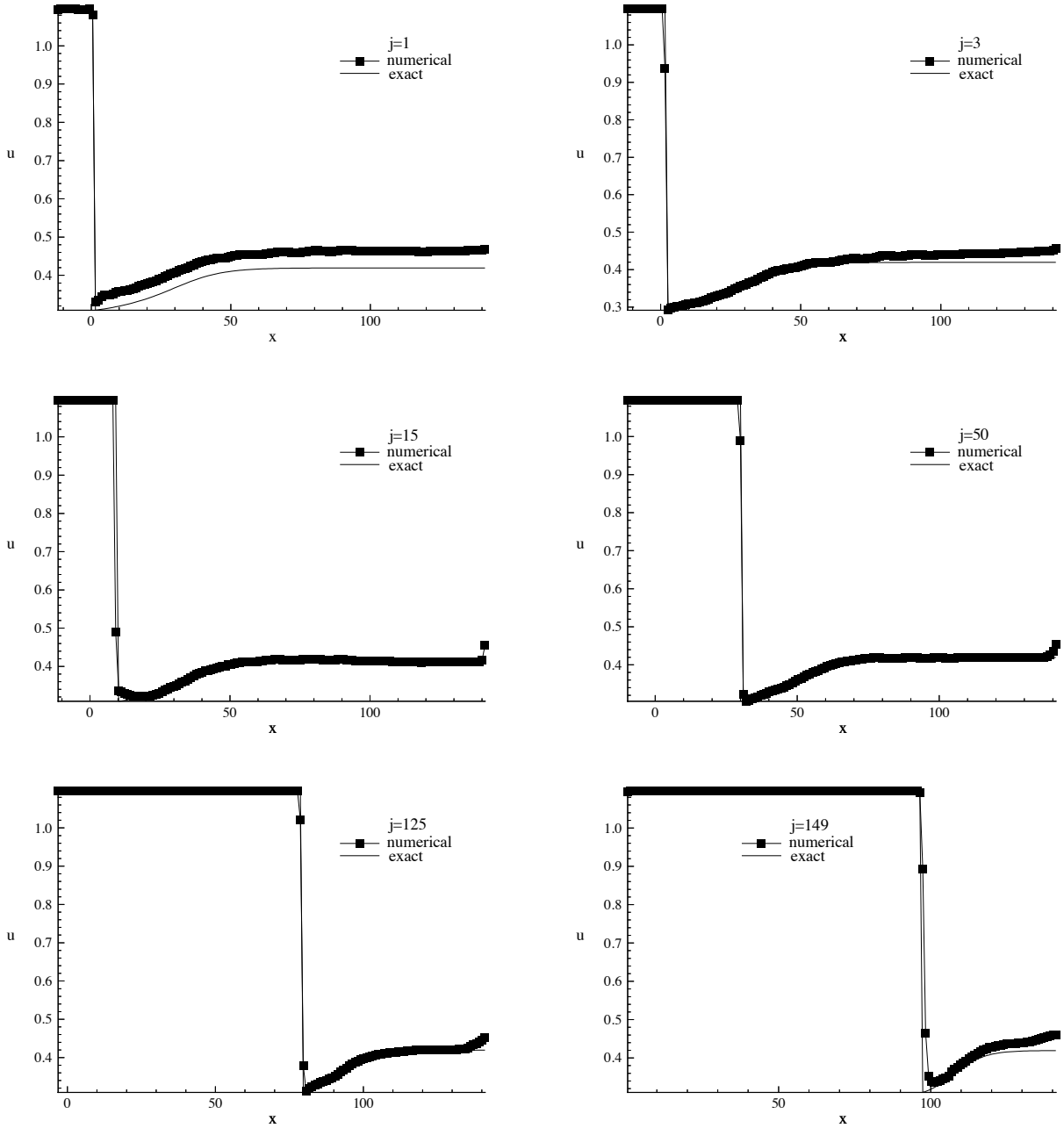


Figure 7.12: Comparison of numerical and exact u velocity component solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).

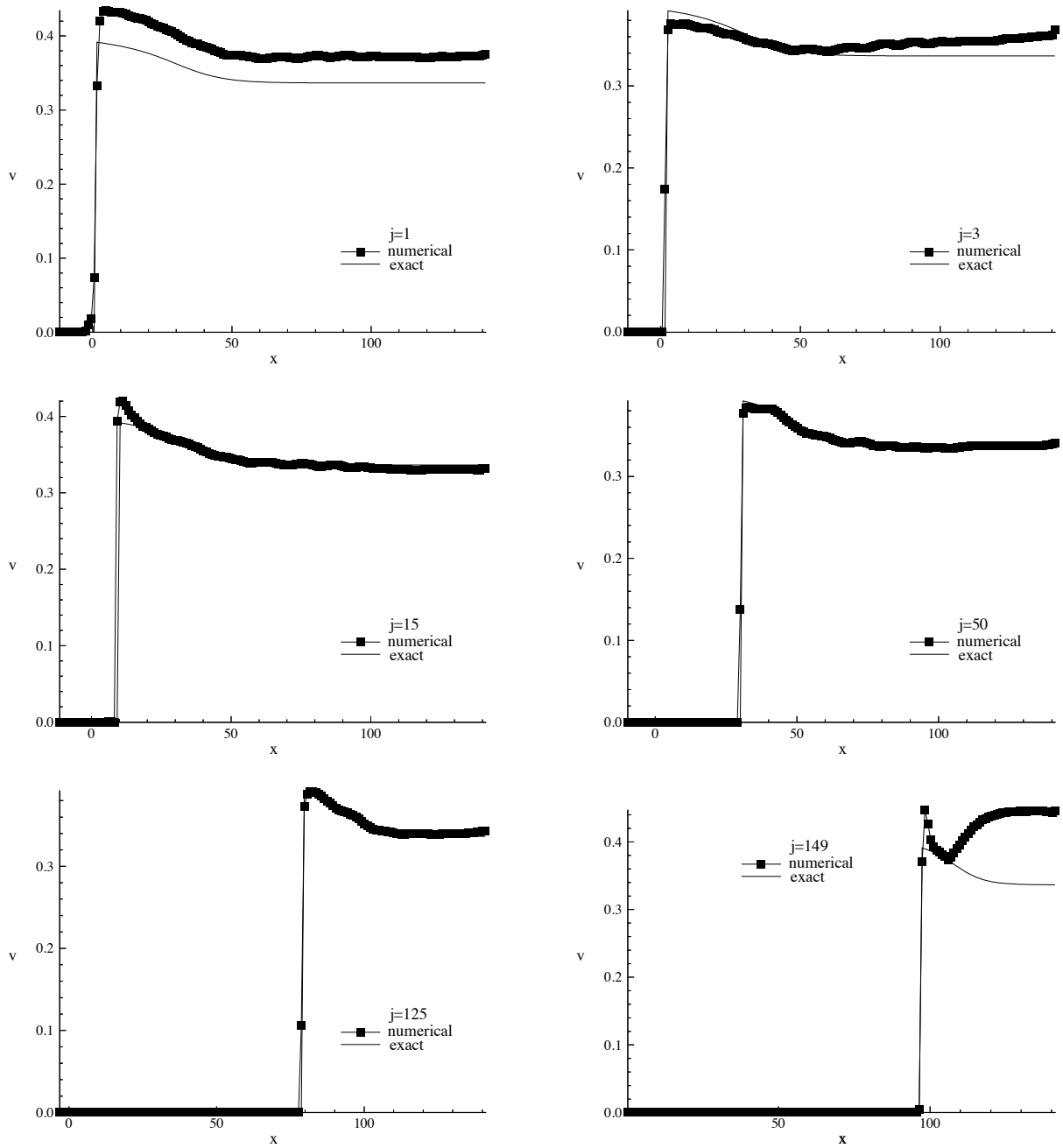


Figure 7.13: Comparison of numerical and exact v velocity component solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).

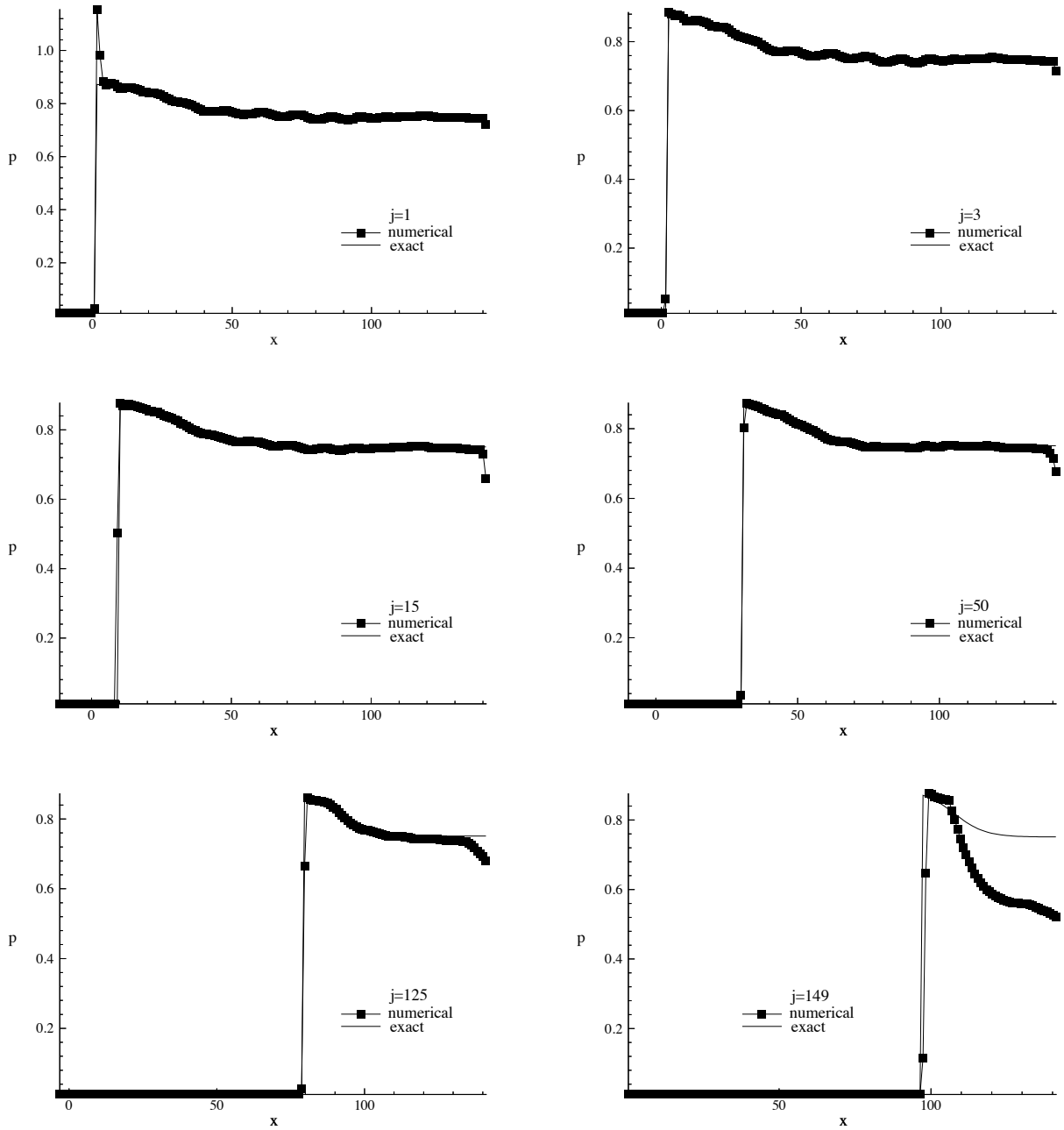


Figure 7.14: Comparison of numerical and exact pressure solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).

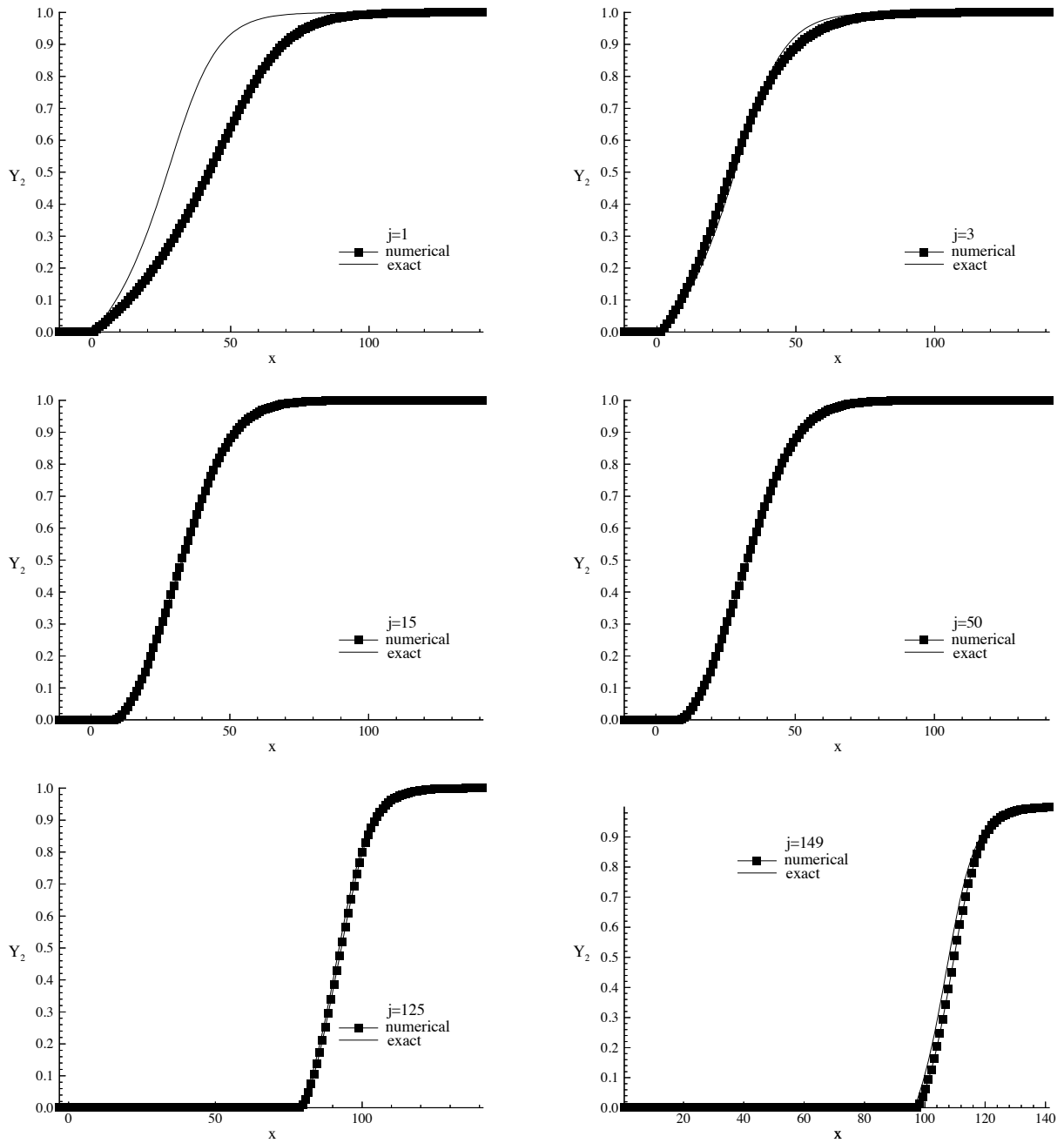


Figure 7.15: Comparison of numerical and exact product mass fraction solutions along $j = \text{constant}$ grid lines ($f_n = 2.0$, 149 by 149 grid, $t = 1869.6$).

7.2 Normal Overdrive $f_n = 1.6$

Next, the case of $f_n = 1.6$ was considered; this was expected to be unstable based upon the one-dimensional results. This level of overdrive was achieved with $\mathcal{M}_0 = 9.0$, and the detonation and initial wedge angles were $\beta = 60.9^\circ$ and $\theta_{max} = 50.2^\circ$, respectively.

The first grid used had 99 by 99 cell centers, approximately 14 points per $L_{1/2}$, and $L_{ratio} = 14.6$. The increased number of points in $L_{1/2}$ compared to the previous case is a result of the thickening of the reaction zone as f_n decreases. For this grid the solution relaxed to a steady state; however, this final solution differed from the exact steady solution. Figure 7.16 shows shaded contours of density and pressure for the steady numerical solution. It is apparent, particularly in the pressure solution, that the contours are no longer straight as in the exact solution. In the progression of the solution to steady state, it appeared a disturbance was beginning to form and progress upward along the detonation front. As it approached the top of the domain, however, the formation of the disturbance slowed, eventually stopping at the steady state shown. This appears in the figure as slight bulges of high density and pressure.

Since the upper boundary seemed to be affecting the progress of the solution, a second case was run in which the grid was extended in the y direction such that $L_{ratio} = 18.1$. Approximately the same grid resolution was maintained by increasing the number of points to 124 in this direction (approximately 14 points per $L_{1/2}$). In this case an unsteady solution somewhat similar to the initial unsteady solution at $f_n = 2.0$ was found. In order to determine if it would withstand grid refinement, a third case with a grid resolution of 124 by 149 (16 points per $L_{1/2}$) was considered. As happened with the initial grid, the solution froze at a state different than the exact steady solution. Once again the top boundary seemed to quell the formation of the initial disturbance.

A final fourth case was run with the upper boundary extended even farther in the y direction while maintaining approximately the same resolution of the previous case

(139 by 184 cell centers, 15 points per $L_{1/2}$, and $L_{ratio} = 22.4$). The unsteady solution returned once again. Figures 7.17, 7.18, and 7.19 show the time-dependent solutions for ρ , p , and Y_2 . The initial formation of a disturbance is apparent in the first frame. By the second frame the disturbance has become triangular in shape, and is a moving region of very high density and pressure. Above this region the shock becomes very strong and begins to bow outwards. The increased shock strength leads to a region of nearly complete reaction immediately behind it. The disturbance progresses upward along the detonation front until it has left the domain through the top boundary.

At this point the upper half of the detonation front begins to move forward into the freestream. A second triangular disturbance then forms about the midpoint of the detonation, and progresses downward along the front (frame 4). Comparing the density and pressure contours, it is apparent that the bottom side of the triangular region is a shock, but the top portion is a contact discontinuity. Courant and Friedrichs [22] refer to this as a three-shock configuration, which is diagrammed in Fig. 7.20. Two intersecting shock fronts S_1 and S_2 result in a reflected shock S' and a contact discontinuity D_c . The reflected shock is necessary to turn the flow following S_1 such that it is parallel to the flow following S_2 ; the contact discontinuity separates the two parallel flows V_1' and V_2 .

The reflected shock S' emanating from the the three-shock configuration can be seen to intersect the wall and form another reflected shock into the rear of the domain. The mass fraction contours show that a “tongue” of complete reaction has progressed downward with the three-shock. By frames five and six the three-shock reaches the wall, resulting in even higher densities and pressures. The following detonation front is highly curved and moving forward into the freestream. Small instabilities are apparent at the front in the product mass fraction contours. Soon after the last frame shown, the detonation reaches the forward computational boundary and attempts to propagate out of the domain. As before, the simulation was stopped at this point.

The forward propagation of the unstable detonation is expected from the one-dimensional results; as the detonation becomes unstable, the detonation velocity changes a great deal. For an attached oblique detonation such as this, the detonation velocity of the exact steady solution is equal to the component of the freestream velocity normal to the wave front. Thus, since the freestream velocity remains fixed, the detonation propagates forward and out of the computational domain as the detonation velocity increases due to the instability. It is hypothesized that the forward propagation of the detonation would eventually end as it was weakened by following rarefactions. An attempt was not made to simulate this due to the prohibitively large number of grid points that would be required to extend the computational domain a large distance in the $-x$ direction.

In order to somewhat quantify the effect of moving the outflow boundary at the top of the domain, the two unsteady density solutions obtained (the second and fourth cases considered) were plotted side-by-side in Fig. 7.21. These solutions should be similar, since they have about the same number of points in the half reaction zone. Three frames from each solution are shown with overlaid Cartesian coordinate grid lines; the three frames on the left side of the plot are from the solution on the 99 by 124 grid, while the frames on the right are from the solution on the 139 by 184 grid. The divisions along each coordinate axis were made the same for each case to ease comparison. The frames were chosen so that the three-shock/disturbance structures were evident and the elapsed times of the solutions were approximately the same. Qualitatively, the solutions appear nearly identical; the approximate locations and general appearance of the three-shock structures are very similar. Table 7.1 shows the actual location and magnitude of the density peak in the three-shock structure of each frame. The numbers show the agreement between the two solutions is relatively good. These results would seem to indicate that once the three-shock structure has formed, the location of the top outflow boundary does not have much effect on the

99 by 124 grid			139 by 184 grid		
ρ_{max}	x	y	ρ_{max}	x	y
10.692	204.7	339.9	10.400	202.3	341.9
10.366	155.9	300.8	10.562	159.8	304.5
11.319	82.4	144.6	11.408	84.9	147.9

Table 7.1: Magnitude and location of density peak in three-shock structure of numerical solutions on two different grids ($f_n = 1.6$).

ensuing solution.

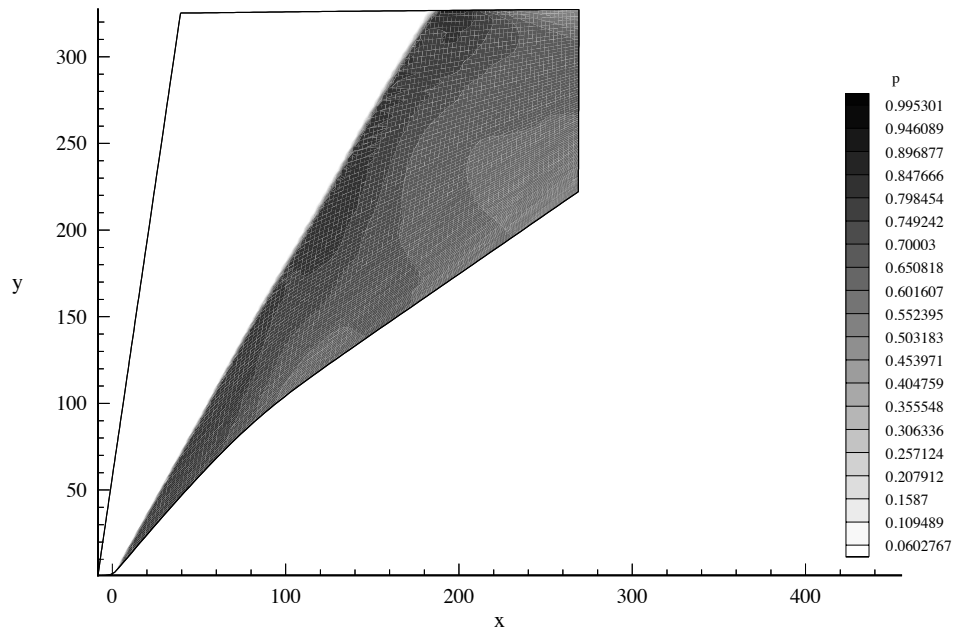
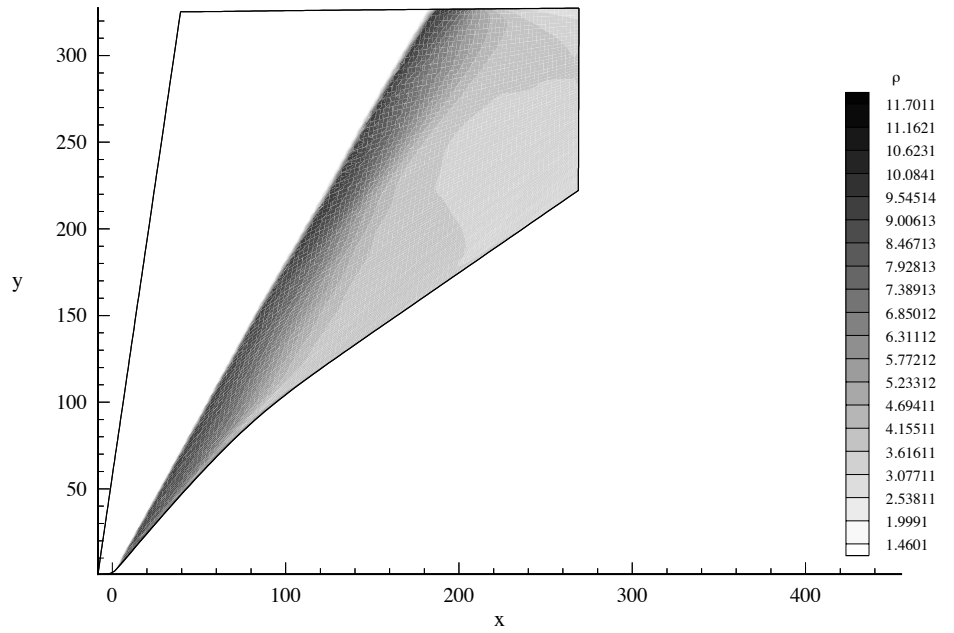


Figure 7.16: Shaded density and pressures contours for steady numerical solution ($f_n = 1.6$, 99 by 99 grid, $t = 3128.2$).

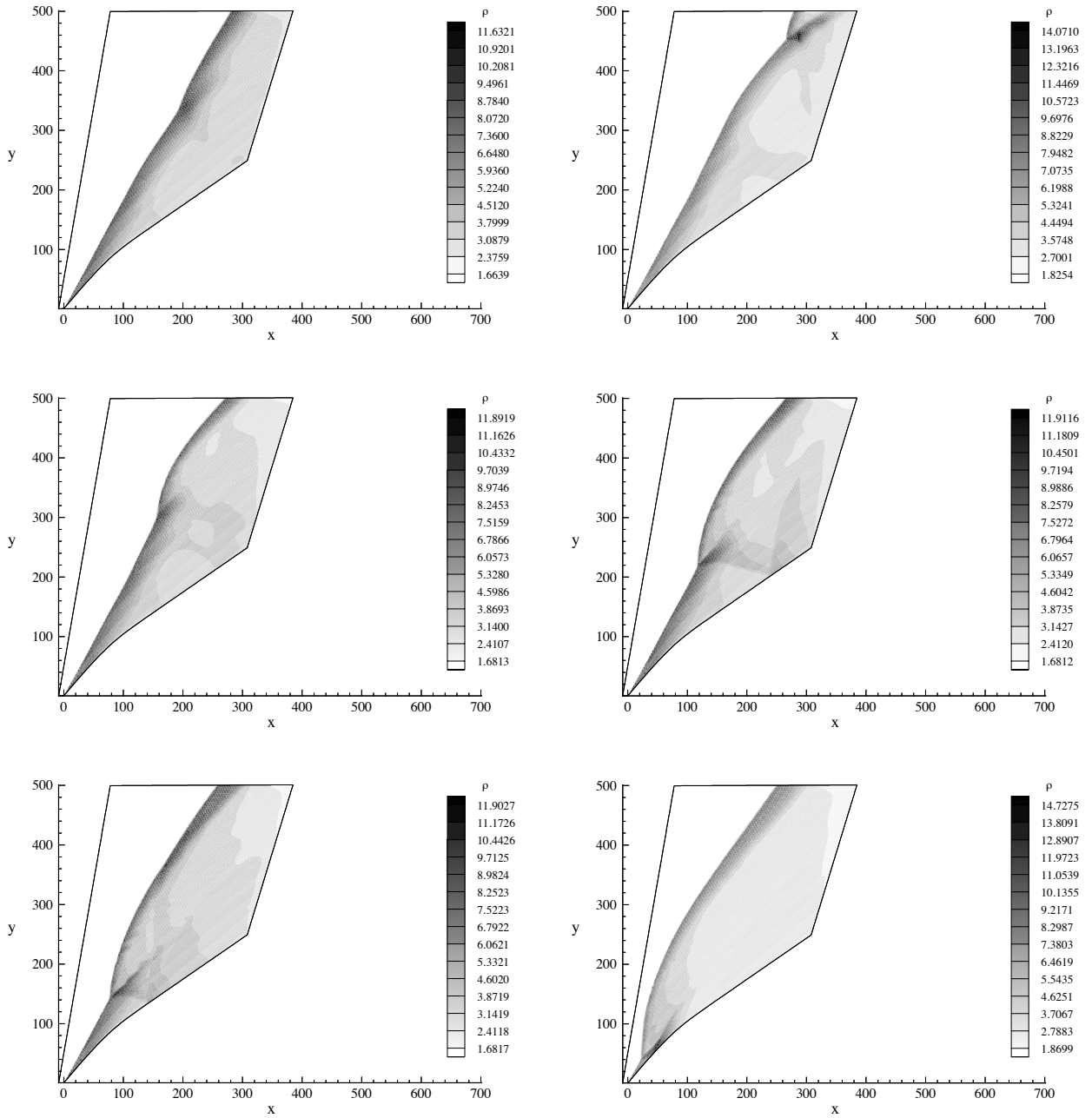


Figure 7.17: Shaded density contours as a function of time ($f_n = 1.6$, 139 by 184 grid, $t = 1176.3, 1929.1, 2776.0, 3481.7, 4234.5, 5175.5$).

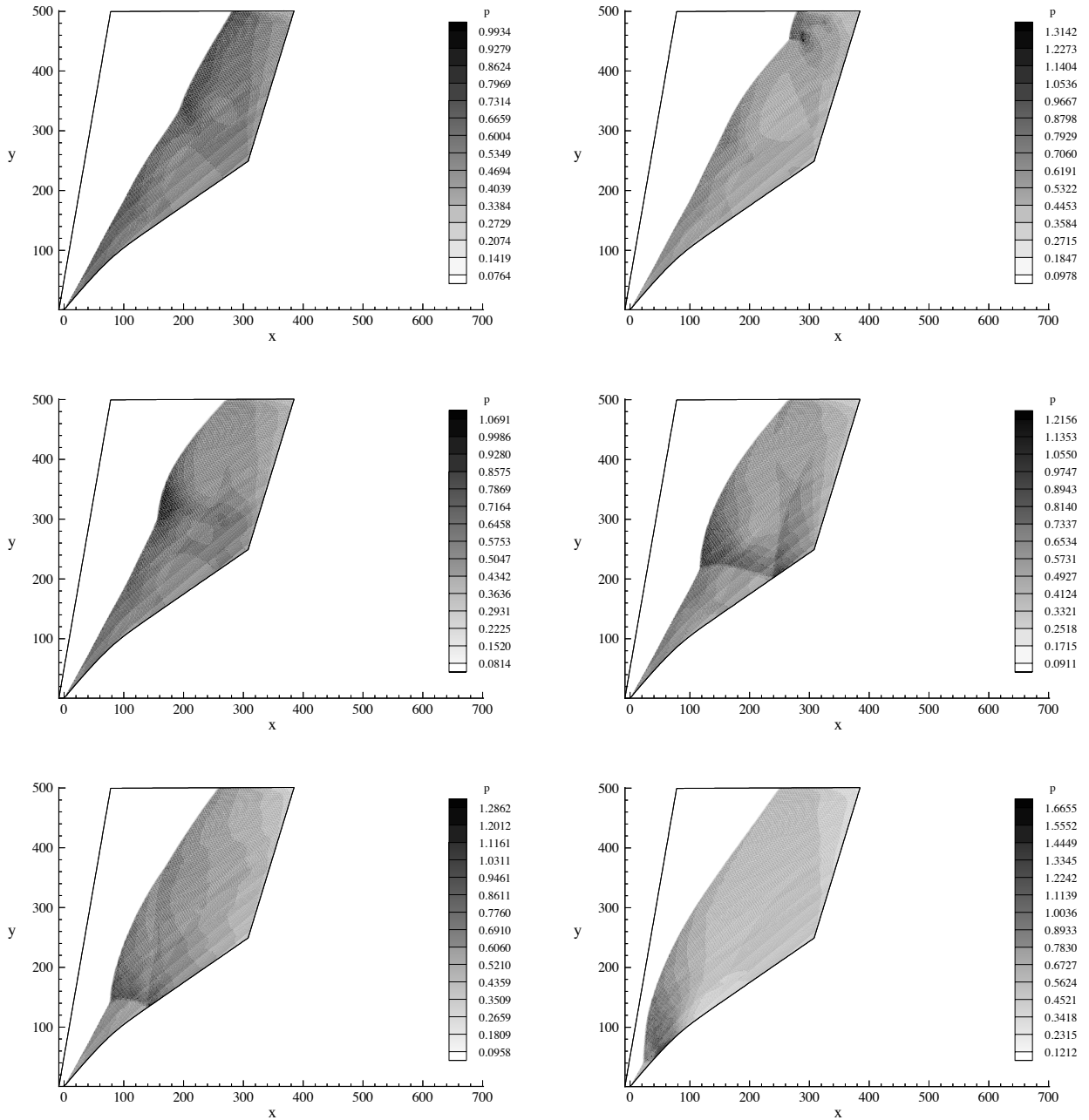


Figure 7.18: Shaded pressure contours as a function of time ($f_n = 1.6$, 139 by 184 grid, $t = 1176.3, 1929.1, 2776.0, 3481.7, 4234.5, 5175.5$).

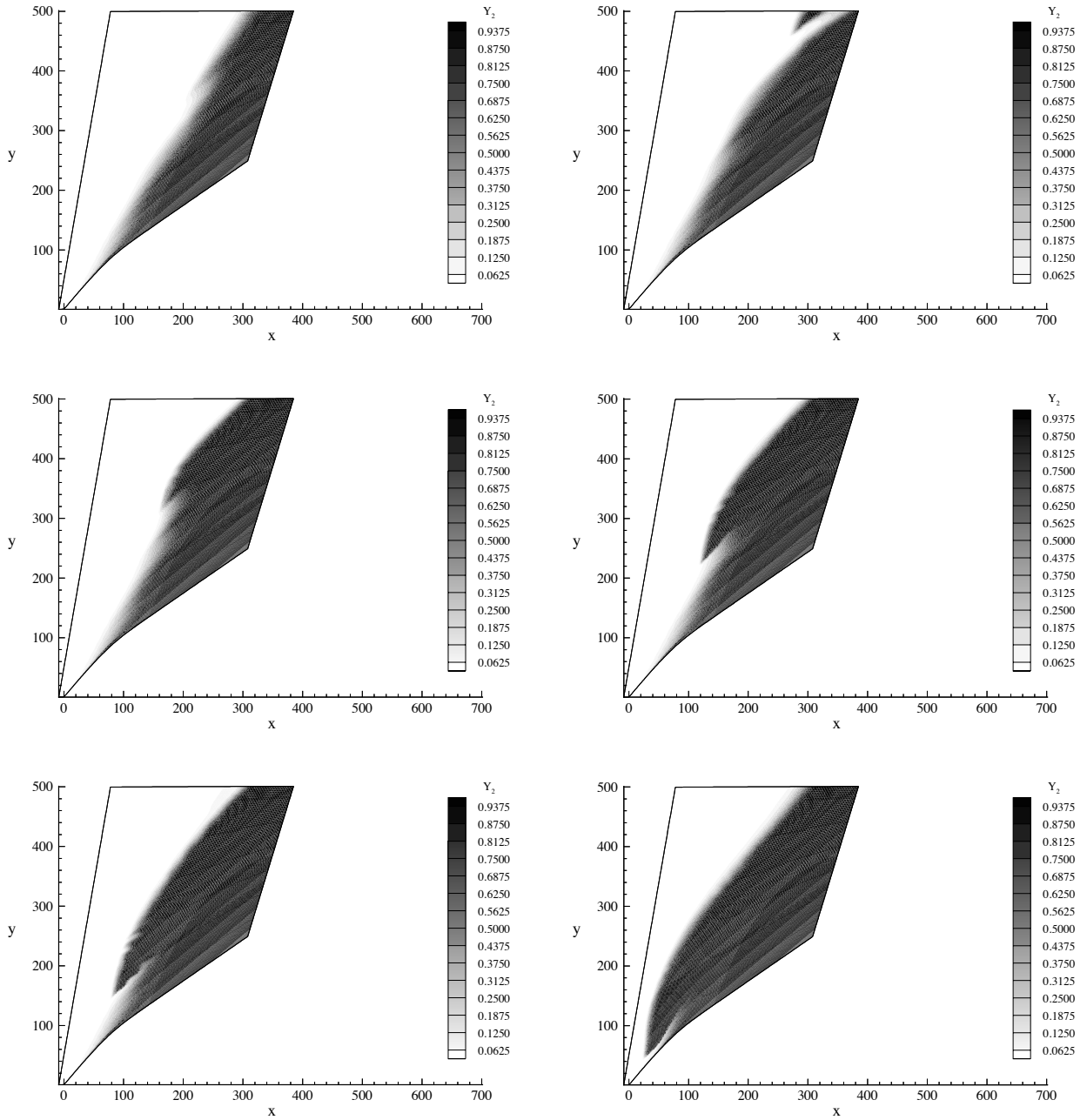


Figure 7.19: Shaded product mass fraction contours as a function of time ($f_n = 1.6$, 139 by 184 grid, $t = 1176.3, 1929.1, 2776.0, 3481.7, 4234.5, 5175.5$).

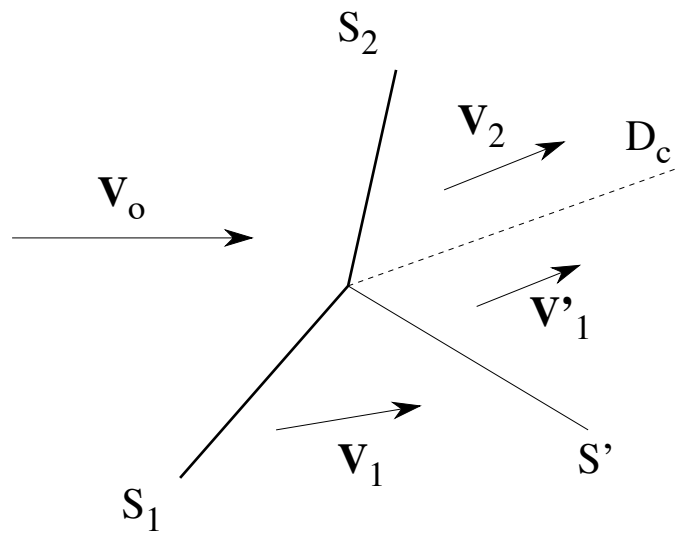


Figure 7.20: Illustration of three-shock structure [22].

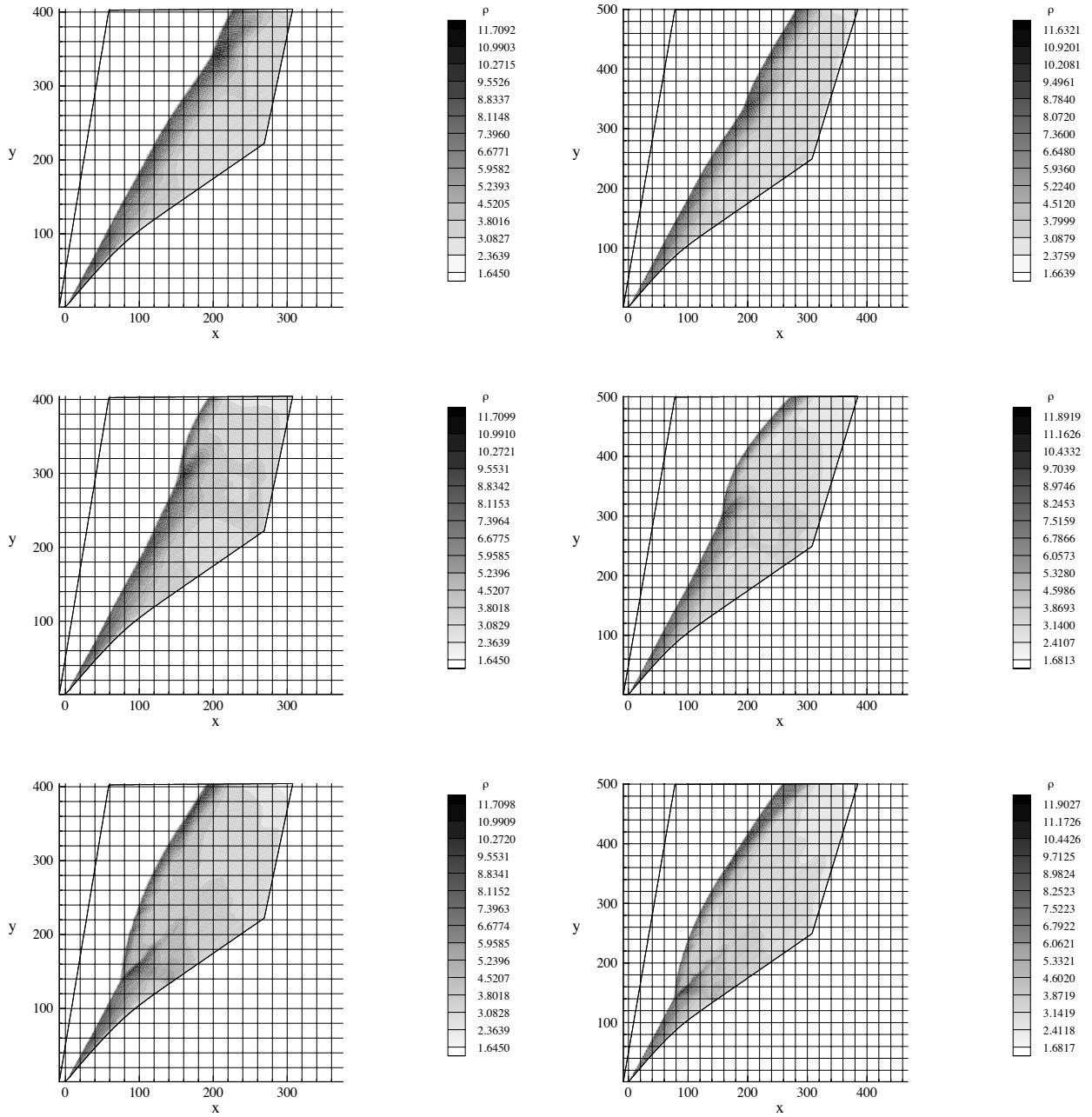


Figure 7.21: Comparison of progression of density solutions on different grids (left - 99 by 124 grid, right - 139 by 184 grid).

7.3 Normal Overdrive $f_n = 1.7$

Having demonstrated cases of stable and unstable oblique detonations, the next step was to more accurately determine the location of the stability boundary. For a one-dimensional detonation an overdrive value of 1.7 is just on the unstable side of the stability boundary, so the equivalent case for the oblique detonation was considered next. This level of overdrive was achieved at $\mathcal{M}_0 = 9.2115$, and the detonation and initial wedge angles were $\beta = 61.6^\circ$ and $\theta_{max} = 50.7^\circ$, respectively.

The initial run on a 124 by 149 grid with approximately 13 points per $L_{1/2}$ and $L_{ratio} = 21.7$ resulted in an unstable solution similar to those discussed. Increasing the grid resolution to 149 by 149 and 15 points per $L_{1/2}$ resulted in the solution stalling at an incorrect steady solution, as described in the previous section. Once again, moving the top outflow boundary upward while maintaining the grid resolution (159 by 184 grid, 15 points per $L_{1/2}$, and $L_{ratio} = 27.1$) resulted in the instability reoccurring. Figures 7.22, 7.23, and 7.24 show ρ , p , and Y_2 for the unsteady solution obtained. In this case a disturbance has begun to form in frame 1, and by frame 2 it has become a triple-shock and moved toward the upper boundary. Instead of exiting through the top of the domain, however, it reverses direction and begins moving downward along the detonation front (this was also the behavior observed on the 124 by 149 grid). At this point an entirely new behavior occurred. As shown in frames 3 and 4, small “bursts” of completely reacted material have jumped forward into the freestream. The densities and pressures in this region are lower than the surrounding areas. Apparently a small region at the detonation front became very unstable quickly, resulting in a localized jump in the detonation velocity and the bursting phenomena observed. By the later frames the bursting has completely disappeared, and the solution follows the same pattern observed before, *i.e.* the three-shock hits the wall and the detonation front moves into the inflow boundary.

Figures 7.25, 7.26, and 7.27 show magnified time series views of the bursting events

for ρ , p , and Y_2 . The initial bursting begins slightly above the triple-shock in frame 1, and frames 2 and 3 show the growth and decay of the two events. By frame 4 the first two bursts have disappeared and two new ones have occurred slightly further above the triple-shock. These also grow and decay, and by the last frame the only structure remaining is the triple-shock. Figure 7.28 shows two magnified views of the bursting regions in frames 2 and 5. These plots combine velocity vectors with shaded density contours. The velocity vectors show an interesting result; within the bursts the flow direction has reversed, and recirculation regions have developed. The reversal of the flow direction can be explained once again by the increased unstable detonation velocity. For a propagating normal shock or detonation, the passage of the wave through a quiescent medium induces a velocity in the medium in the direction of propagation. For the high-speed bursting, the induced velocity to the left due to the burst is greater than the shocked freestream velocity to the right. Thus, the net effect is that the flow reverses direction immediately behind the burst. Apparently, the flow reversal combined with the large generation of vorticity due to the highly curved bursting detonation front result in the recirculation zones. It should be noted that no bursting events occurred in the unsteady solution on the 124 by 149 grid.

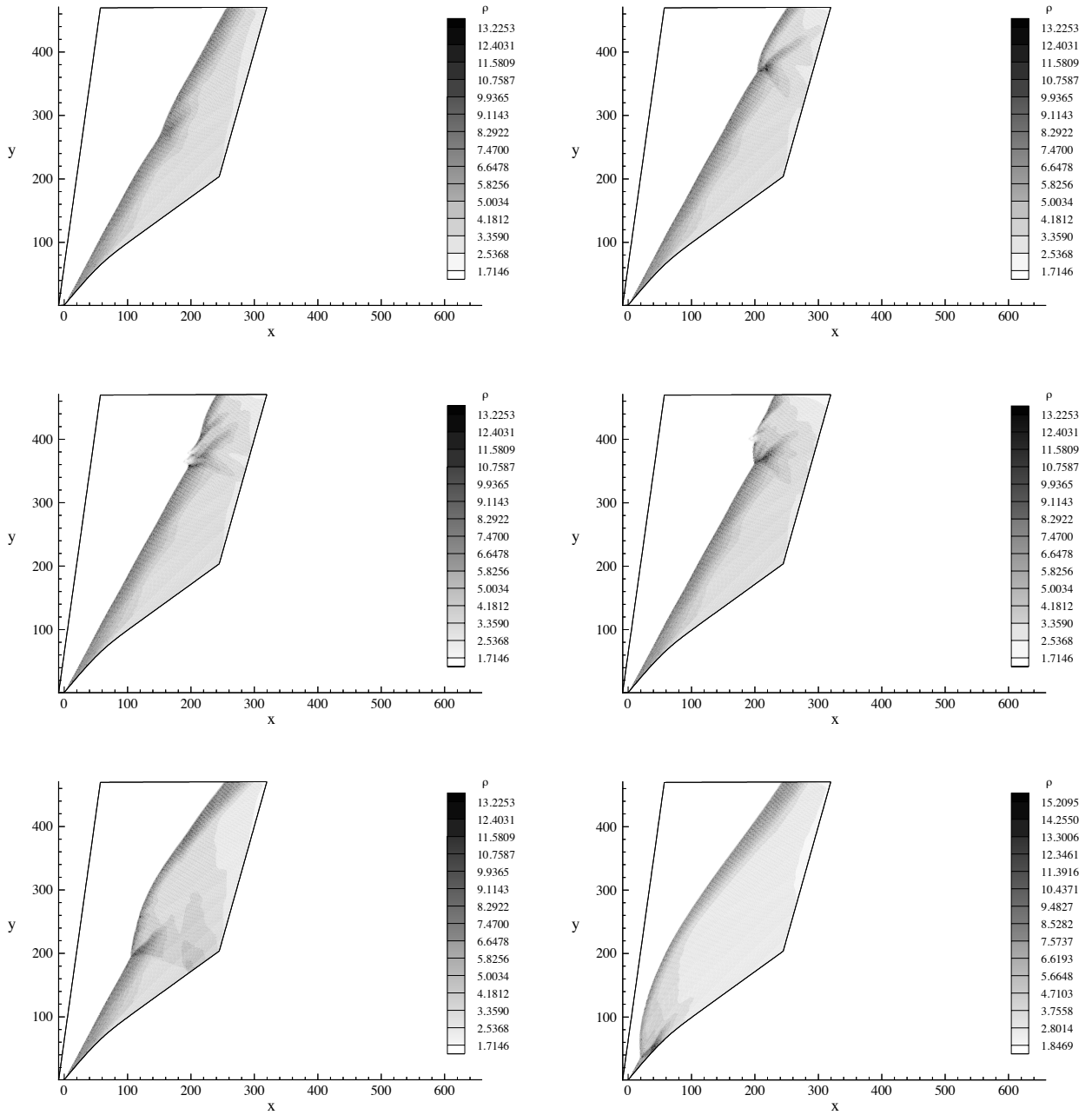


Figure 7.22: Shaded density contours as a function of time ($f_n = 1.7, 159$ by 184 grid, $t = 960.5, 1870.6, 2831.1, 3791.7, 7077.8, 9251.7$).

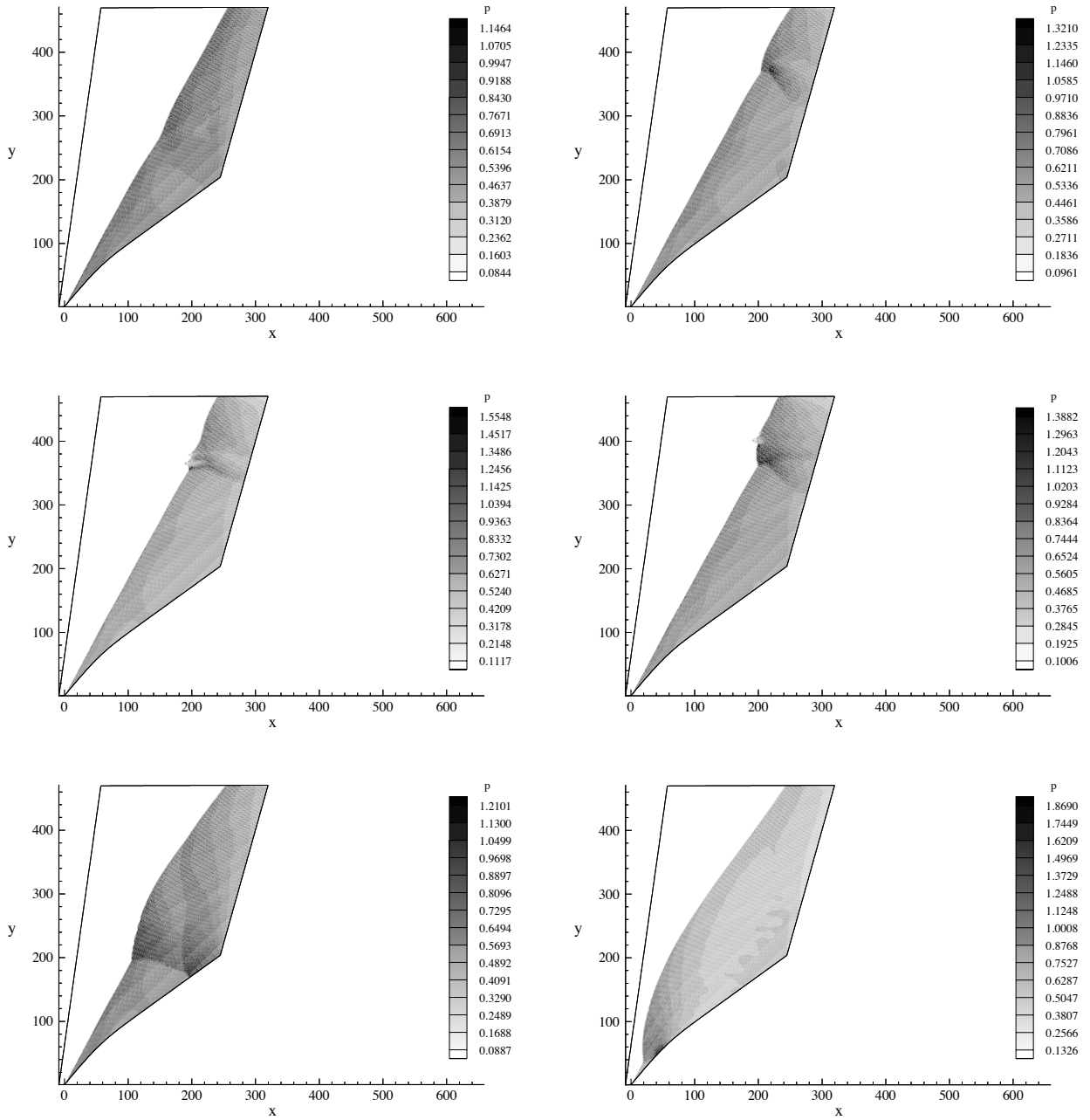


Figure 7.23: Shaded pressure contours as a function of time ($f_n = 1.7$, 159 by 184 grid, $t = 960.5, 1870.6, 2831.1, 3791.7, 7077.8, 9251.7$).

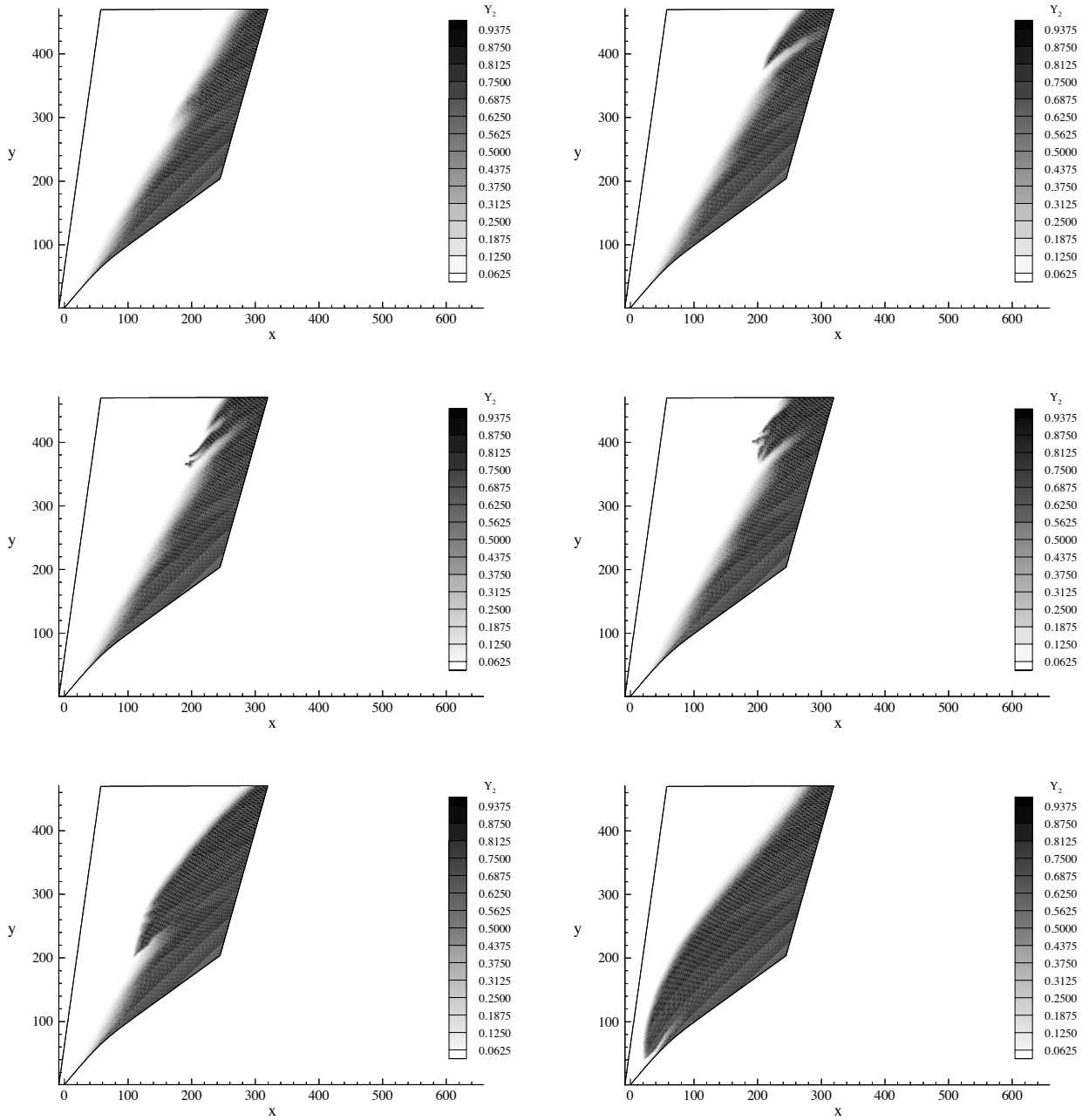


Figure 7.24: Shaded product mass fraction contours as a function of time ($f_n = 1.7$, 159 by 184 grid, $t = 960.5, 1870.6, 2831.1, 3791.7, 7077.8, 9251.7$).

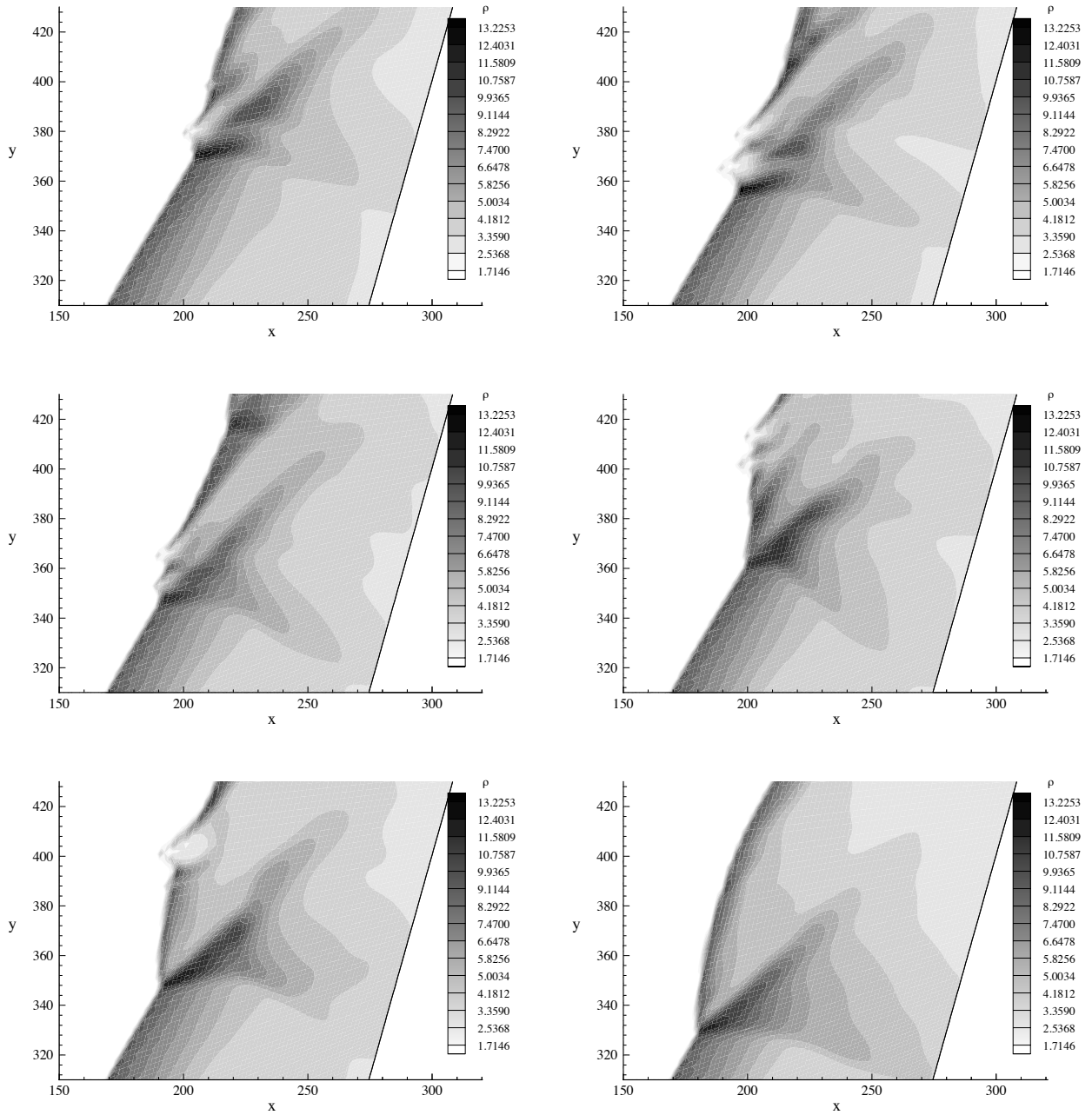


Figure 7.25: Shaded density contours as a function of time for bursting events ($f_n = 1.7, 159$ by 184 grid, $t = 2527.8, 2831.1, 3336.7, 3741.1, 3892.8, 4145.6$).

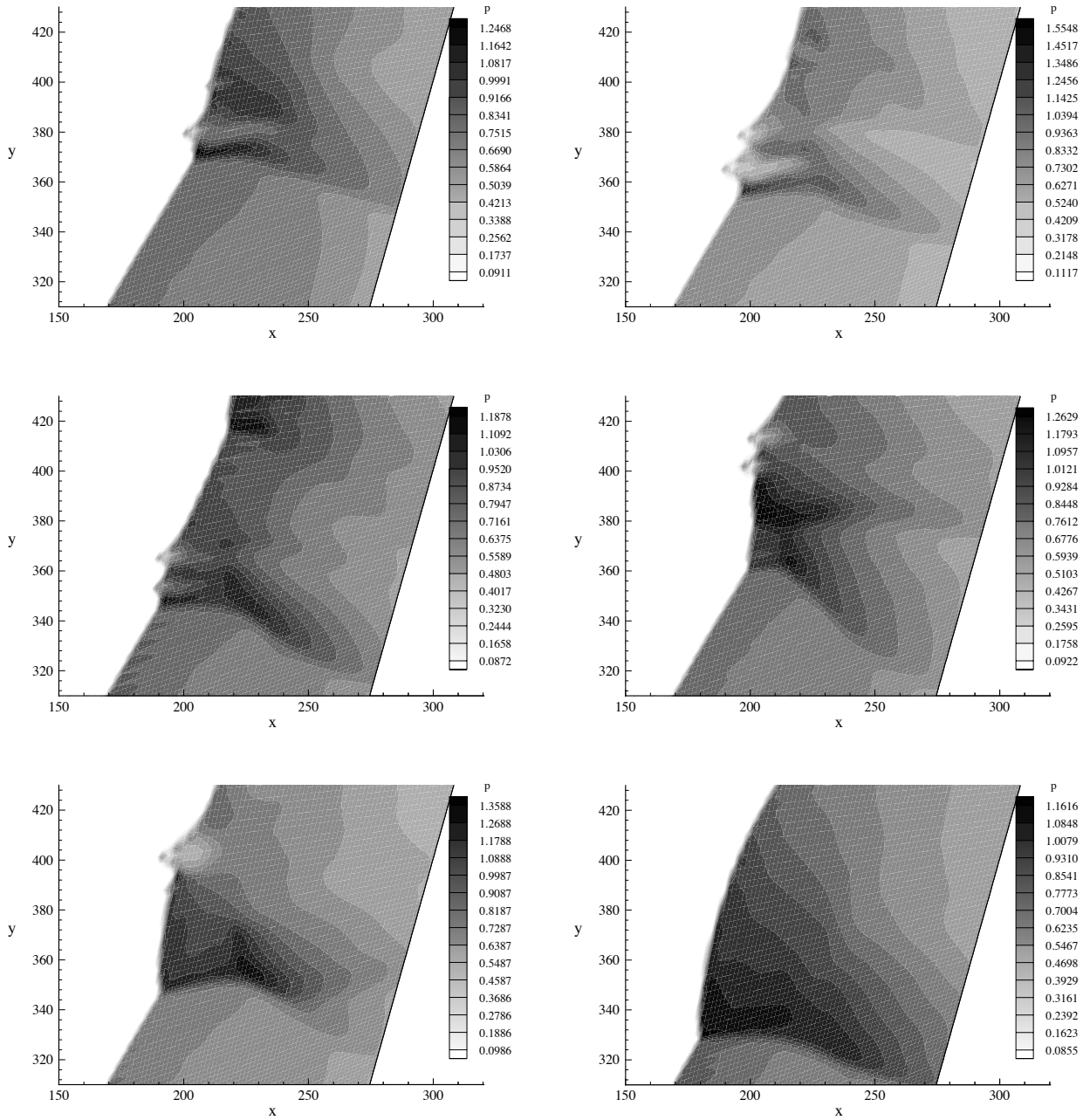


Figure 7.26: Shaded pressure contours as a function of time for bursting events ($f_n = 1.7, 159$ by 184 grid, $t = 2527.8, 2831.1, 3336.7, 3741.1, 3892.8, 4145.6$).

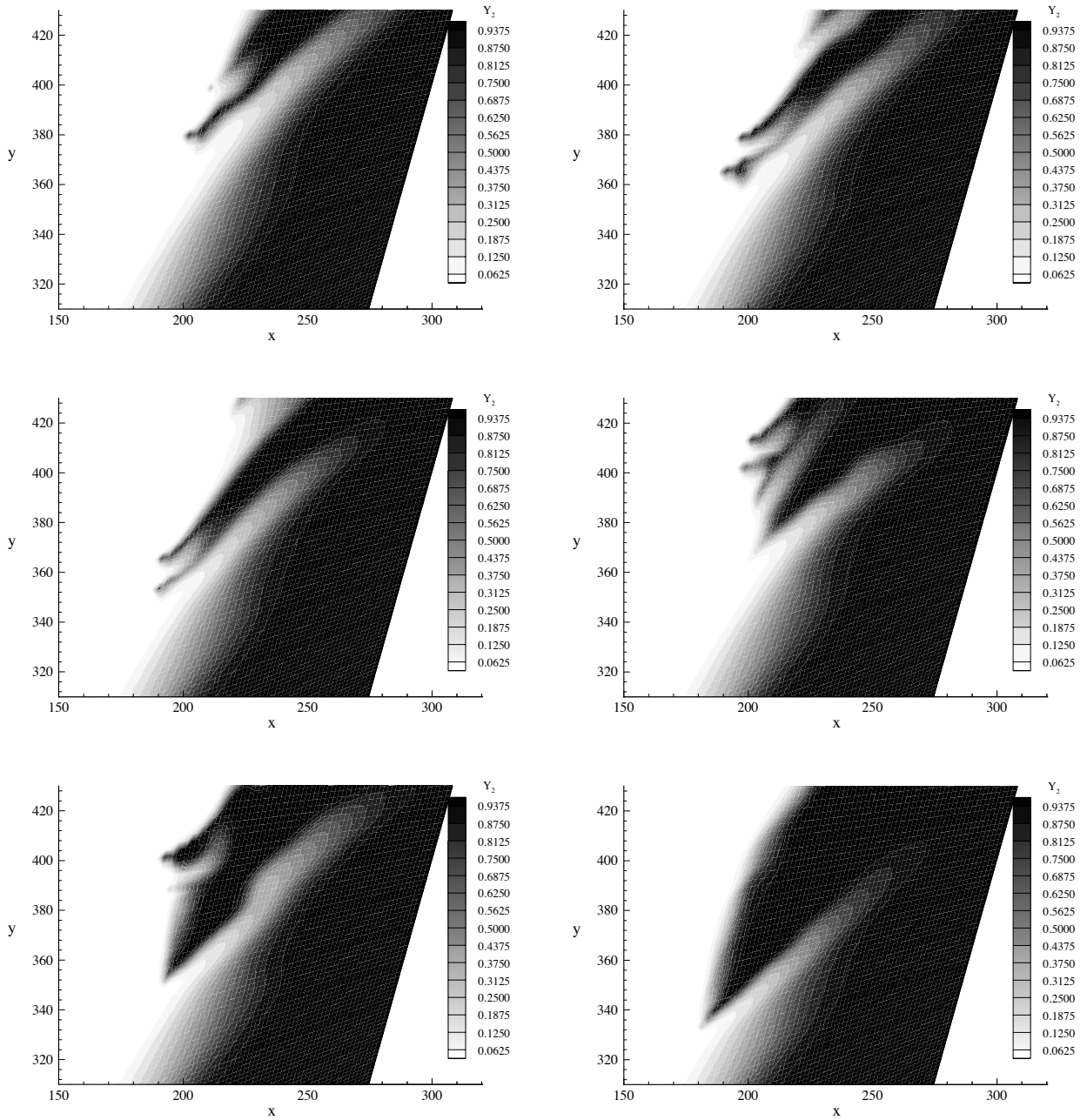


Figure 7.27: Shaded product mass fraction contours as a function of time for bursting events ($f_n = 1.7$, 159 by 184 grid, $t = 2527.8, 2831.1, 3336.7, 3741.1, 3892.8, 4145.6$).

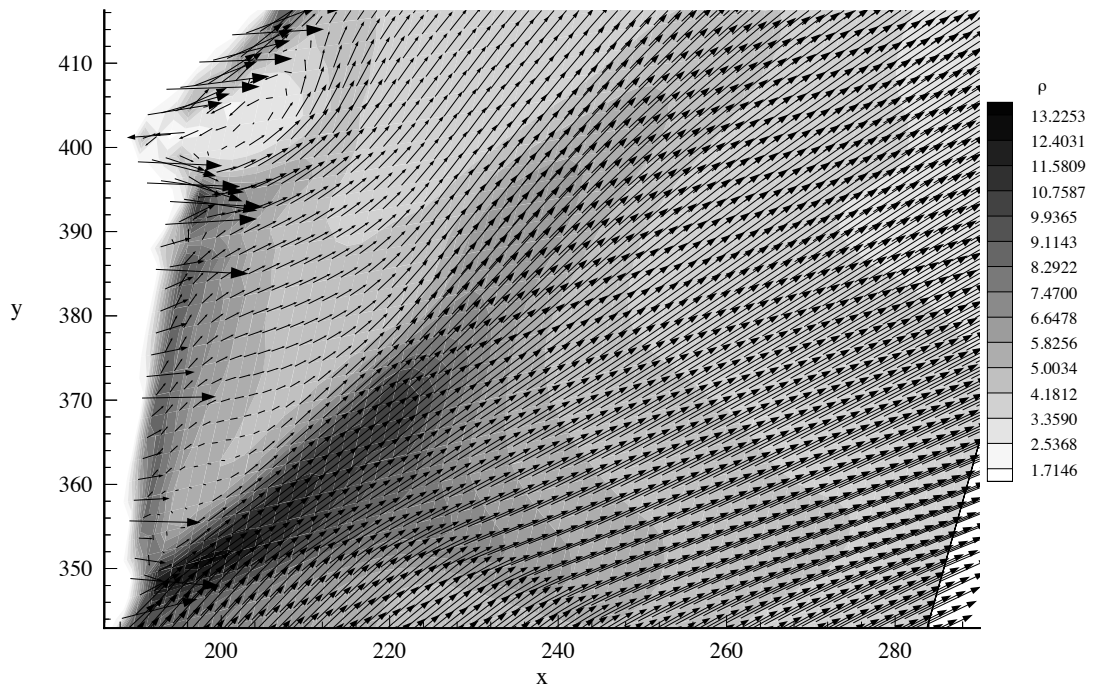
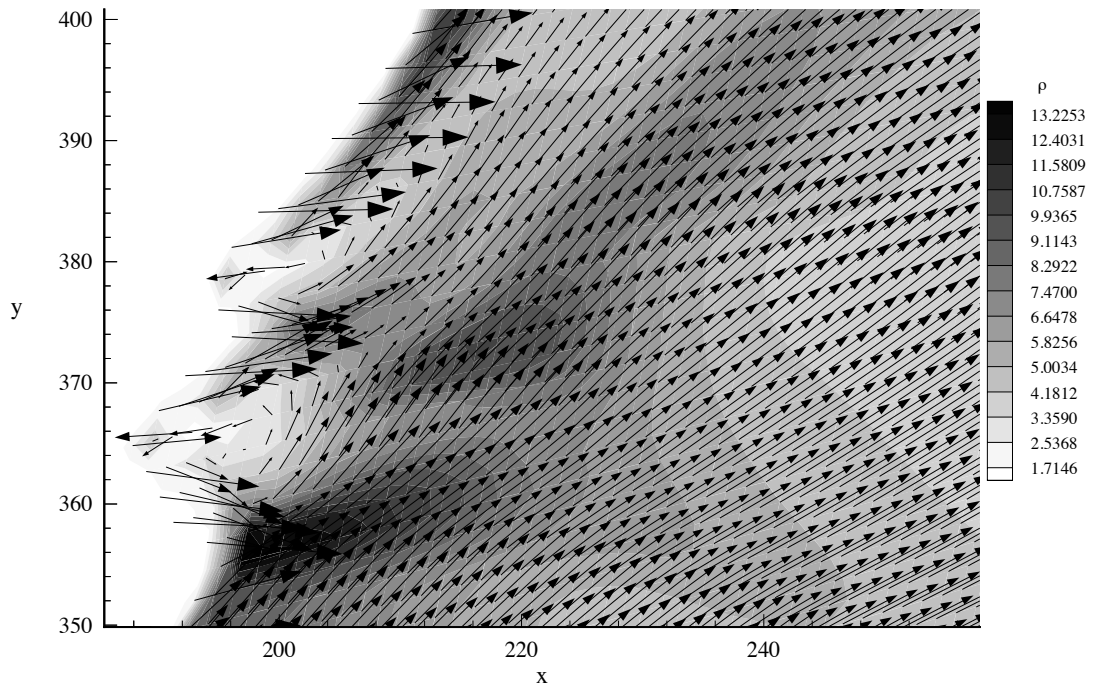


Figure 7.28: Shaded density contours and velocity vectors within the bursting regions ($f_n = 1.7$, 159 by 184 grid, $t = 2831.1$ and 3892.8).

7.4 Normal Overdrive $f_n = 1.76$

For a one-dimensional detonation, $f = 1.76$ is slightly on the stable side of the stability boundary. For the oblique detonation, however, $f_n = 1.76$ led to an unstable solution. For this case the freestream Mach number, detonation angle, and initial wedge angle were $\mathcal{M}_0 = 9.3363$, $\beta = 62.0^\circ$, and $\theta_{max} = 50.9^\circ$.

Five different cases were run to verify the unsteady nature of the detonation at this level of overdrive. The first grid had 124 by 149 cell centers (approximately 12 points per $L_{1/2}$ and $L_{ratio} = 24.9$) and resulted in a typical unstable solution. The grid resolution was increased to 149 by 149 (approximately 13 points per $L_{1/2}$), which led to a stalled, incorrect steady solution. Moving the top boundary and maintaining the resolution (159 by 184 grid, approximately 13 points per $L_{1/2}$, and $L_{ratio} = 31.2$) restored the instability once again. As a further check the grid resolution was increased again to 184 by 209 (approximately 15 points per $L_{1/2}$), which still resulted in an unstable solution. Finally, to ensure that the slight artifacts apparent in the one-dimensional detonation verification studies were not influencing the solution, the simulation was rerun on this final grid using a CFL number of 0.35. This also led to an unstable solution.

Figures 7.29, 7.30, and 7.31 show the unsteady solutions for density, pressure, and product mass fraction obtained on the 184 by 209 grid with CFL = 0.35. As with the previous unsteady solutions, the disturbance begins to develop in the first frame and has transitioned to a three-shock structure by the second frame. In this case, however, a second three-shock began to develop near the top of the domain, but slowly dissipated as the first three-shock structure began to move upwards. The rate at which the three-shock moved upward along the detonation front was much slower in this case than in unsteady results of the previous sections. By frame three its upward progress has stopped, and it very slowly begins to move downward. Though difficult to see in these figures, the remnants of the second structure follow the remaining

CFL = 0.4			CFL = 0.35		
ρ_{max}	x	y	ρ_{max}	x	y
11.785	170.0	277.5	12.149	169.6	275.8
11.529	174.7	324.7	11.534	177.1	323.6
11.374	58.9	107.6	11.448	53.1	93.4

Table 7.2: Magnitude and location of density peak in three-shock structure of numerical solutions for different time step criteria ($f_n = 1.76$).

three-shock downward. As before, the three-shock eventually hits the wedge and the detonation front moves forward into the inflow boundary.

In order to quantify any differences between the solutions due to the different time step restriction, another side-by-side comparison was made. Figure 7.32 shows the density solutions at different times on the 184 by 209 grid for CFL = 0.35 and 0.4. As before, the elapsed time is approximately the same for each row of the figure. Qualitatively, the first two frames of each solution appear nearly identical; however, it is apparent at the last time that the three-shock structure is closer to the wedge surface for the case of CFL = 0.35 than it is for CFL = 0.4. Table 7.2 lists the locations and magnitudes of the density peak within the three-shock for both solutions. The peak densities agree relatively well for all three times, with the worst agreement occurring at the first time. The peak density locations agree well for the first two times, but not well for the last time, as suggested by the qualitative assessment. For instabilities such as this, slight differences in the initial development of the solution can lead to relatively large differences later on. This is apparently the case here. Overall, both solutions are very similar and indicate instability at this level of normal overdrive.

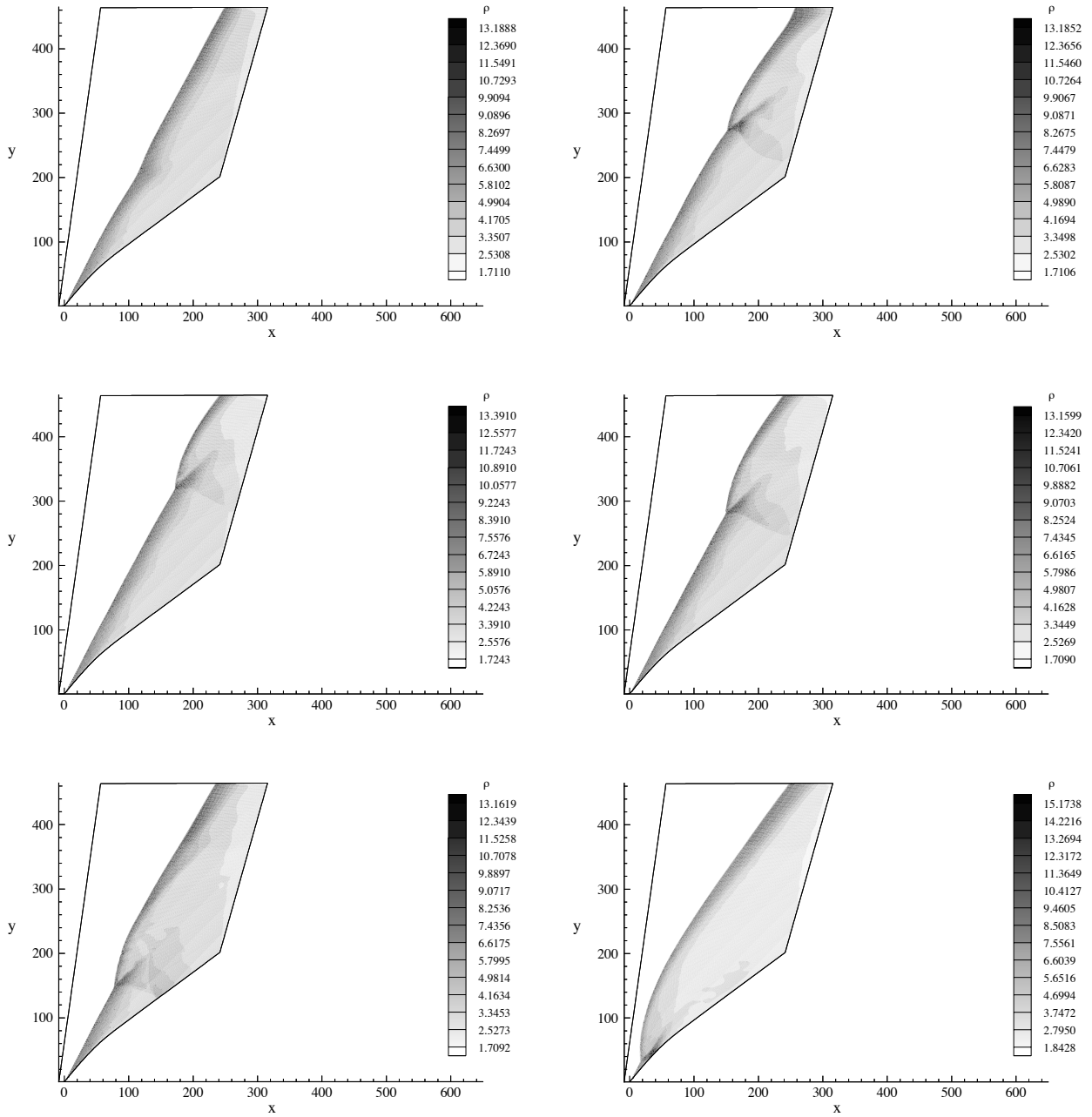


Figure 7.29: Shaded density contours as a function of time ($f_n = 1.76$, CFL = 0.35, 184 by 209 grid, $t = 754.5, 1349.9, 2660.1, 4168.8, 7146.5, 8655.2$).

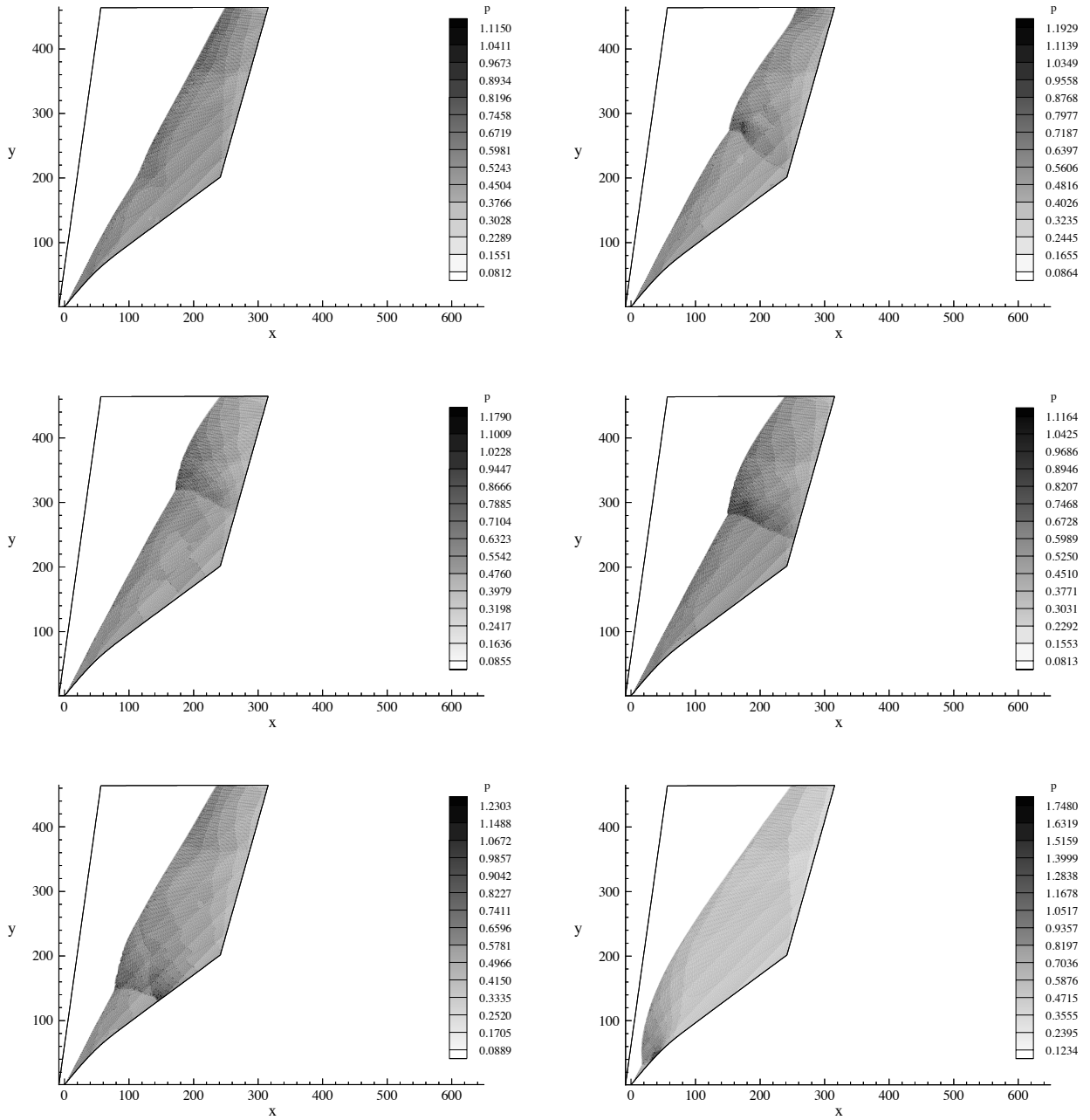


Figure 7.30: Shaded pressure contours as a function of time ($f_n = 1.76$, CFL = 0.35, 184 by 209 grid, $t = 754.5, 1349.9, 2660.1, 4168.8, 7146.5, 8655.2$).

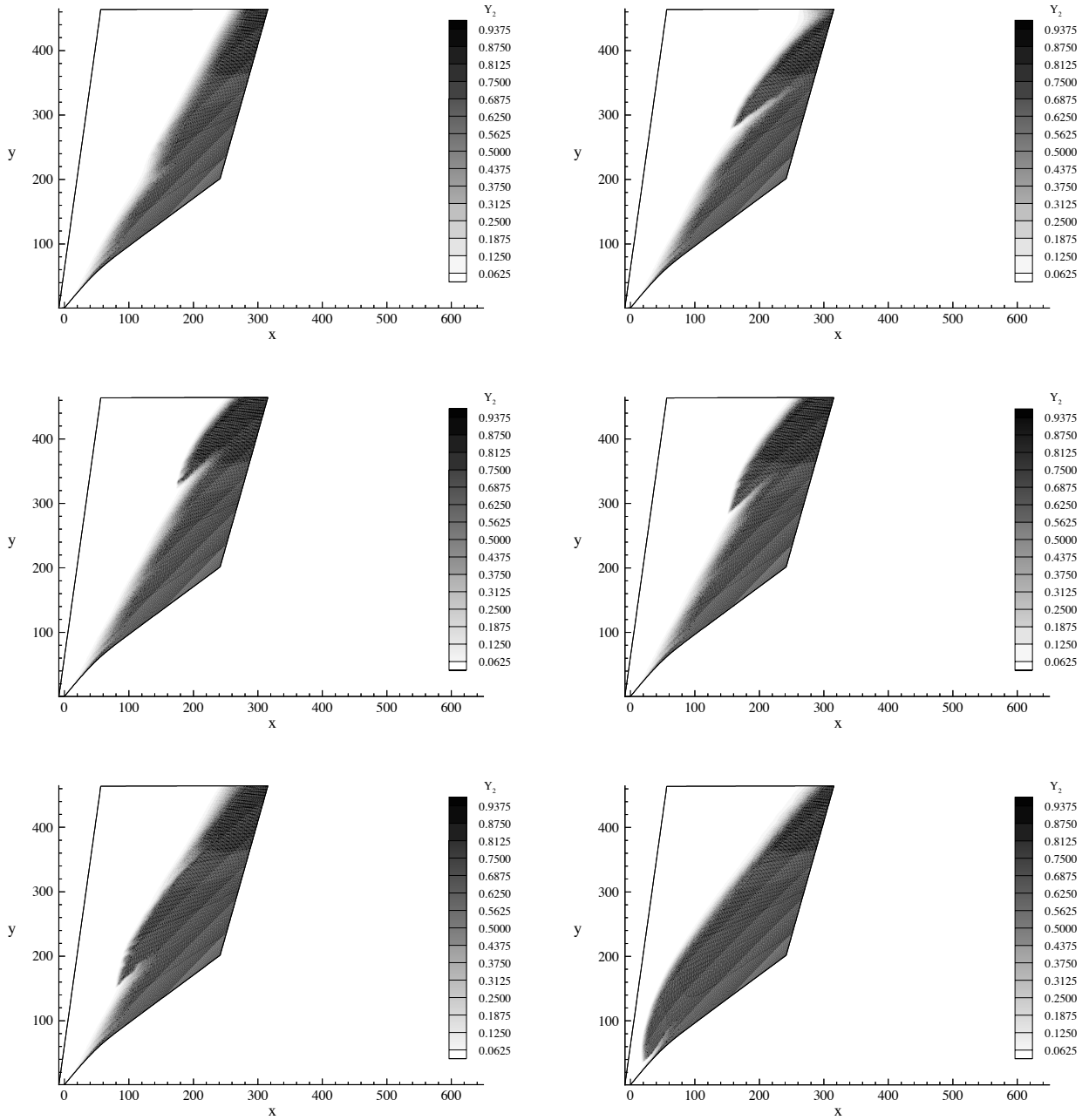


Figure 7.31: Shaded product mass fraction contours as a function of time ($f_n = 1.76$, CFL = 0.35, 184 by 209 grid, $t = 754.5, 1349.9, 2660.1, 4168.8, 7146.5, 8655.2$).

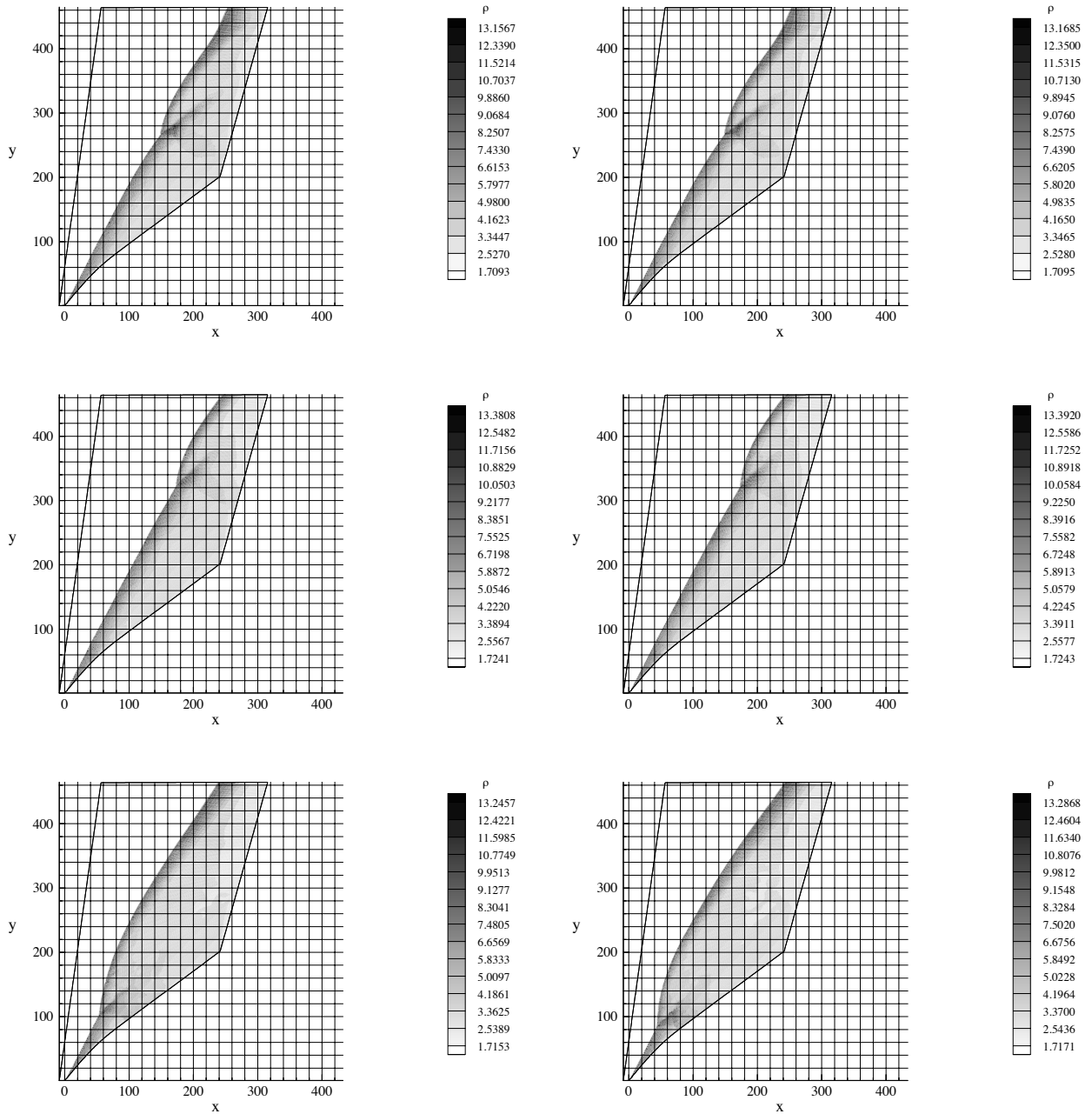


Figure 7.32: Comparison of progression of density solutions on identical grids for different time step criteria (left - CFL = 0.4, right - CFL = 0.35).

7.5 Normal Overdrive $f_n = 1.78$

The stability threshold was finally crossed when $f_n = 1.78$ was considered. The freestream Mach number, detonation angle, and initial wedge angle for this case were $\mathcal{M}_0 = 9.3776$, $\beta = 62.2^\circ$, and $\theta_{max} = 51.0^\circ$, respectively. Figures 7.33, 7.34, and 7.35 show the time series solutions obtained for ρ , p , and Y_2 on a 184 by 209 grid with approximately 16 points per $L_{1/2}$ and $L_{ratio} = 29.8$. This grid was chosen to approximately match the characteristics of the final grid used for $f_n = 1.76$, as there was a relatively high degree of confidence in those results. As with $f_n = 1.76$ a three-shock structure forms and slowly moves upward along the detonation front; a second three-shock also begins to form, but propagates out the top of the domain. Instead of slowing and then reversing direction, however, the first three-shock continues to move slowly upward until it has also left the domain. Following its passage, the detonation front returns to a nearly straight configuration.

Figure 7.36 shows a comparison of the steady numerical and exact solution contours for identical contour values. It is apparent that the agreement between the solutions is not as good as the case of $f_n = 2.0$. In this case the numerical detonation front bows out slightly in the upper half of the domain. There are also extraneous numerical contours in the rear of the domain, indicating the rear boundary condition was not doing as well as before. Figures 7.37, 7.38, and 7.39 show comparisons between the numerical and exact solutions for ρ , p , and Y_2 along various $j = \text{constant}$ gridlines. Along the interior gridlines ($j = 15$ and $j = 50$) the numerical solutions for ρ and Y_2 show good agreement with the exact solutions. The pressure initially shows relatively good agreement, but then oscillations seem to overtake the solution towards the rear of the domain. Along the $j = 1$ and $j = 3$ gridlines the numerical solutions for ρ and Y_2 do not agree very well with the exact solution, while the solution for p is about the same as it was on the interior grid lines. Along the upper gridlines ($j = 125$ and $j = 184$), each of the numerical solutions is offset somewhat in front of the exact

solutions, as was indicated in the contour comparison. Other than the offset, once again the numerical density and product mass fraction solutions look relatively good, while the pressure solution is somewhat lacking. As in the verification studies, the effect of the top outflow boundary condition can be seen particularly at $j = 184$, as well as at the larger values of x on some of the lower $j = \text{constant}$ gridlines. A likely reason for the relatively poor agreement between the numerical and exact solutions is the estimated nature of the boundary conditions at the top and rear of the domain. The formation and passage of the three-shock structure altered the flowfield to such a degree that the boundary conditions were no longer able to maintain the original steady solution.

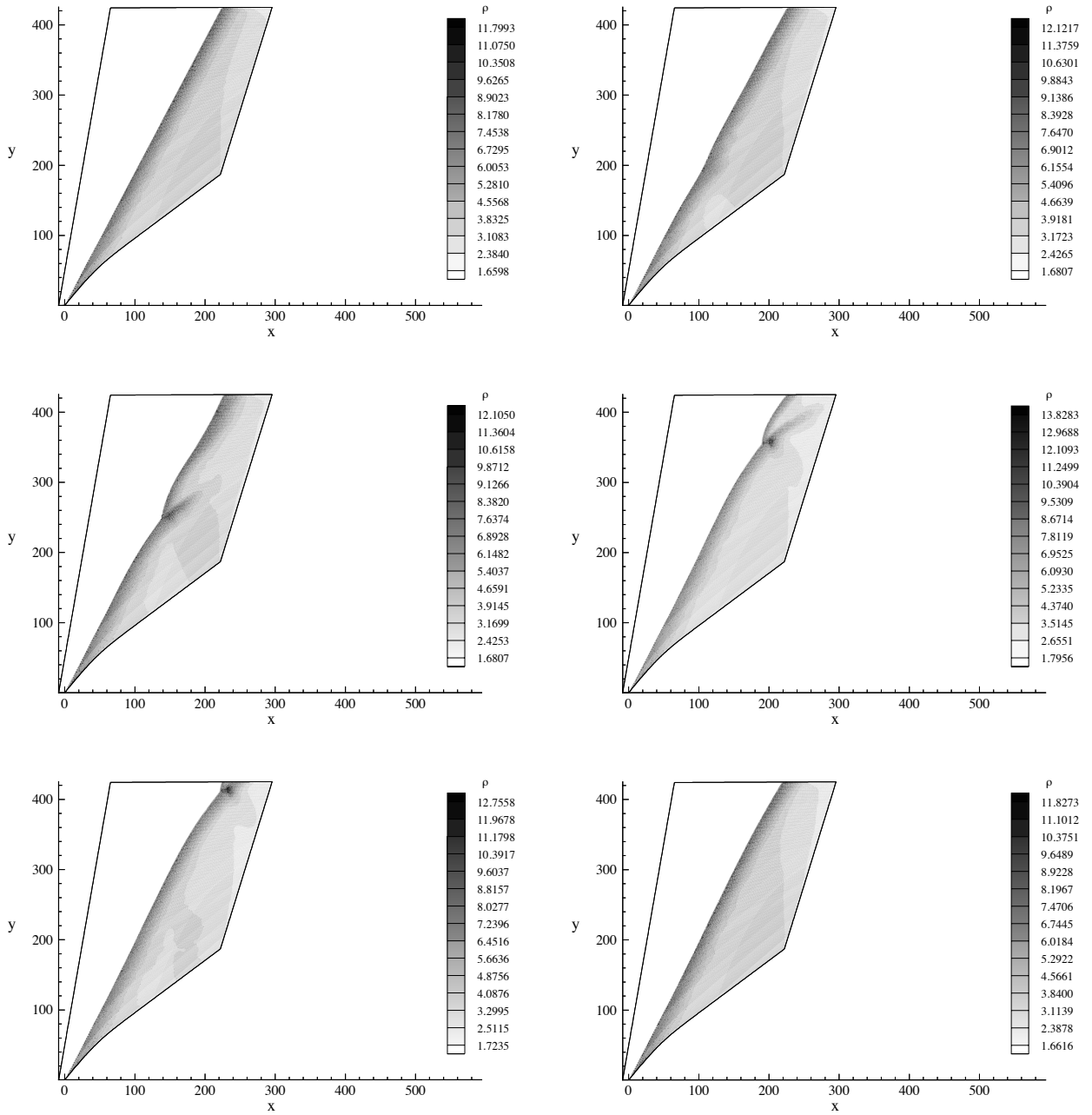


Figure 7.33: Shaded density contours as a function of time ($f_n = 1.78$, 184 by 209 grid, $t = 290.9, 756.3, 1105.4, 2181.7, 2647.2, 3839.9$).

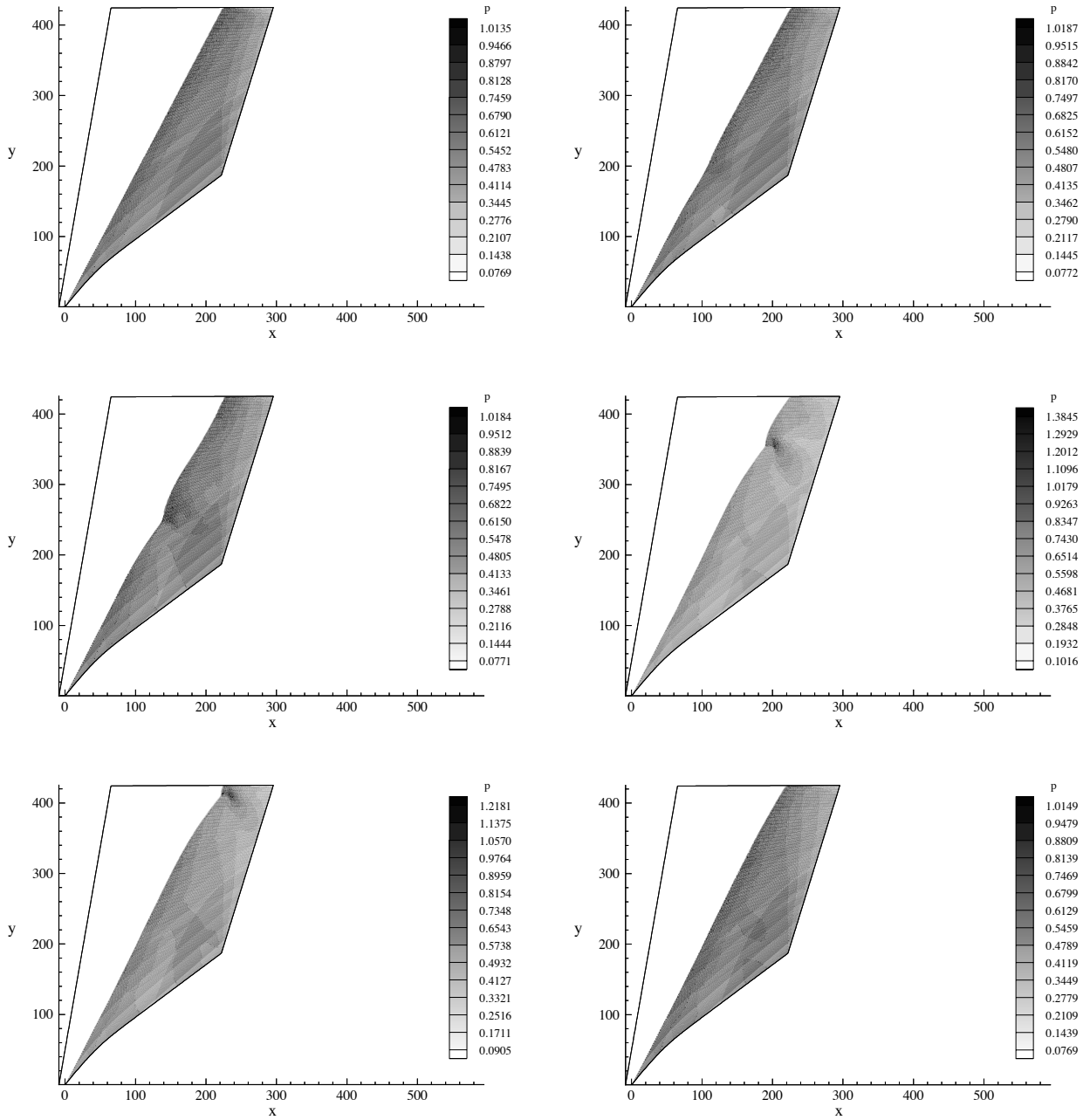


Figure 7.34: Shaded pressure contours as a function of time ($f_n = 1.78$, 184 by 209 grid, $t = 290.9, 756.3, 1105.4, 2181.7, 2647.2, 3839.9$).

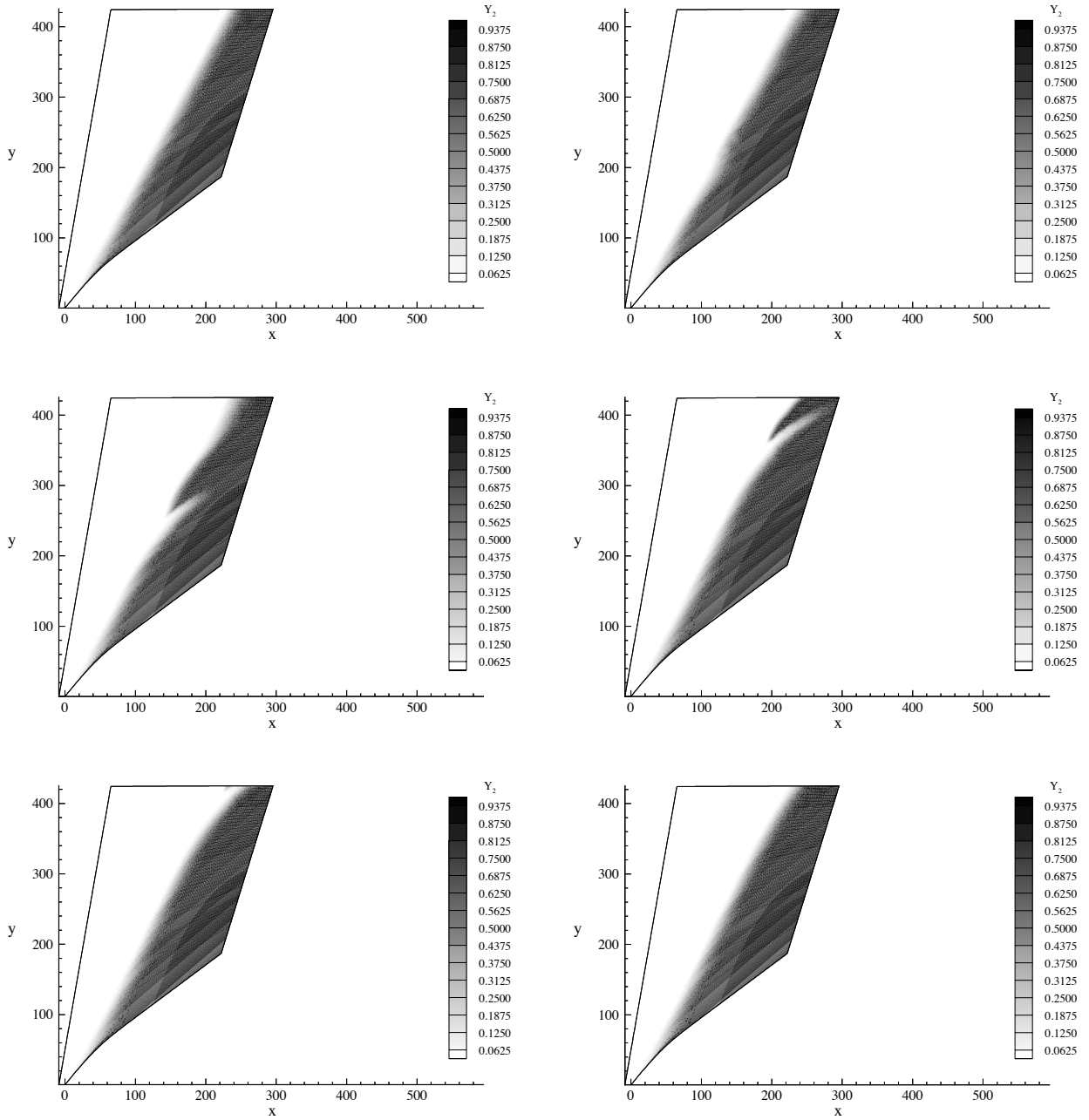


Figure 7.35: Shaded product mass fraction contours as a function of time ($f_n = 1.78$, 184 by 209 grid, $t = 290.9, 756.3, 1105.4, 2181.7, 2647.2, 3839.9$).

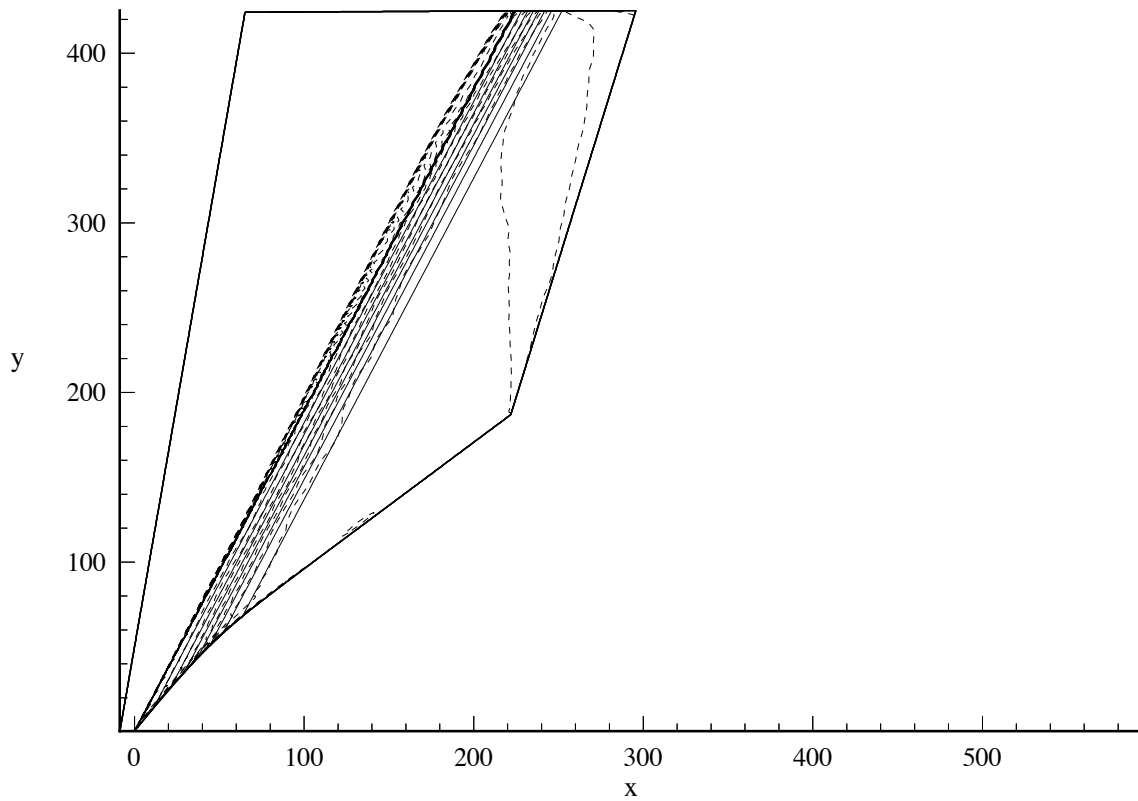


Figure 7.36: Comparison of numerical and exact density contours ($f_n = 1.78$, 184 by 209 grid, $t = 3839.9$, dashed = numerical, line = exact).

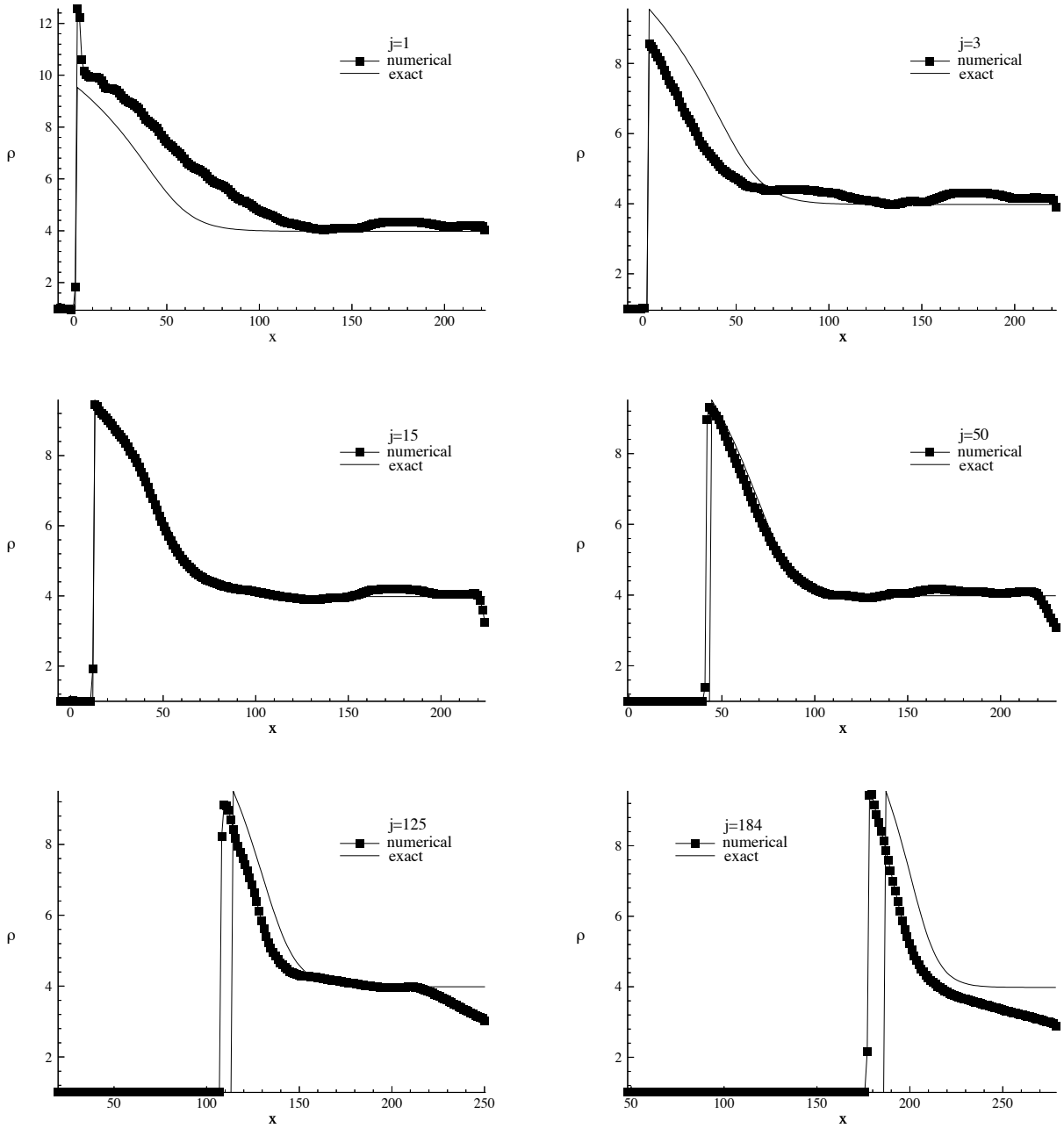


Figure 7.37: Comparison of numerical and exact density solutions along $j = \text{constant}$ grid lines ($f_n = 1.78, 184$ by 209 grid, $t = 3839.9$).

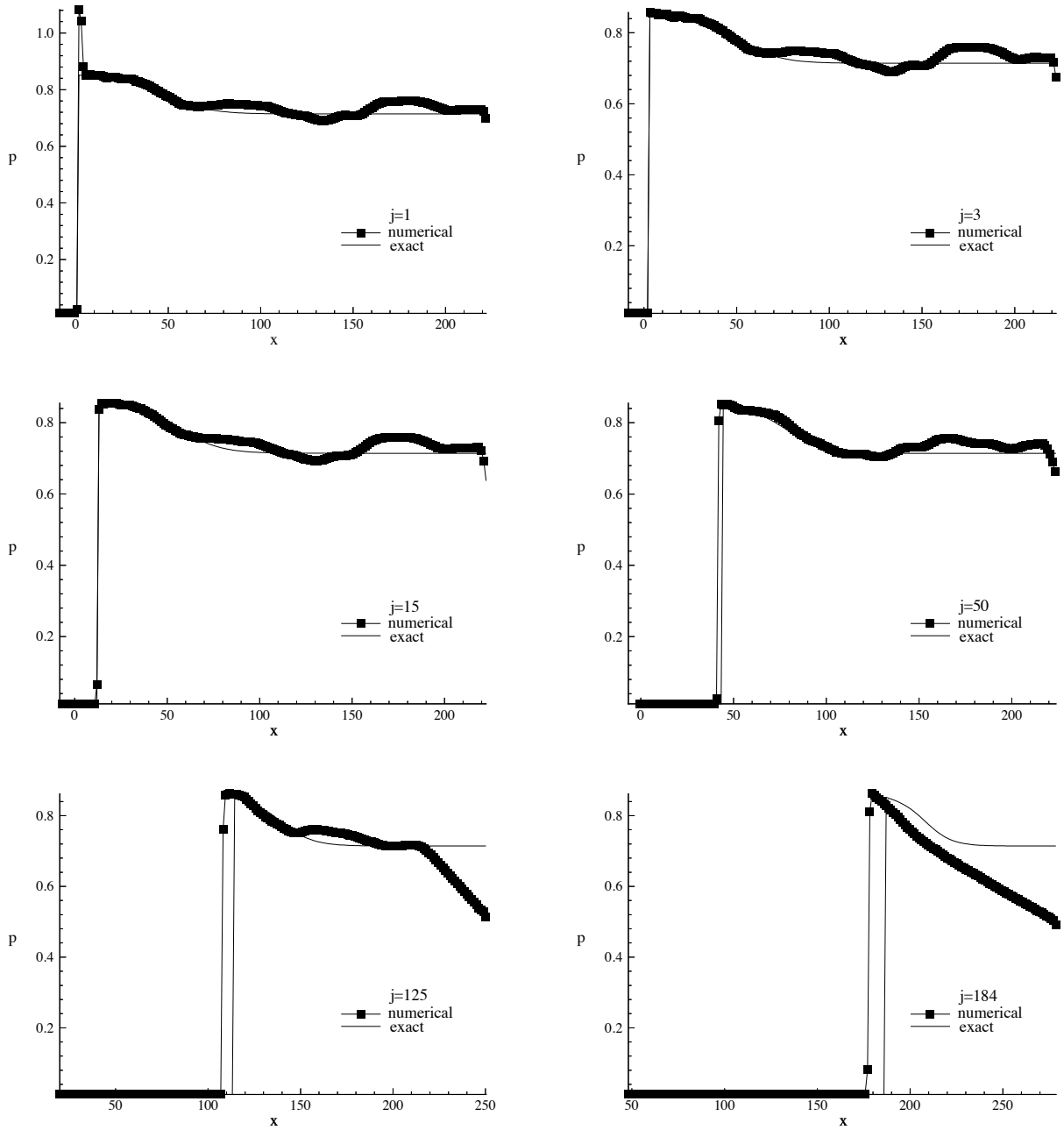


Figure 7.38: Comparison of numerical and exact pressure solutions along $j = \text{constant}$ grid lines ($f_n = 1.78, 184$ by 209 grid, $t = 3839.9$).

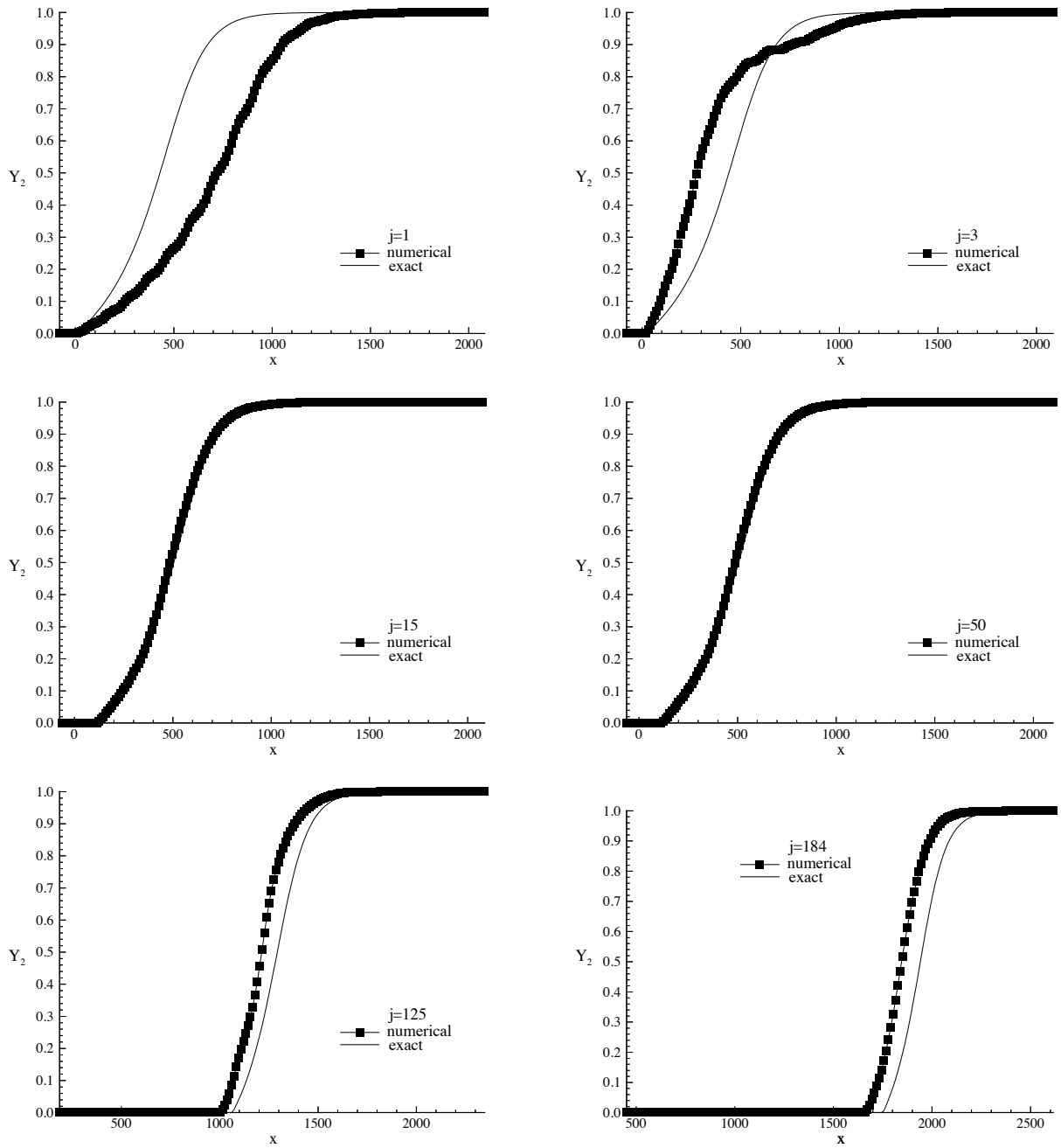


Figure 7.39: Comparison of numerical and exact product mass fraction solutions along $j = \text{constant}$ grid lines ($f_n = 1.78, 184$ by 209 grid, $t = 3839.9$).

7.6 Normal Overdrive $f_n = 1.8$

As a final check on the steady results obtained in the previous section, a number of cases at $f_n = 1.8$ were considered. The freestream Mach number, detonation angle, and initial wedge angle were $\mathcal{M}_0 = 9.419$, $\beta = 62.3^\circ$, and $\theta_{max} = 51.1^\circ$, respectively. Five cases in all were considered, with the number of cell centers ranging from 99 by 99 to 184 by 209, the approximate number of points per $L_{1/2}$ ranging from 12 to 16, and L_{ratio} ranging from 14.7 to 30.7. As was done at $f_n = 1.76$, a final case was run on the largest, most resolved grid with CFL = 0.35. In all cases a stable, steady solution was obtained.

Figures 7.40, 7.41, and 7.42 show the time series solutions for ρ , p , and Y_2 on the 184 by 209 grid with CFL = 0.35. In this case it is immediately apparent that two three-shock structures form, and then both begin to propagate up and out of the domain. Figure 7.43 compares the final steady numerical solution contours for density to the exact steady solution contours. As was the case with $f_n = 1.78$, the final numerical solution bulges somewhat in front of the exact solution in the upper half of the domain. Figures 7.44, 7.45, 7.46, 7.47, and 7.48 show comparisons of the exact and numerical solutions along various $j = \text{constant}$ gridlines for ρ , u , v , p , and Y_2 . The numerical solutions for ρ , p , and Y_2 follow essentially the same pattern of relative agreement/disagreement as they did for $f_n = 1.78$. This pattern also holds true for u and v .

It should be noted that the results of this and previous sections are in some sense similar to the results obtained by Bourlioux and Majda [8] for an unstable two-dimensional detonation. Their results show detonation fronts having multiple cusps with reflected shocks and contact discontinuities behind the front. However, the flow structures generated downstream of the unstable detonation front were much more complex than those observed here. Part of the reason for this may be that the detonations observed here may not have developed fully before leaving the wedge and

exiting through the front of the domain. Bourlioux and Majda considered moving detonations with a grid tracking scheme, and perturbed them initially so as to observe the instability more quickly. In any case, this previous research supports the general form of the detonation front structures observed here.

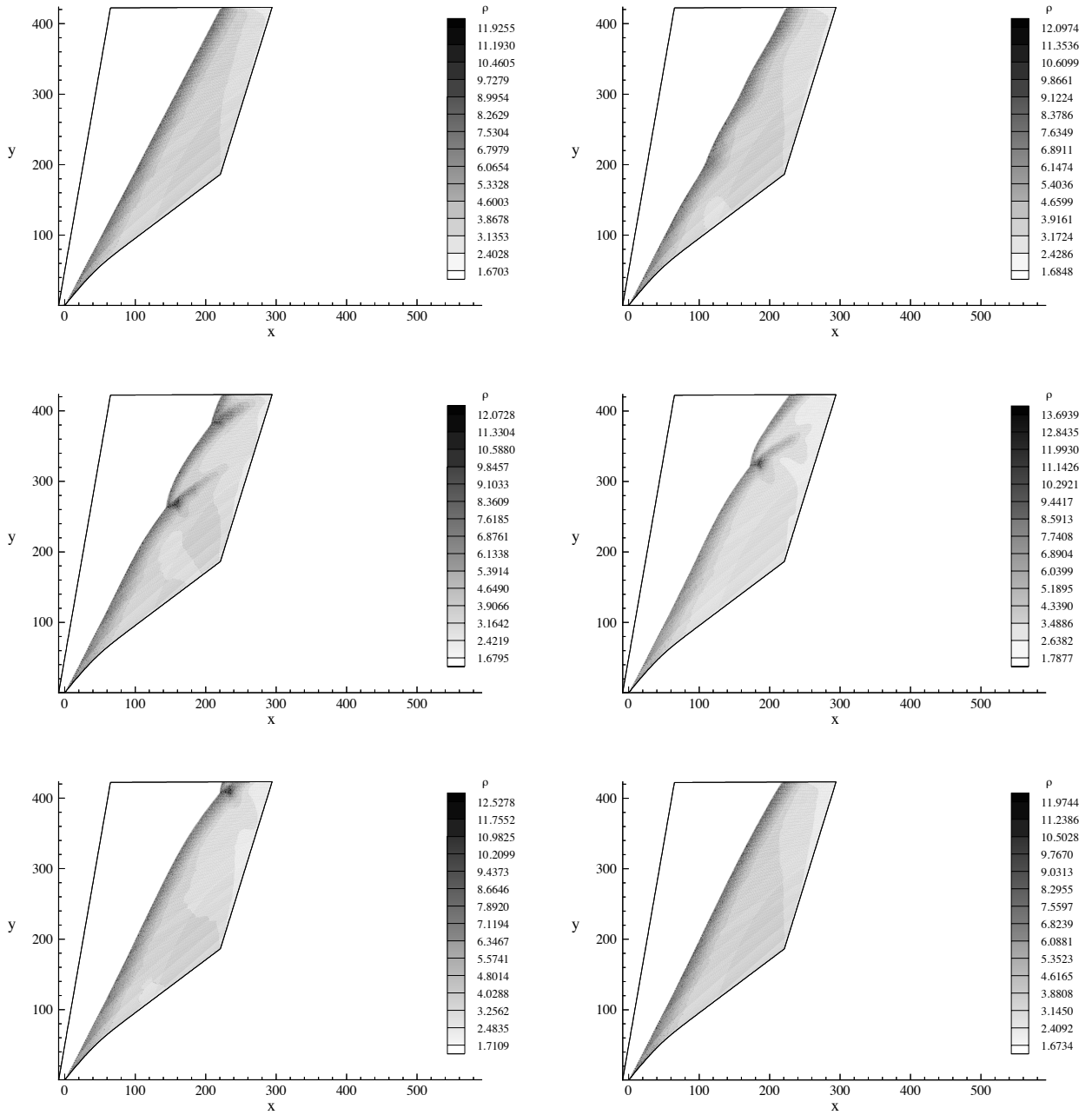


Figure 7.40: Shaded density contours as a function of time ($f_n = 1.8$, CFL = 0.35, 184 by 209 grid, $t = 253.1, 810.1, 1316.6, 1924.2, 2658.5, 4557.5$).

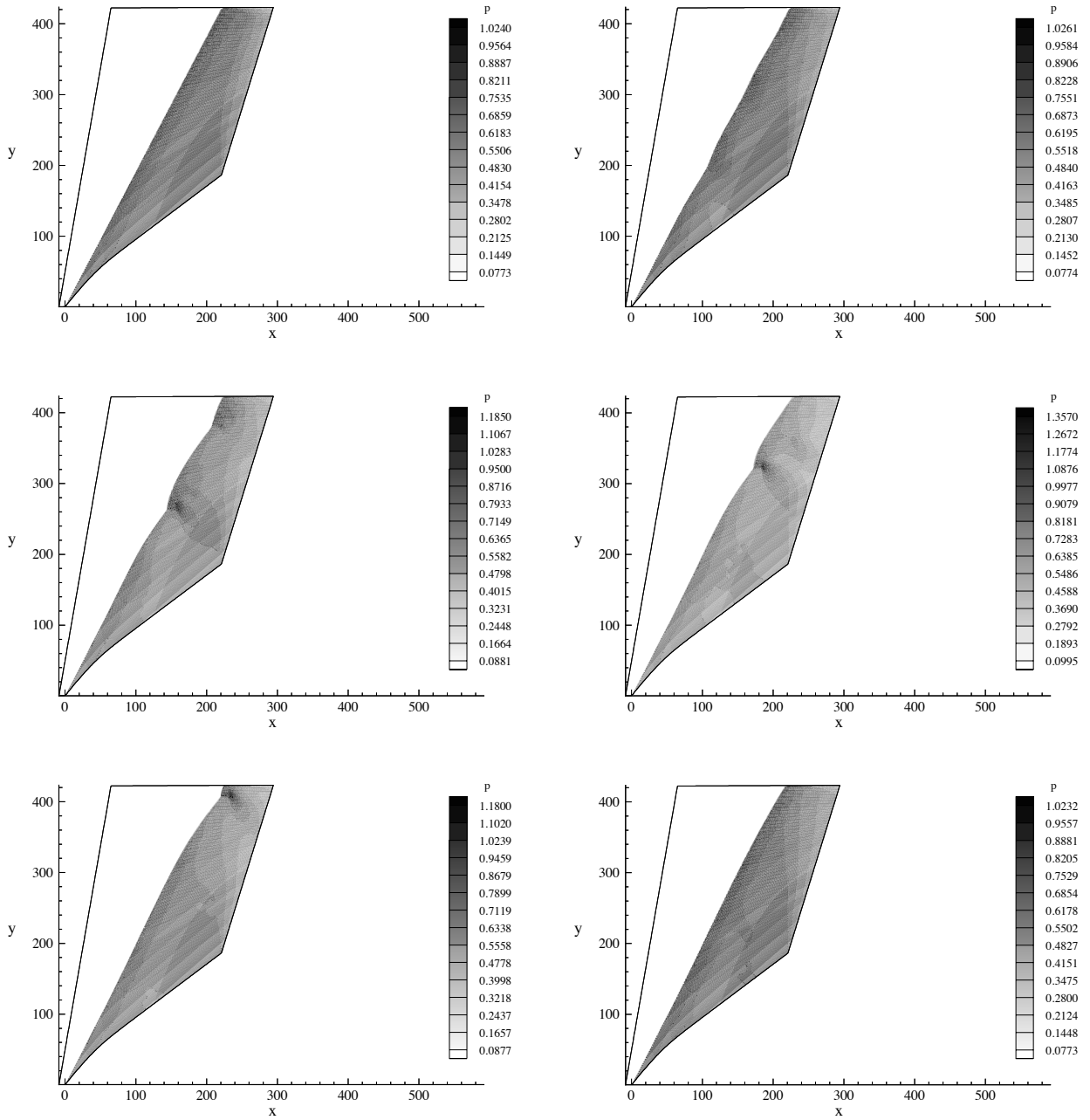


Figure 7.41: Shaded pressure contours as a function of time ($f_n = 1.8$, CFL = 0.35, 184 by 209 grid, $t = 253.1, 810.1, 1316.6, 1924.2, 2658.5, 4557.5$).

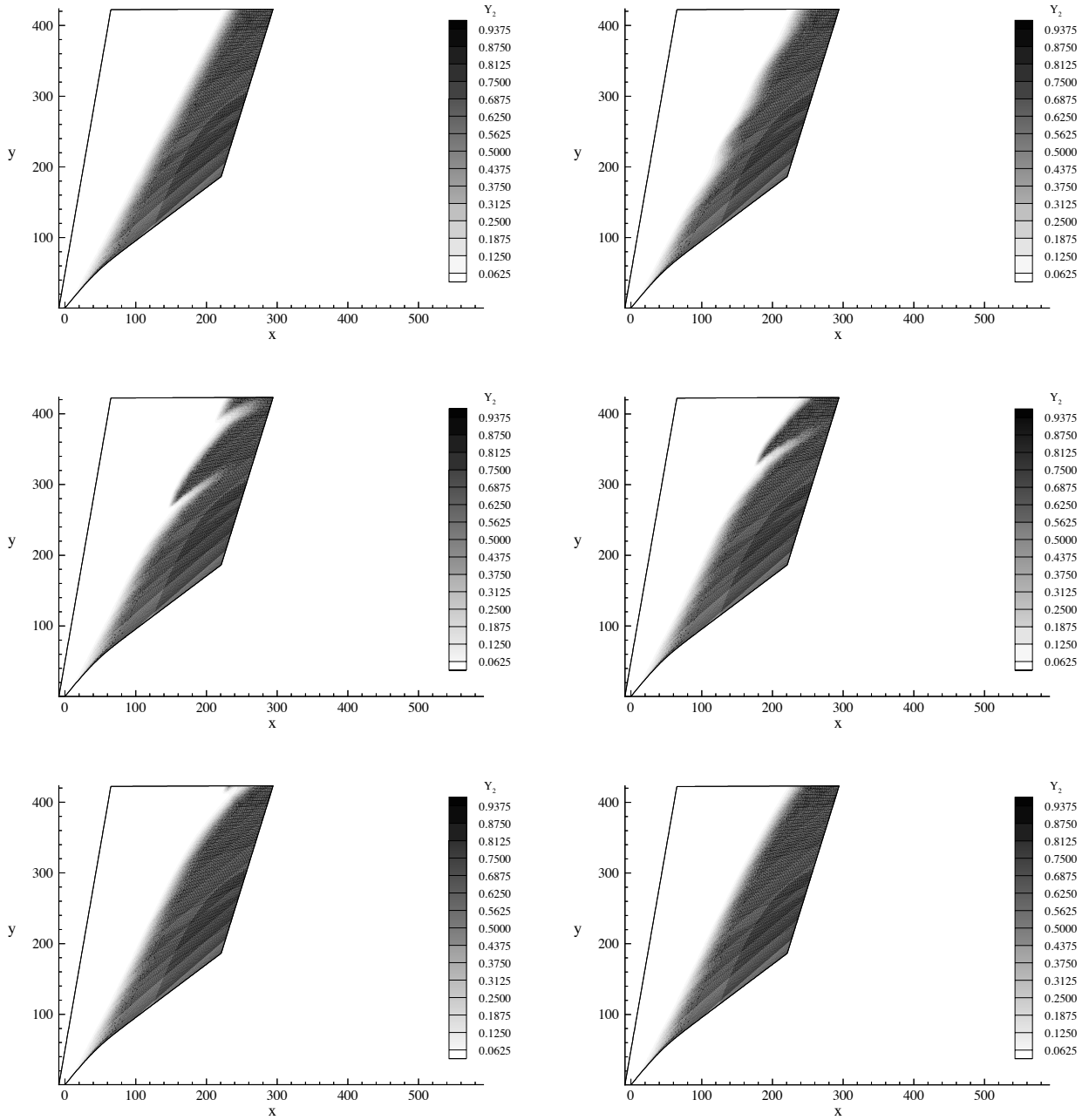


Figure 7.42: Shaded product mass fraction contours as a function of time ($f_n = 1.8$, CFL = 0.35, 184 by 209 grid, $t = 253.1, 810.1, 1316.6, 1924.2, 2658.5, 4557.5$).

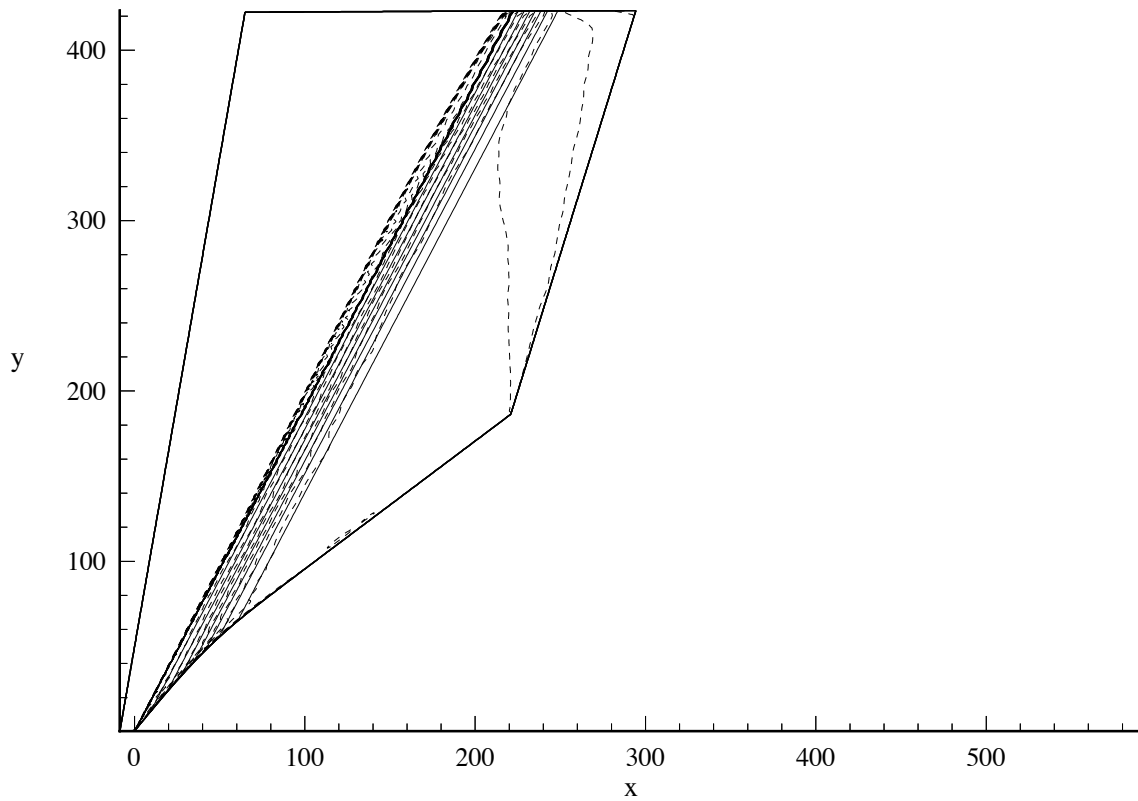


Figure 7.43: Comparison of numerical and exact density contours ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$, dashed = numerical, line = exact).

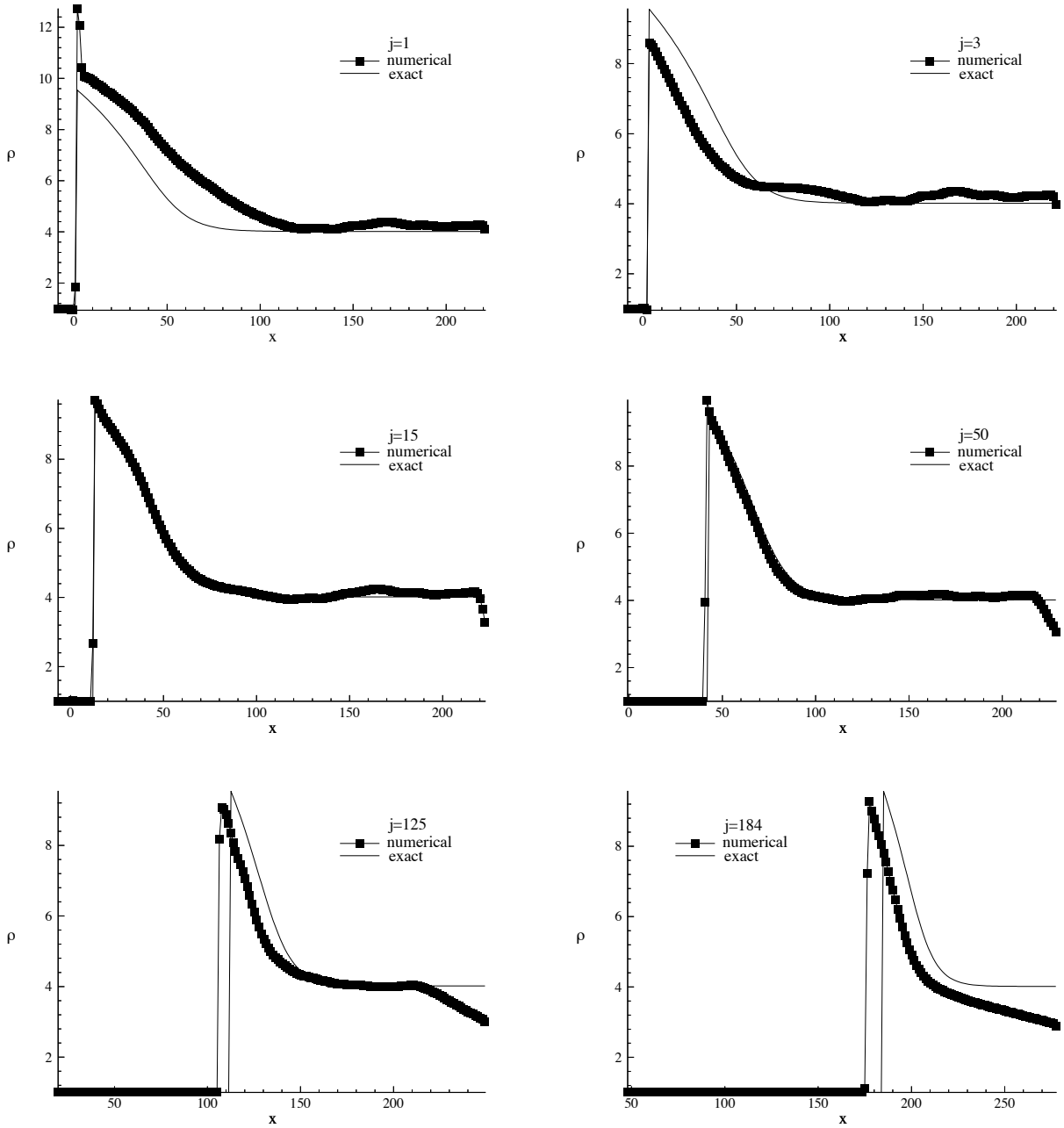


Figure 7.44: Comparison of numerical and exact density solutions along $j = \text{constant}$ grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).

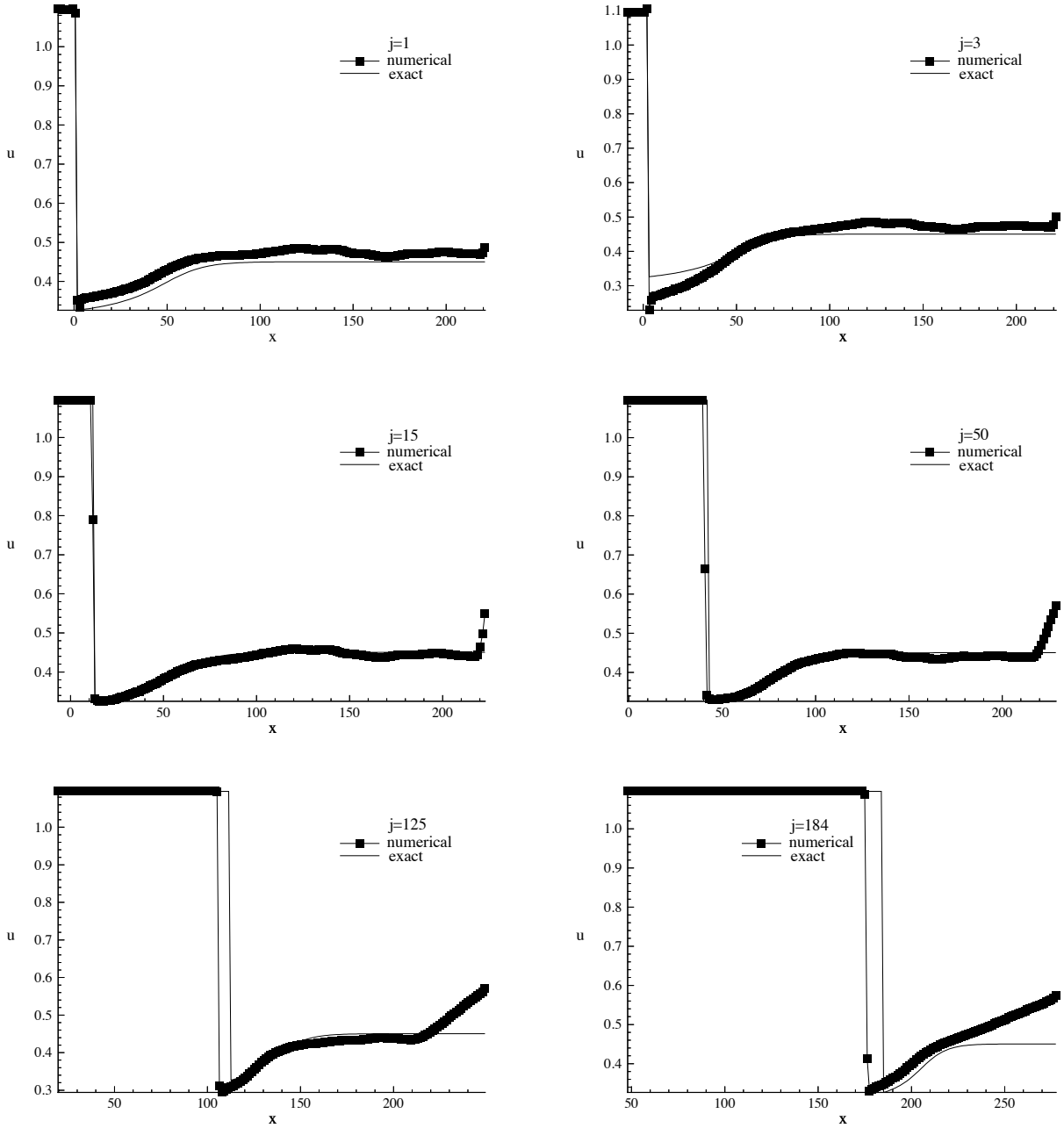


Figure 7.45: Comparison of numerical and exact u component of velocity solutions along $j = \text{constant}$ grid lines ($f_n = 1.8, 184$ by 209 grid, $t = 4557.5$).

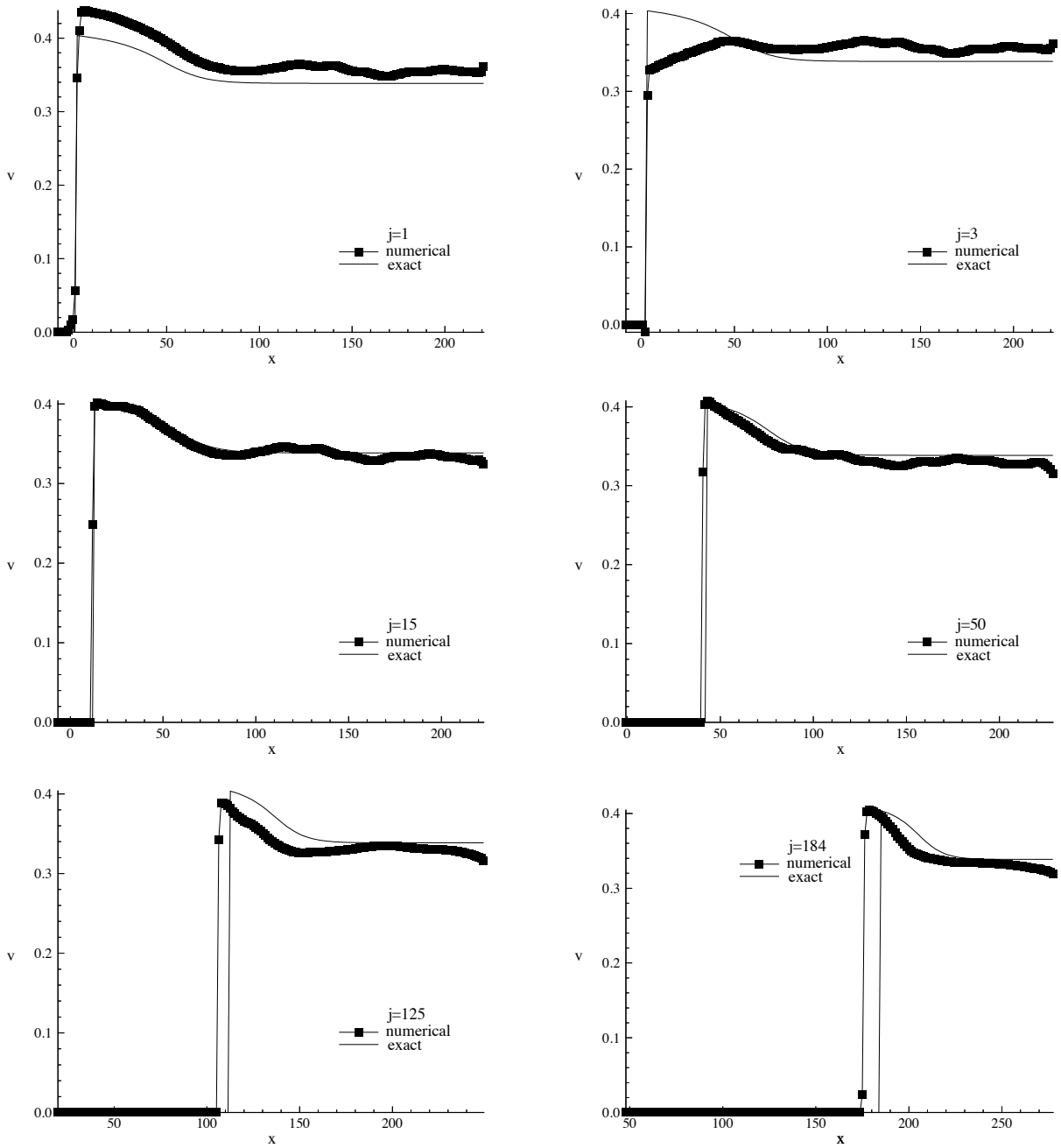


Figure 7.46: Comparison of numerical and exact v component of velocity solutions along $j = \text{constant}$ grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).

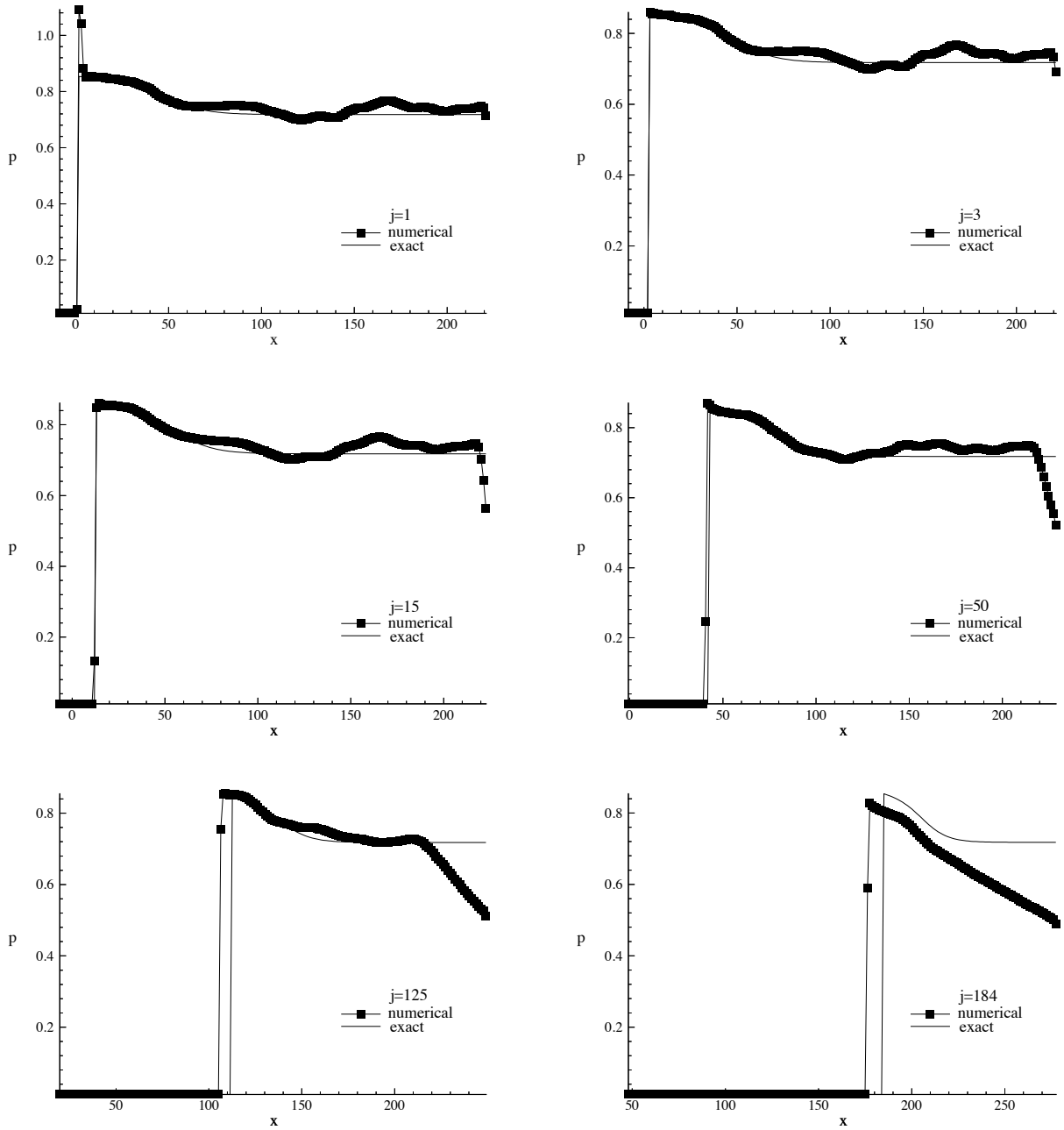


Figure 7.47: Comparison of numerical and exact pressure solutions along $j = \text{constant}$ grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).

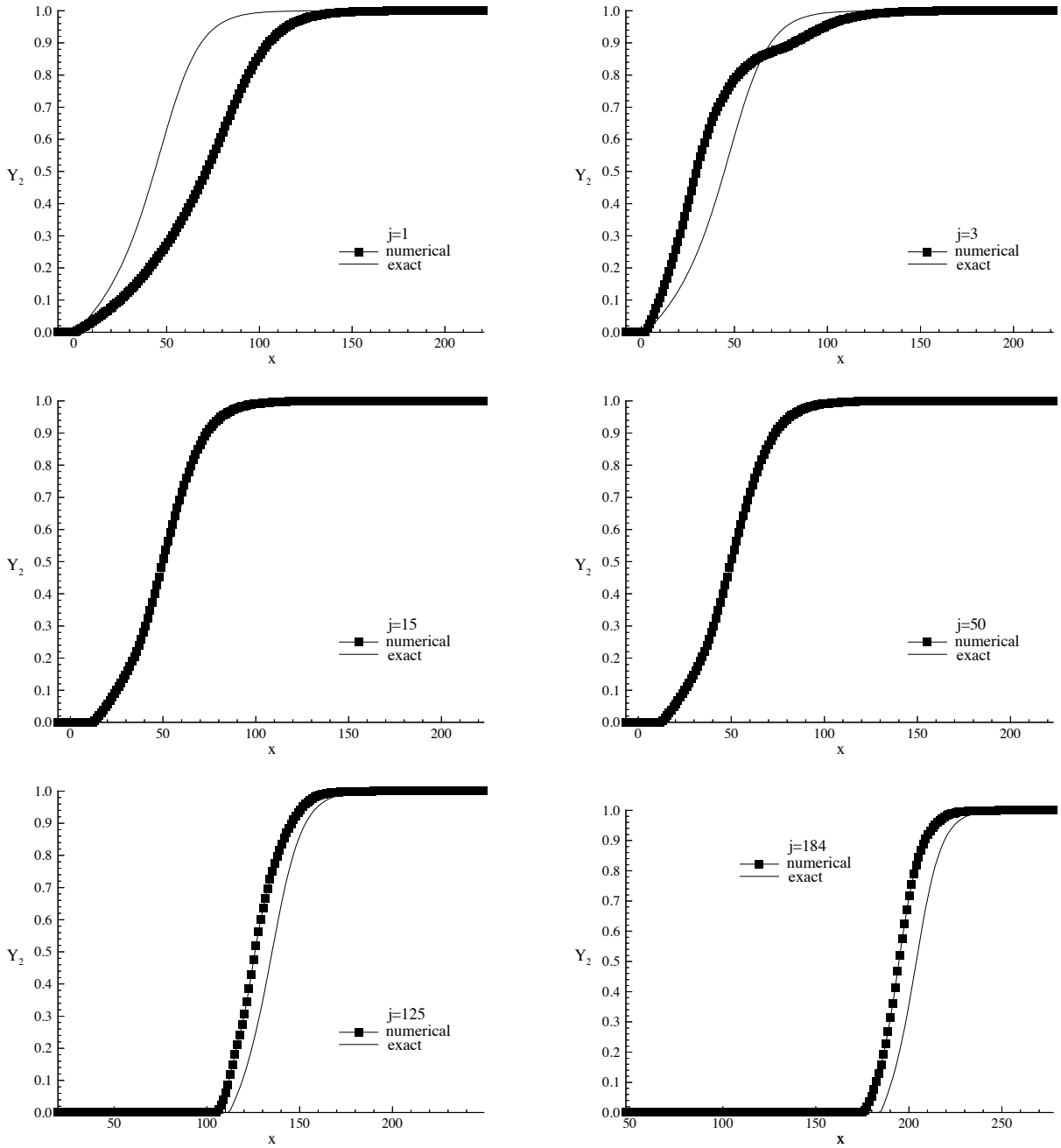


Figure 7.48: Comparison of numerical and exact product mass fraction solutions along $j = \text{constant}$ grid lines ($f_n = 1.8$, 184 by 209 grid, $t = 4557.5$).

Chapter 8

RAM ACCELERATOR CALCULATIONS

This chapter will present a methodology for determining the steady propagation speed of a ram accelerator. The chosen methodology will then be applied to a ram accelerator configuration using a version of the two-dimensional code developed in the preceding chapters. This earlier version used explicit Euler time stepping (eq. (5.71)), and thus had only first-order temporal accuracy; it was verified with the same tests used in Ch. 6. The results of this chapter were recently accepted for publication in a reviewed journal [33].

8.1 Methodology

As discussed in Ch. 2, most recent theoretical studies of ram accelerators and ODWE's have not given analysis to determine a steady propagation speed. Typically the related problem of flow with a fixed incoming Mach number over a fixed geometry is examined. Emphasis is placed on characterizing the resulting flow field, and only a small number of incoming Mach numbers is studied. With this approach, the thrust can be determined as a function of flight speed. This is useful in a transient analysis which presumes that fluid relaxation time scales are short compared to the time it takes for the projectile to relax to a steady velocity.

A general theoretical approach to predict the steady speed is as follows: a math-

emathical model for the fluid and a representative geometry are selected. The model equations are studied in the reference frame in which the projectile is stationary; thus, the incoming flow velocity, which is the steady propagation speed, is thought of as an adjustable parameter at this stage. For a given incoming velocity, solution of the model equations leads to a stress distribution on the projectile surface which may or may not result in a net force on the projectile. Should the particular incoming velocity lead to zero net force on the projectile, that velocity is a candidate for a steady propagation speed. The static stability of the candidate solutions is easily determined. Should a perturbation in the incoming velocity lead to a net force which tends to restore the projectile to a speed at which there is zero net force, the solution is stable in a static sense (we call such solutions stable); otherwise the solution is unstable. A further step, not considered here, is to account for the inertia of the projectile and surrounding fluid so as to determine the dynamic stability.

8.2 Model Problem

The methodology is illustrated through the use of a model problem which is related to the ram accelerator and ODWE. For tractability, an idealized model and geometry are considered which retain the essential features of real systems. The geometry, shown in Fig. 8.1, is a symmetric double wedge with half angle θ and length \tilde{L}_p . Two cowl surfaces are placed symmetrically about the wedge and are separated by height \tilde{H}_c . The depth of the double wedge and cowl is taken to be infinite and the flow is assumed to have no variation in this direction. The Cartesian coordinate system, with its origin at the leading edge and with the \tilde{x} axis aligned with the incoming flow is also indicated. It is appropriate to think of a ram accelerator as the axisymmetric analog of Fig. 8.1 in which the projectile moves while the cowl is stationary; likewise, an aerospace plane powered by an ODWE can be thought of as the axisymmetric analog of Fig. 8.1 in which the cowl moves with the wedge. In both scenarios one

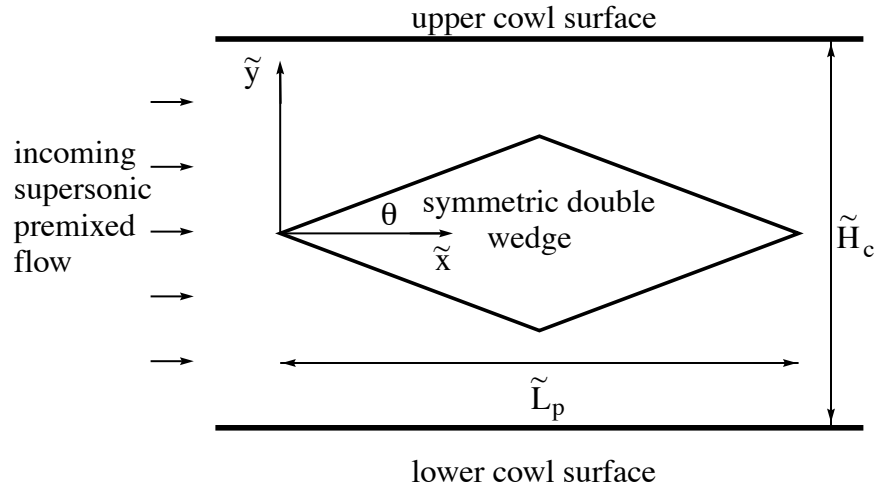


Figure 8.1: Schematic of generic configuration

must assume that the incoming fuel and oxidizer are completely mixed; in actuality this is more appropriate for the ram accelerator than the ODWE.

Analysis of the geometry of Fig. 8.1 leads in general to a complicated interaction of shocks, rarefactions, and combustion processes as the flow propagates between the projectile and cowl surface. To further simplify, we consider only the limit $\tilde{H}_c \rightarrow \infty$, Fig. 8.2. Consequently, our geometry shares only a rudimentary resemblance to actual

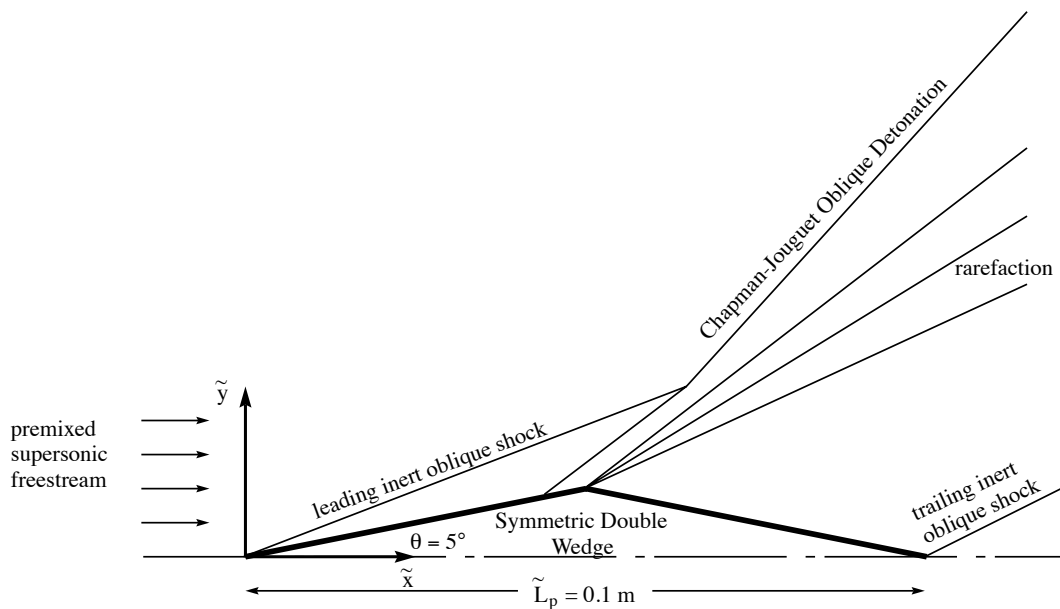


Figure 8.2: Detailed schematic for $\tilde{H}_c \rightarrow \infty$

devices, but has the advantage of being amenable to simple analysis.

As discussed previously, the flow model employed also has only a rudimentary resemblance to commonly used models for real devices. Fig. 8.2 indicates the general flow features. The ambient, premixed freestream fluid encounters an attached oblique shock at the leading edge of the projectile. No appreciable reaction occurs within the shock or near the front of the projectile. Near the apex of the wedge appreciable reaction begins, and at the apex the flow is turned through a centered Prandtl-Meyer expansion until it attains a velocity parallel to the lee wedge surface. The reaction then proceeds to completion on the leeward side of the projectile. The flow passes through a final oblique shock at the tail of the projectile, resulting in a velocity only in the \tilde{x} direction. The net force on the projectile is determined by integrating the pressure over the entire surface area. The lead oblique shock wave is strengthened by the reaction such that in the far-field the shock angle reaches the Chapman-Jouguet wave angle; this will be demonstrated in the following section. Chapman-Jouguet detonations are known to be unstable with this reaction mechanism, however, so additional study is needed to completely verify this result.

8.3 Results

For the unconfined double wedge, steady propagation speeds were sought which gave rise to a force balance as the heat release parameter \bar{q} was varied, $11.908 \leq \bar{q} \leq 13.456$. Other parameters were held constant at $\gamma = 7/5$, $\theta = 5^\circ$, and $\bar{E}_a = 12.32$. For presentation of results the corresponding dimensional values were $\tilde{p}_0 = 1.01325 \times 10^5$ Pa, $\tilde{\rho}_0 = 1.225$ kg/m³, $\tilde{E}_a = 1.019 \times 10^6$ J/kg, $\tilde{L}_p = 0.1$ m, 0.985×10^6 J/kg $\leq \tilde{q} \leq 1.113 \times 10^6$ J/kg, and $\tilde{B} = 2.64 \times 10^7$ s⁻¹. These values were chosen not so much to model a real system but so that the method could be successfully illustrated and an interesting bifurcation phenomenon predicted. For models which better represent physical systems, it is certain that the method given here can be applied and plausible

that the predictions will have the same essence. A common 199 x 99 fixed computational grid was used in all cases, and convergence to steady-state was typically achieved in about 5000 time iterations.

As shown in Eq. 4.30, the CJ Mach number $\mathcal{M}_{0_{CJ}}$ is determined solely by \bar{q} and γ ; for the parameters listed $4.275 \leq \mathcal{M}_{0_{CJ}} \leq 4.517$. For cases attempted in which $\mathcal{M}_0 \leq \mathcal{M}_{0_{CJ}}$, a normal detonation would form and propagate forward in the domain until it hit the inflow boundary. This corresponds to the detonation attempting to reach its natural, unsupported propagation speed. All cases considered here in which force balances were found had an incoming Mach number well above $\mathcal{M}_{0_{CJ}}$ ($5.5 \leq \mathcal{M}_0 \leq 8.5$); consequently, the steady propagation speeds found are in the superdetonative regime.

The projectile achieves a steady velocity when the force due to pressure wave drag, which tends to retard the motion, is balanced by forces induced by combustion, which tend to accelerate the projectile. The dimensionless net force per unit depth F_{net} is given by the pressure force integrated over the circumference of the diamond-shaped wedge:

$$F_{net} = \oint pn_i ds, \quad (8.1)$$

where ds is an element of arc length of the diamond-shaped wedge of Fig. 8.1. Due to symmetry, the only non-zero component of F_{net} is in the x direction. This force is defined to be positive if it points in the negative x direction. For the numerical analysis, numerical integration of the pressure field gave the net thrust. Figure 8.3 shows F_{net} plotted vs. \mathcal{M}_0 for the four indicated values of \tilde{q} .

For low heat release F_{net} is negative; the thrust force induced by combustion is not sufficient to overcome the wave drag. At a critical value of heat release, $\tilde{q} = 0.992 \text{ MJ/kg}$ ($\bar{q} = 11.993$), there is a balance of combustion-induced thrust and drag such that $F_{net} = 0$. This occurs at $\mathcal{M}_0 = 7.95$. As heat release continues to increase, there are two distinct Mach numbers for which there is no net thrust. A

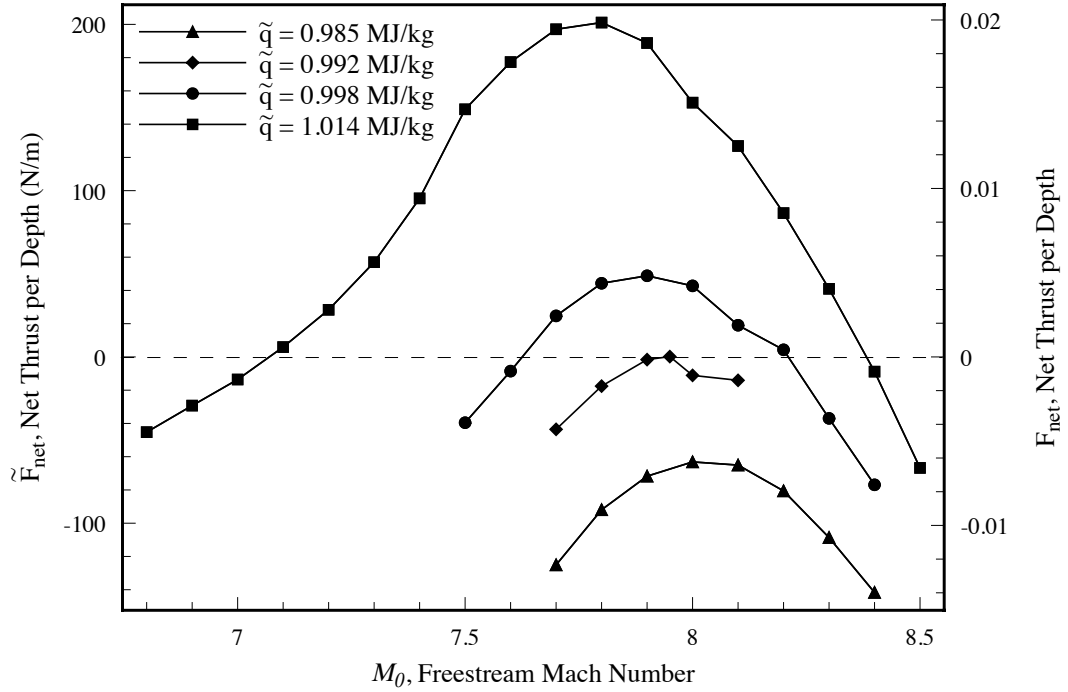


Figure 8.3: Net thrust force versus Mach number for varying heat release.

perturbation in the Mach number for the steady solution at the lower Mach number results in a net force which tends to accelerate the projectile away from the equilibrium Mach number. Consequently, this is a statically unstable equilibrium. In the same manner, the equilibrium solution at the higher Mach number is statically stable to such perturbations. As heat release is increased, the stable equilibrium Mach number increases and the reaction zone is located closer to the expansion fan, while the unstable Mach number decreases and the reaction zone is located closer to the trailing edge. The same trends of these numerical results can be predicted with a simple Rankine-Hugoniot analysis combined with thermal explosion theory [61, 60].

These results are summarized in the bifurcation diagram shown in Fig. 8.4, where Mach number versus heat release \tilde{q} are plotted. The lower branch is unstable while the upper branch is stable. On the stable branch near the bifurcation point, an increase in \tilde{q} causes the flight speed to increase. The curve of $\mathcal{M}_{0_{CJ}}$ versus \tilde{q} is also shown on this figure to demonstrate that the cases considered were all above the CJ point.

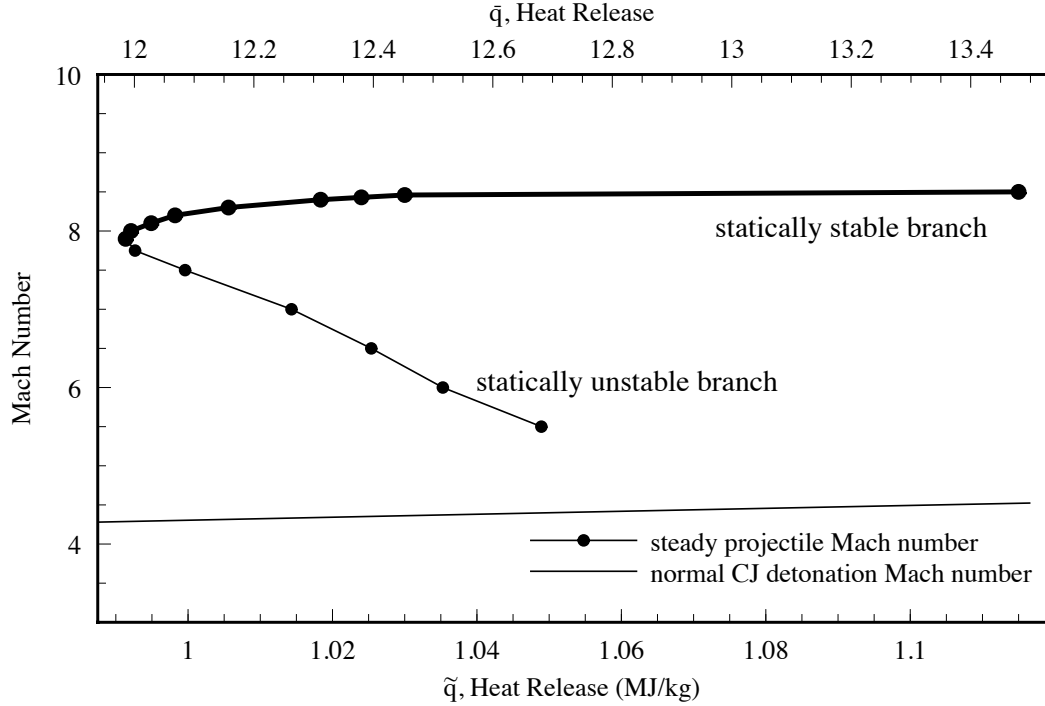


Figure 8.4: Bifurcation diagram for steady state speed versus heat of reaction.

The solutions shown here correspond to stable flight speeds in the range of $2,700\text{ m/s} \lesssim \tilde{u}_0 \lesssim 2,900\text{ m/s}$, $7.9 \lesssim \mathcal{M}_0 \lesssim 8.5$.

For a particular value of heat release, $\tilde{q} = 1.014\text{ MJ/kg}$, detailed plots of pressure contours and product mass fraction contours are given for the stable case ($\mathcal{M}_0 = 8.4$) and the unstable case ($\mathcal{M}_0 = 7.1$) in Figs. 8.5, 8.6, 8.7, 8.8, respectively. In the stable configuration, the lead oblique shock undergoes a sudden increase in angle of inclination from approximately 11° to 31° . A similar rise from 12° to 38° occurs for the unstable case. This appears to be associated with the chemical reaction. The reaction occurs sooner for the stable case which is at the higher Mach number. This is readily apparent in the product mass fraction contours. Along the wedge surface chemical reaction reaches completion very near the apex for the stable case, while in the unstable case the reaction completes further downstream. This may be explained in the following way: for low \mathcal{M}_0 , chemical reaction occurs off the leeward wedge surface far downstream, resulting in a net drag force. As the Mach number

is increased, the reaction moves forward onto the wedge, eventually reaching a point at which the wave drag of the projectile is balanced by the thrust due to chemical reaction. Increasing \mathcal{M}_0 past this equilibrium point moves the reaction closer to the wedge apex on the leeward side, resulting in positive F_{net} . Increasing \mathcal{M}_0 still further pushes the reaction over the apex and onto the front of the wedge. The pressure increase due to chemical reaction on the front of the wedge is then balanced by the resulting higher pressure on the leeward side of the wedge.

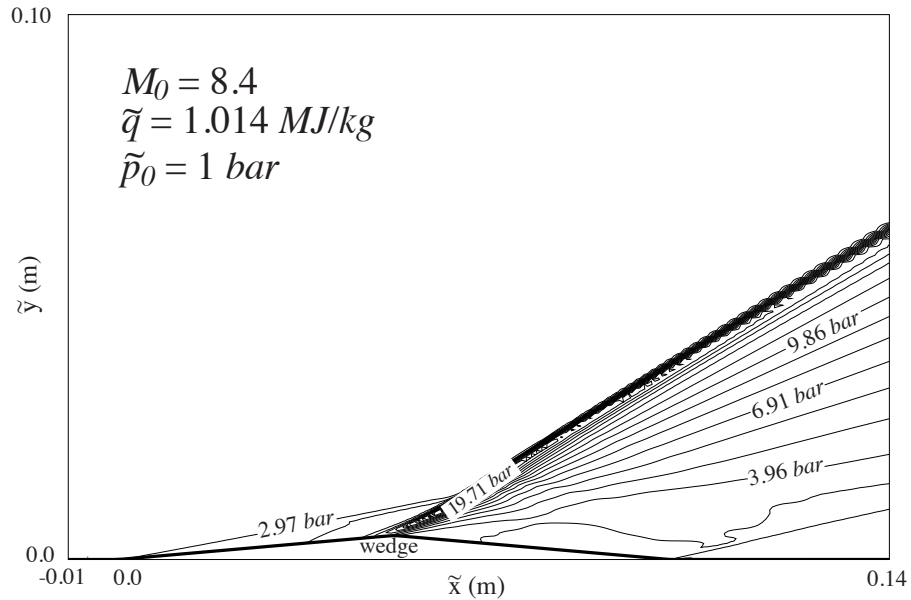


Figure 8.5: Pressure contours for statically stable steady configuration.

Figure 8.9 shows plots of the pressures along the lines of symmetry and projectile surface for the stable and unstable cases of interest. Here there are about 90 grid points distributed on the wedge surface. As a further verification of the code's ability to predict two-dimensional flows, plots of the exact and numerical pressure traces for an inert flow over the projectile are also given. The numerical pressure closely follows the exact solution, showing the biggest discrepancies at the shock and rarefaction discontinuities. The discontinuities are still captured well, however, and there is no

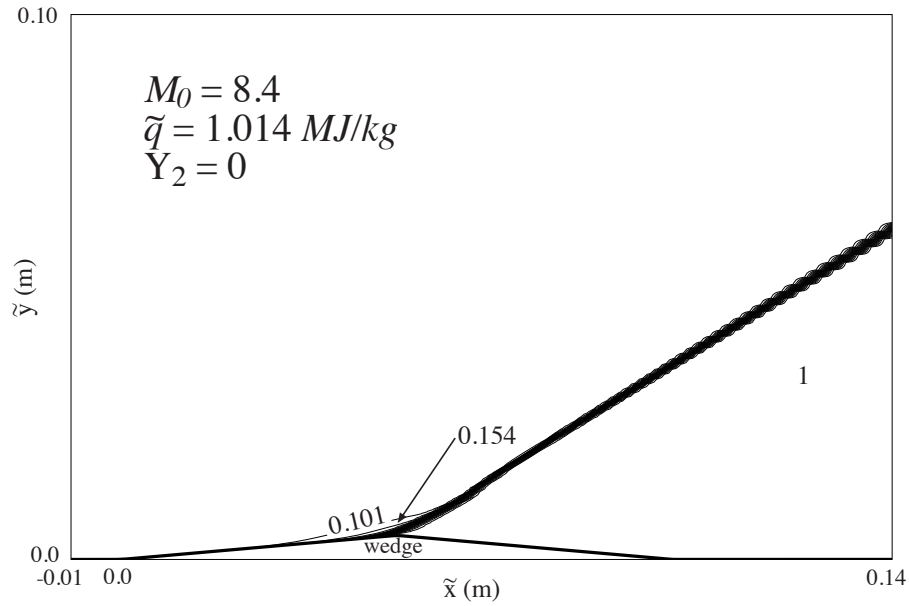


Figure 8.6: Product mass fraction contours for statically stable configuration.

evidence of the Gibbs phenomena observed in some of the verification studies of Ch. 6. Between the discontinuities, the numerical solution reaches constant states very close to those of the exact solution. Drag calculations for the exact and numerical solutions show that they are in excellent agreement.

Considering the surface pressure for the stable case in Fig. 8.9, it is apparent that significant reaction occurs on the front face of the wedge. The pressure begins to rise slowly following the initial shock, and then very suddenly prior to the rarefaction at the wedge apex. Following the apex the pressure remains well above the inert case, and then jumps again because of the trailing shock at the end of the projectile. The unstable case shows a much different solution. Once again the pressure begins to rise slowly following the initial shock, but in this case the peak prior to the rarefaction is much lower. The pressure drops through the rarefaction discontinuity, but once again remains above the inert pressure. The pressure peaks on the back of the wedge where the reaction reaches completion, and then jumps through the trailing shock. This is

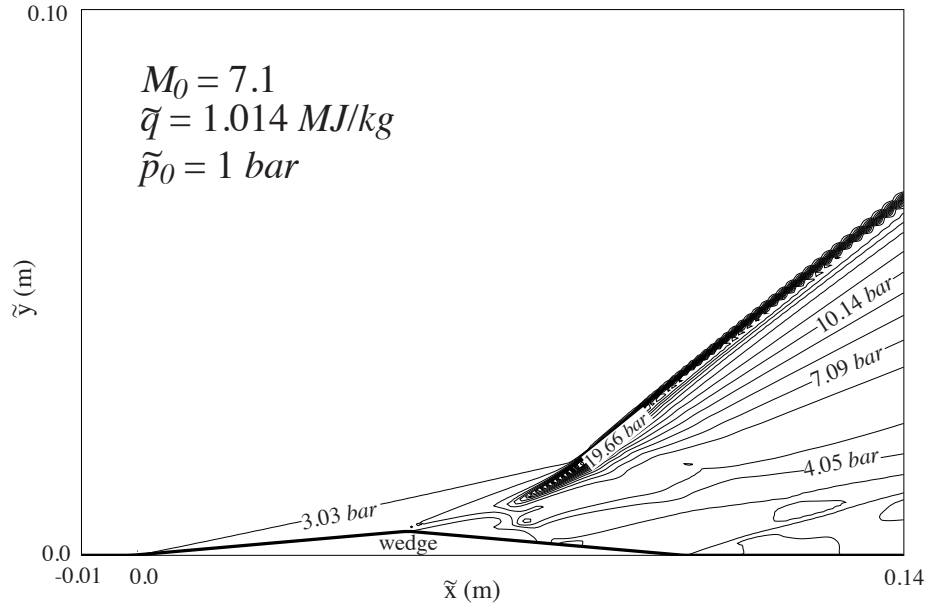


Figure 8.7: Pressure contours for statically unstable steady configuration.

consistent with the prior discussion.

In order to determine the performance characteristics of the simple configuration studied, a number of additional calculations were made using the results obtained for $\tilde{q} = 1.014 \text{ MJ/kg}$. As indicated in Fig. 8.3, the maximum thrust on the projectile was obtained at this heat release. For reference, the heats of combustion of propane, methane, and hydrogen are 50, 56, and 120 MJ/kg , respectively [82]; thus, this level of heat release is small compared to real systems. Approximate position, velocity and acceleration histories for the projectile were obtained by fitting the thrust curve of Fig. 8.3 with a sixth order polynomial to get $\tilde{F}_{net}(\tilde{u}_p)$ (see Fig. 8.10), and then integrating Newton's second law

$$\tilde{F}_{net}(\tilde{u}_p) = \tilde{m} \frac{d\tilde{u}_p}{d\tilde{t}}, \quad (8.2)$$

where \tilde{m} is the dimensional projectile mass per unit depth. Equation (8.2) was integrated from a speed slightly greater than that of the unstable steady propagation point (it was assumed that $\tilde{t} = \tilde{x} = 0$ initially), to a speed slightly less than that

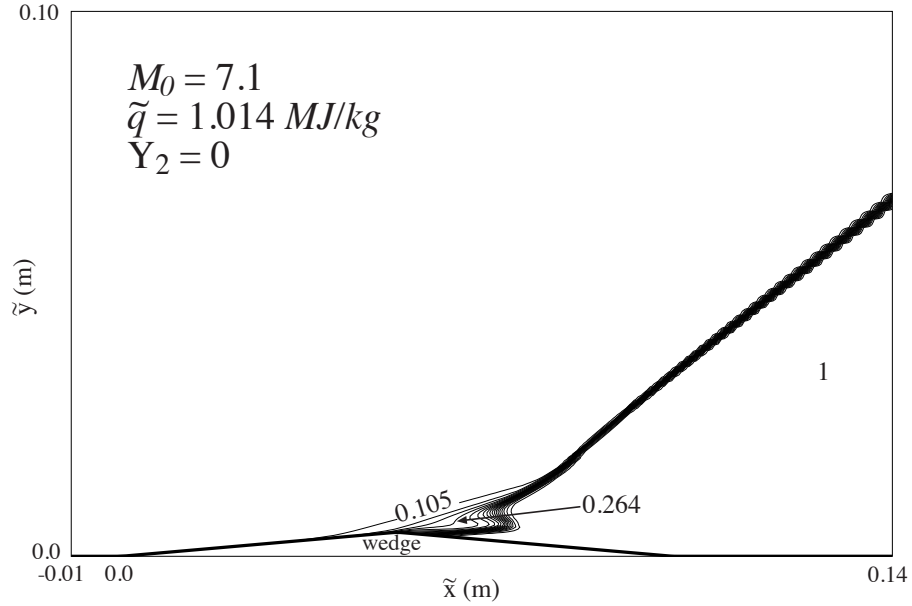


Figure 8.8: Product mass fraction contours for statically unstable steady configuration.

of the stable point. The projectile was assumed to be constructed of aluminum, which resulted in a mass per depth of $\tilde{m} = 1.2108 \text{ kg/m}$. Figures 8.11 and 8.12 show acceleration as a function of time and distance. As expected, the acceleration peaks at the point of maximum thrust. The accelerations found here are much lower than those which Hertzberg reported (approx. $160,000 \text{ m/s}^2$ [38]). The low value obtained here is a result of the low levels of heat release considered, and the lack of cowl surfaces, which would increase \tilde{F}_{net} by creating a system of shock reflections on the rear of the projectile. Figures 8.13 and 8.14 show the velocity histories for the same projectile mass. The velocity changes quickly in the region of maximum thrust, but changes very slowly near the equilibrium points. In this case the total velocity change $\Delta\tilde{u}_p \simeq 430 \text{ m/s}$ ($\Delta u_p \simeq 0.19$) is only slightly less than that observed by Hertzberg (approx. 500 m/s). Figure 8.15 shows the projectile position as a function of time. In this case the curve is almost perfectly straight, since the velocity change due to the chemical reaction is relatively small compared to the overall speed of the

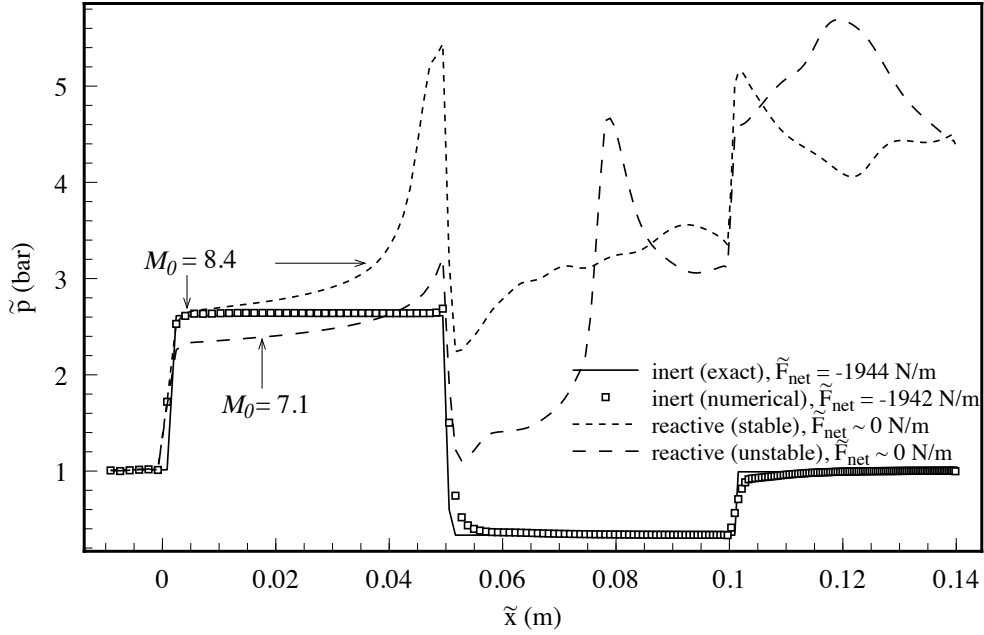


Figure 8.9: Pressure traces on wedge surface.

projectile. The time required for the projectile to transition from the unstable point to the stable point was $\Delta\tilde{t} \simeq 17 \text{ s}$ ($\Delta t \simeq 4.6 \times 10^8$), while the corresponding distance covered was $\Delta\tilde{x} \simeq 44,000 \text{ m}$ ($\Delta x \simeq 5.2 \times 10^8$). This length is obviously impractical; the experimental studies covered a total length of slightly less than 5 m [38].

Finally, it is noted that in the far-field limit the oblique detonation apparently becomes an obliquely travelling, unsupported CJ detonation; the wave angle β of the detonation is such that the component of the Mach number normal to the wave is equal to \mathcal{M}_{0CJ} . This is illustrated in Fig. 8.16, which compares the exact CJ wave angle to the wave angle predicted from the numerical results for the steady speeds of propagation (Fig. 8.4). Error bars of $\pm 1^\circ$ are shown on the numerical results to indicate the estimated error in measuring the numerical detonation angle. Note that β is a function of both \mathcal{M}_0 and \bar{q} , both of which are different for each plotted point on the figure. Figure 8.17 shows the same type of comparison, except that in this case the freestream Mach number has been held fixed at $\mathcal{M}_0 = 8.4$. Thus, it is possible

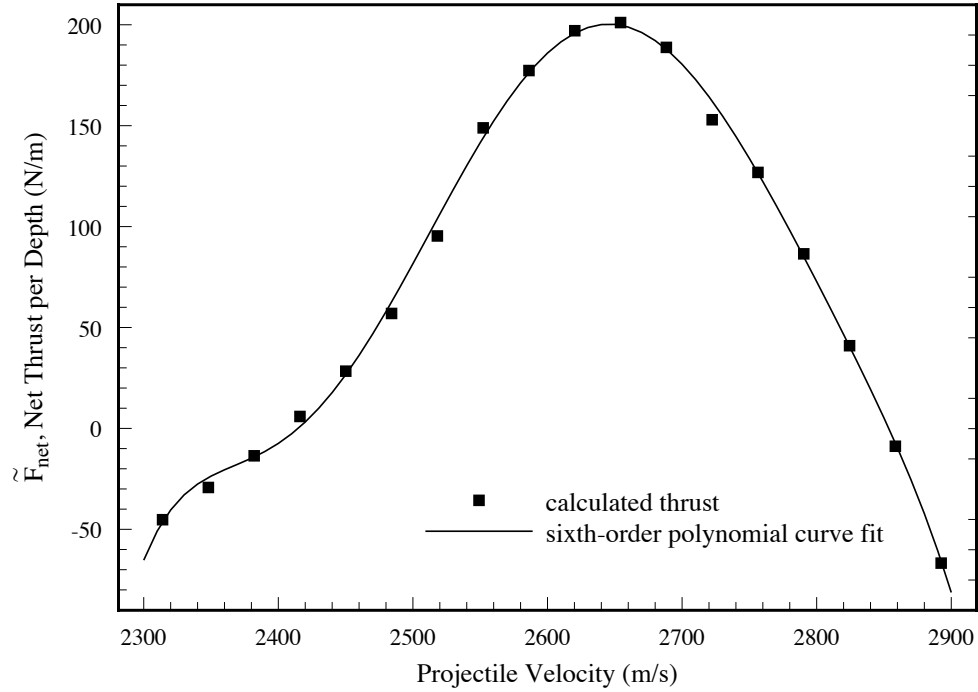


Figure 8.10: Sixth-order polynomial curve fit of thrust curve for $\tilde{q} = 1.014 \text{ MJ/kg}$.

to plot an exact curve for the CJ wave angle. In both figures it can be seen that the numerical prediction and the exact solution are equal within the measurement error. This result gives numerical confirmation to ideas put forth by Chernyi [18].

Figure 8.18 shows a comparison between the exact CJ detonation pressure solution and the numerically predicted far-field pressure solution along a line normal to the oblique detonation ($\mathcal{M}_0 = 8.4$ and $\tilde{q} = 0.985 \text{ MJ/kg}$). The numerical solution is interpolated since the computational grid does not follow this line. It is apparent that the numerically predicted peak pressure is significantly lower than the exact pressure, and that the numerical solution drops well below the exact solution for larger values of \tilde{x} . The discrepancy in the peak pressures is likely the result of an insufficient number of cell centers (approx. 3–4) within the reaction zone of the numerical solution. The increased pressure drop in the numerical solution is due to the Prandtl-Meyer expansion emanating from the apex of the projectile. As previously mentioned, CJ detonations are known to be unstable with this reaction mechanism,

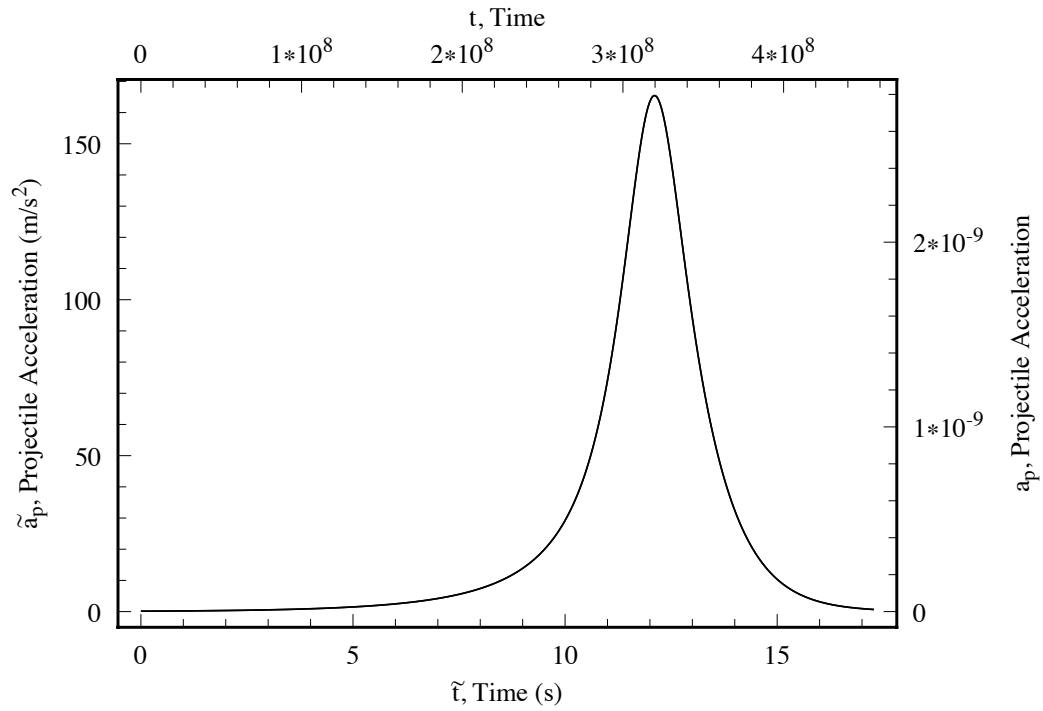


Figure 8.11: Projectile acceleration as a function of time ($\tilde{q} = 1.014 \text{ MJ/kg}$, $\tilde{m} = 1.2108 \text{ kg/m}$).

and yet no instabilities were found in the cases studied. The verification studies of Ch. 6 indicated, however, that a much greater number of points within the reaction zone is required to accurately capture detonation instability.

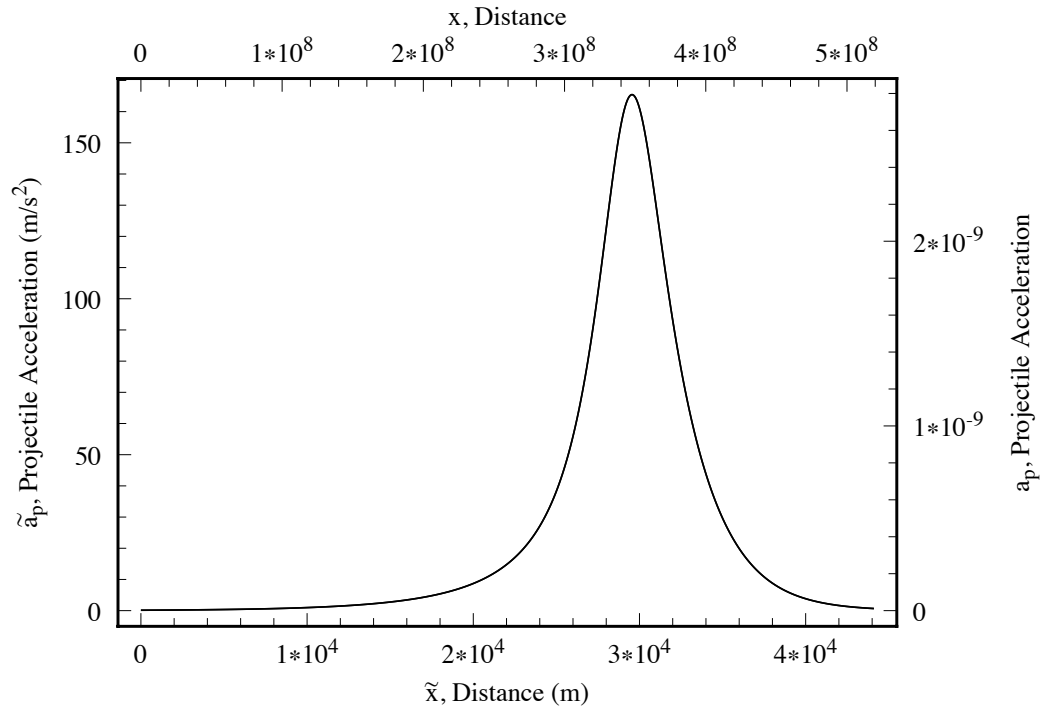


Figure 8.12: Projectile acceleration as a function of distance ($\tilde{q} = 1.014 \text{ MJ/kg}$, $\tilde{m} = 1.2108 \text{ kg/m}$).

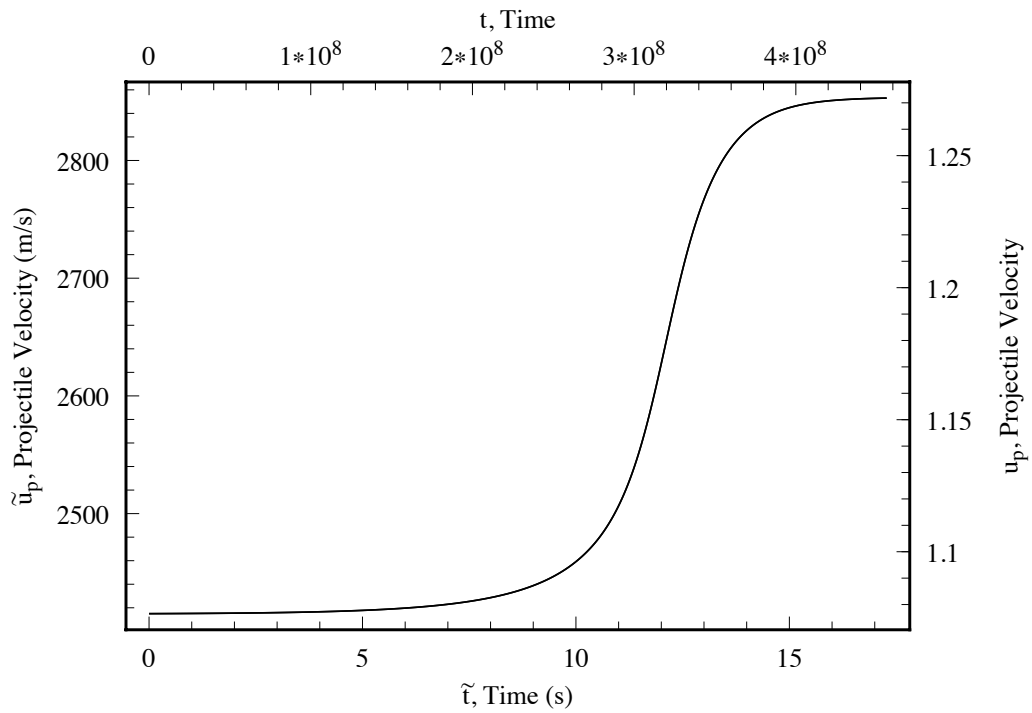


Figure 8.13: Projectile velocity as a function of time ($\tilde{q} = 1.014 \text{ MJ/kg}$, $\tilde{m} = 1.2108 \text{ kg/m}$).

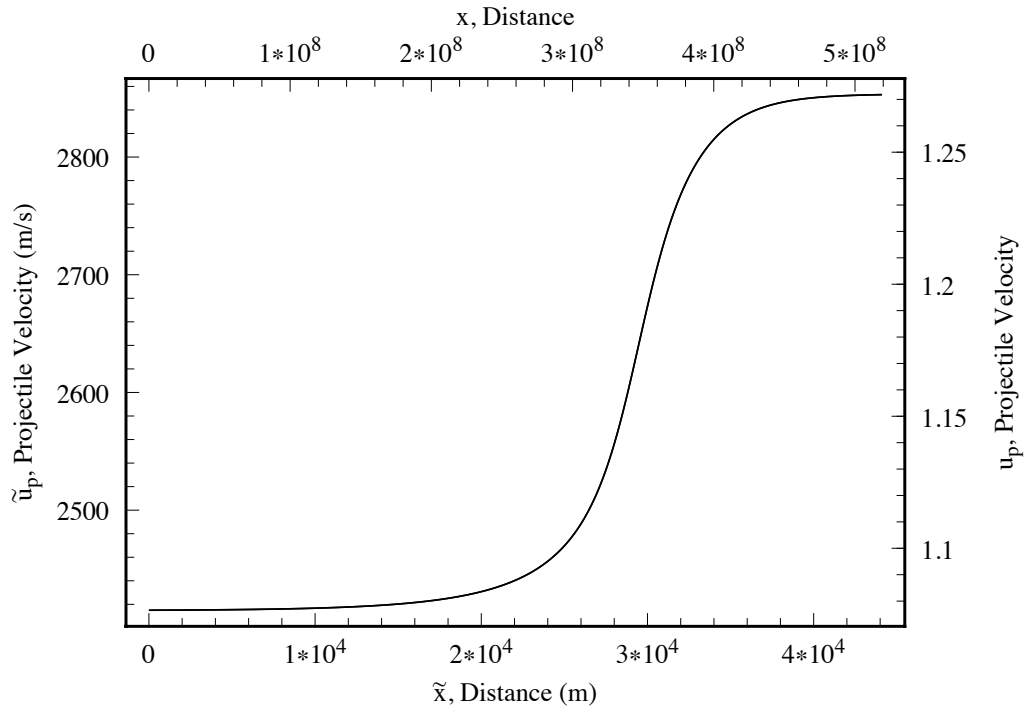


Figure 8.14: Projectile velocity as a function of position ($\tilde{q} = 1.014 \text{ MJ/kg}$, $\tilde{m} = 1.2108 \text{ kg/m}$).

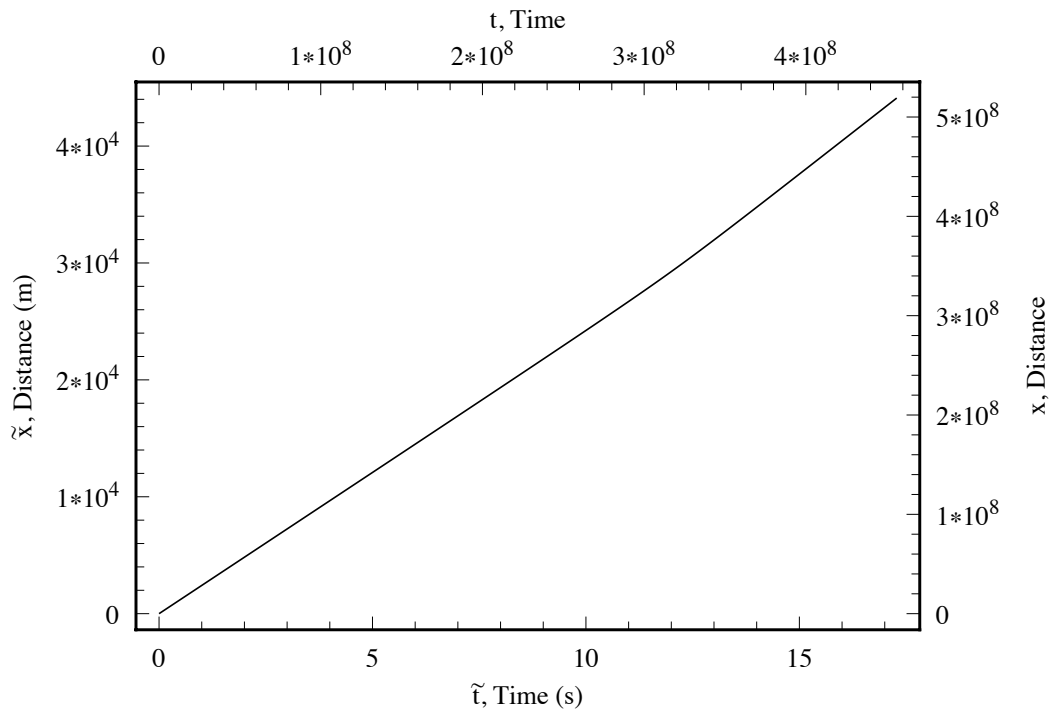


Figure 8.15: Projectile position as a function of time ($\tilde{q} = 1.014 \text{ MJ/kg}$, $\tilde{m} = 1.2108 \text{ kg/m}$).

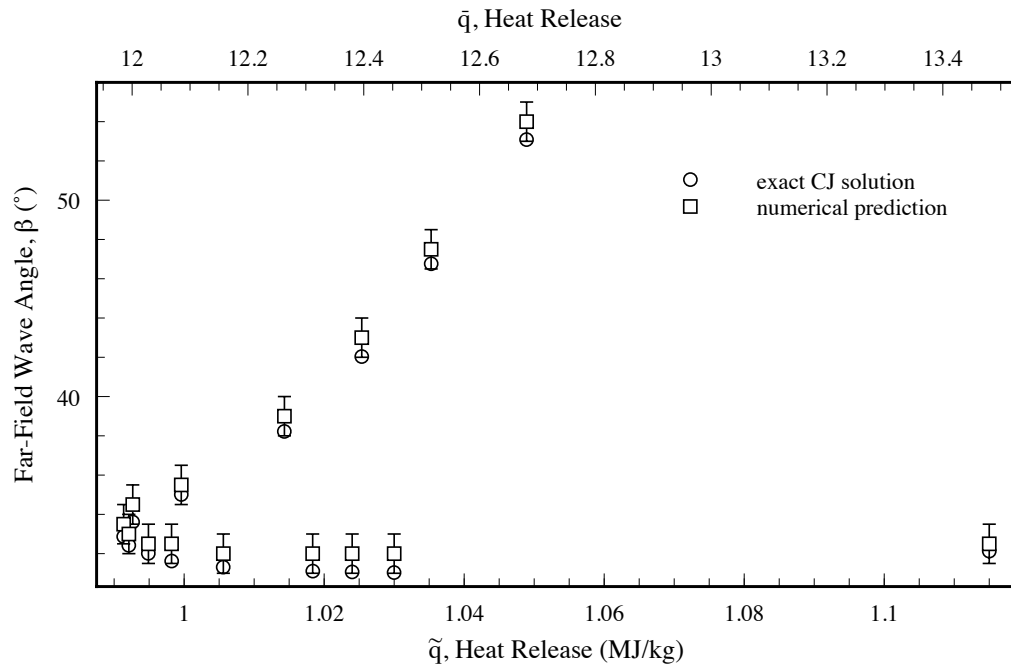


Figure 8.16: Comparison of exact and predicted detonation wave angles for steady speeds of propagation.

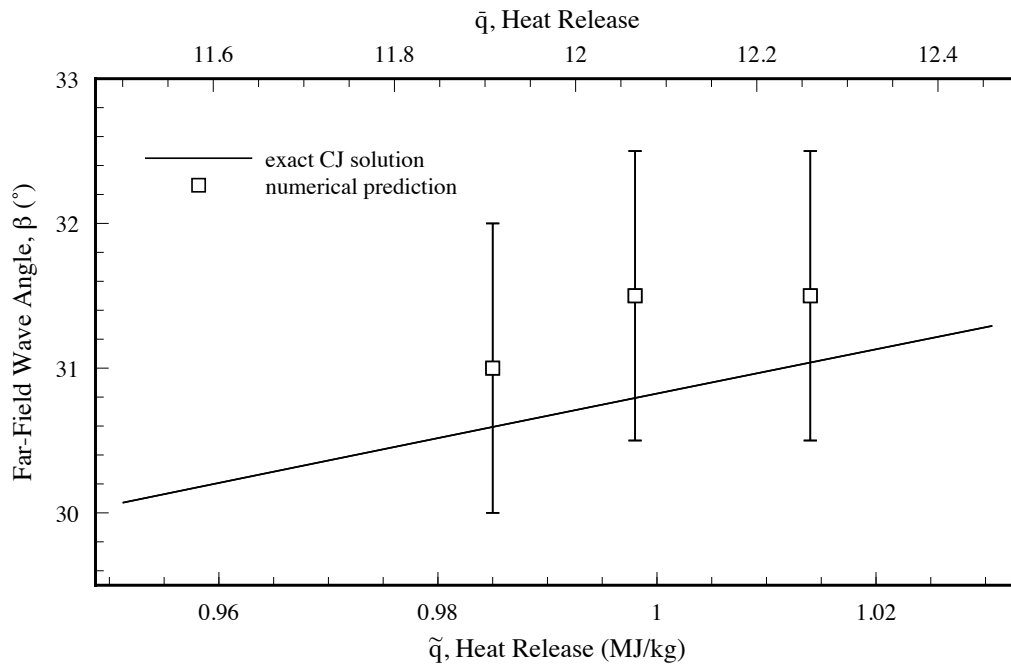


Figure 8.17: Exact and predicted detonation wave angles as a function of heat release ($\mathcal{M}_0 = 8.4$).

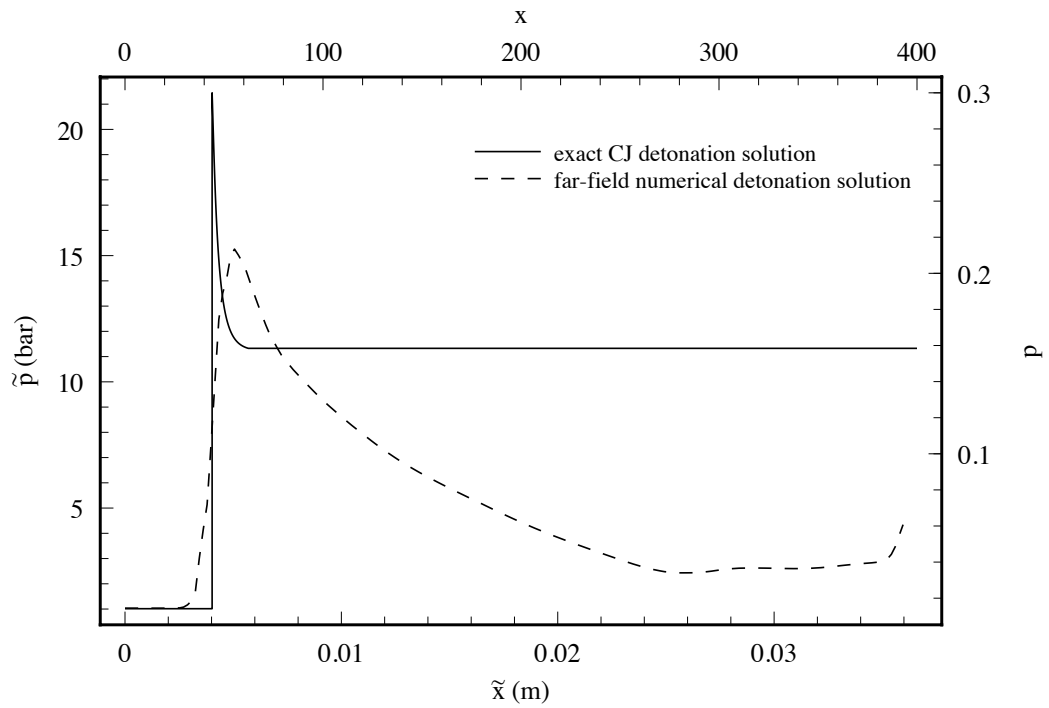


Figure 8.18: Comparison of exact CJ detonation pressure solution with far-field numerical pressure solution along a line normal to the oblique detonation wave ($\mathcal{M}_0 = 8.4$, $\tilde{q} = 0.985 \text{ MJ/kg}$).

Chapter 9

CONCLUSIONS

9.1 Unsteady Oblique Detonations

The results presented here suggest that straight shock, curved wall oblique detonations have a stability threshold slightly higher than their one-dimensional counterparts. For the parameters studied, the oblique detonation stability threshold was at a normal overdrive value of approximately $f_n = 1.77$, while the corresponding one-dimensional detonation threshold occurs at an overdrive of $f = 1.73$. The unstable oblique detonation solutions were characterized by the formation of one or more three-shock structures at the detonation front. These structures would move downward along the front, and eventually impact with the wedge surface. The passage of the three-shock resulted in a curved detonation front which propagated forward into the freestream until encountering the front computational boundary. The initial transient stages of the stable oblique detonations were also characterized by the three-shock structures; however, in those cases the disturbance would simply propagate out the top of the domain, and the detonation would return approximately to its initial steady state. The slightly higher stability threshold for the straight oblique detonation suggests that the initial “one-dimensional” instability, which decayed in one dimension, transitions to unstable two-dimensional structures for a small range of normal overdrive values. Above that range the two-dimensional structures that form are too weak to destabilize the detonation.

As suggested by the one-dimensional detonation stability results, an important parameter in the two-dimensional studies was the number of computational cells within the half reaction zone length of the initial steady detonation. The numerical algorithm was able to correctly predict one-dimensional instability with as few as 10 points per $L_{1/2}$. For all of the two-dimensional studies the points per $L_{1/2}$ was kept above this value. A second parameter that became important in the oblique detonation studies was the length in units of $L_{1/2}$ of the initial detonation within the computational domain. Defining the domains such that L_{ratio} was less than approximately 20-25 tended to result in incorrectly predicted stability. It is entirely possible that further extending the top of the domain to increase L_{ratio} past the values used could change the stability threshold observed. The instability in the solution at $f_n = 1.76$ occurred away from the upper boundary, however, so it is unlikely that the threshold would decrease. Finally, it is also possible that extending the computational domain horizontally to allow the curved detonation room to move forward could result in the eventual return of the detonation to a (straight) steady state.

As with any numerical study, particularly one involving the study of stability, there are a number of caveats concerning the above conclusions. The implementation of accurate wall boundary conditions along the bottom of the computational domain, and outflow boundary conditions along the top of the domain, presented the biggest challenge in this study; entirely satisfactory results were never obtained. It was deemed necessary to extend the computational grid somewhat in front of the wedge tip in order to allow the detonation room to adjust in this region; however the density and pressure spikes resulting from the grid discontinuity along the wall grid line are problematic. Additionally, downstream of the wedge tip and along the wall there was always some discrepancy between the exact solution and the numerical solution, even for those variables in which there was not an initial overshoot. This was partly a result of the spike in density and pressure, but was also observed to a lesser degree

in inert cases for which the grid discontinuity was much smaller. The net effect of these two problems was the creation of an unphysical numerical boundary layer in the region near the wall. As for the outflow condition along the top of the domain, there were obvious discrepancies that occurred when a shock exited through this boundary. Certainly the method used to determine the value of K was simplistic; however, there still remain fundamental problems in accurately estimating unknown physical quantities entering the domain along inflowing characteristics.

A second issue is the unknown effect on the observed detonation instability of the slight oscillations observed in the numerical solutions when compared to exact solutions. Certainly where issues of stability are concerned these are undesirable artifacts. Other researchers [21] have advocated adding very small amounts of artificial dissipation in higher-order Godunov schemes in order to eliminate such phenomena. This was avoided here due to the possible effects on the numerical prediction of instability. Without exact two-dimensional steady and unsteady solutions with which to compare, it is difficult to know which is the correct choice. A related issue is the nonunique choice of a limiter for the second order scheme. Two were compared here, but there are many more possibilities invented each year. Which limiter is used will certainly affect the solution, but, as with artificial dissipation, the correct choice is not necessarily evident.

The last issue is the numerical modelling of reactive flows, even those with very simple chemistry models. A recent report by Stewart and Bdzil notes some of the difficulties being encountered by researchers in this area [74]:

“In this case, the test problem was for the physically important case of a Chapman-Jouguet detonation, with large temperature sensitivity in the reaction rate. The attempts at solving the problem with different state-of-the-art codes seemed to have produced a wide variation in results. A main objective of the Workshop was to illustrate the variation in results obtained by the different methods. Our experience with CAVEAT and CMHOG [two different codes] shows that there can be profound qualitative and quantitative differences in the solutions obtained with different codes . . . The dramatic differences in the solutions obtained by these two codes, that had both passed many tests and that were being used with reasonable (numerical) parameters, is disconcerting at best. The binary difference in the qualitative behavior obtained with these codes points out that advanced algorithms must reduce accuracy errors wherever possible, if realistic engineering problems are to be solved accurately.”

According to the report, the equations being solved were exactly the reactive Euler equations with a one-step reaction model which were considered for this study.

Much work remains to be done in this area. Some of the above issues could be resolved by tracking the detonation front and using it as a boundary of the domain. This would allow exact Rankine-Hugoniot jump conditions to be used at the discontinuity, thus eliminating any Gibbs phenomena following the shock. Using an even higher order scheme could resolve some of the issues concerning the modelling of reactive flows. Perhaps most important of all, better numerical boundary conditions for finite domains need to be devised. In terms of more fully understanding oblique detonation stability, parametric studies of such parameters as the wall shape and initial detonation wave angle remain to be considered. Real devices, such as the ram accelerator, have walls instead of open boundaries, thus it would be useful to consider that effect as well.

9.2 Steady Propagation of a Ram Accelerator

This study has given indication of the importance of the interaction of kinetic length scales with geometric length scales in determining steady propagation velocities for high Mach number propulsion devices. The trends of our variation of net thrust with Mach number for fixed heat release are consistent with those of Ref. [10] and

[88]. Most importantly, the idea of using the heat release to vary the propagation speed, as shown in the bifurcation diagrams, has been demonstrated. In an ODWE environment, the equivalence ratio could presumably be varied to achieve this effect. Alternatively, one may be able to use the wedge angle as a bifurcation parameter to vary the propagation speed.

Due to the relatively low values of heat release considered in the study and the lack of cowl surfaces in the configuration, performance parameters, such as the maximum projectile acceleration, were much lower than what has been reported in the literature. Even with these limitations, however, large accelerations and changes in velocity were predicted, indicating the potential for this type of propulsion.

The relevance of the CJ state for this model and configuration has also been demonstrated. Since the CJ velocity is independent of the reaction kinetics, while the steady flight speed is dependent on the kinetics, the only relevance of the CJ Mach number was as an overly conservative lower bound for flight speeds. Additionally, the CJ oblique detonation angle was predicted in the far-field for cases with and without force balances, and thus does not have any particular relevance in the determination of the flight speed.

In the results presented here the oblique detonations exhibited no instabilities, which is consistent with the results of other researchers [10, 89, 88, 87]. However, there were as few as 3-4 cells within the reaction zones of the detonations. The one-dimensional ZND results indicate that this is too few to accurately capture the possible unsteady behavior of the detonation, though unsteady solutions were still found with relatively few cells in the reaction zone. Certainly it is of interest to consider more resolved solutions, as the unsteady detonation phenomena observed in Ch. 7 may appear.

For the future it would be useful to consider the case of dynamic stability. This would be accomplished by considering the equations of motion in an accelerating

frame of reference. The computed forces on the projectile would enter into the acceleration of the reference frame, which would then alter the inflow Mach number dynamically. Also as suggested in a review by Powers [59], it would be useful to study this problem in the context of other well-documented inert flows such as a Busemann biplane or flow over a thin airfoil.

Appendix A

PROGRAM LISTING

The following is a listing of the FORTRAN program and subroutines written to solve the reactive Euler equations using the second-order Roe scheme.

A.1 roe2d.f

```
program roe2d
c This program solves the 2D,unsteady, reactive Euler equations for an
c arbitrary mesh using the Roe scheme decomposed into one-dimensional
c sweeps (fractional step method). The chemistry is assumed frozen
c while the fluid mechanics is solved, and then the fluid mechanics
c is held frozen while the chemistry is solved (half steps).
c The Van Leer flux limiter is being used. Used 2nd order Runge Kutta
c time integration and symmetric splitting to obtain second-order
c temporal accuracy. Also, analytically integrated species equation
c following Colella, Majda and Roytburd.
c by Matt Grismer, 5/4/94
  implicit double precision (a-h,o-z)
  parameter (nx=100,ny=100,gamma=1.4d0)
  dimension xv(nx,ny),yv(nx,ny),xc(0:nx,0:ny),yc(0:nx,0:ny),
+          xixc(0:nx,0:ny),xiyc(0:nx,0:ny),etaxc(0:nx,0:ny),
+          etayc(0:nx,0:ny),xixii(nx,ny-1),xiyii(nx,ny-1),
+          etaxij(nx-1,ny),etayij(nx-1,ny)
  double precision jacc(0:nx,0:ny),jacii(nx,ny-1),jacij(nx-1,ny),
+          m0,m02,jac
  dimension rhoc(0:nx,0:ny),pc(0:nx,0:ny),uc(0:nx,0:ny),
+          vc(0:nx,0:ny),bigec(0:nx,0:ny),bighc(0:nx,0:ny),
+          q(0:nx,0:ny,5),fflux(nx,ny-1,5),gflux(nx-1,ny,5),
+          y2c(0:nx,0:ny),qbar(0:nx,0:ny,5),
+          eigvlx(nx,5),eigvcx(nx,5,5),dwx(nx,5),
+          eigvly(ny,5),eigvcy(ny,5,5),dwy(ny,5)
  dimension rhoold(nx-1,ny-1),pold(nx-1,ny-1),uold(nx-1,ny-1),
+          vold(nx-1,ny-1),beold(nx-1,ny-1)
  logical rerun,dimens,intrvl,topwal
  character*20 intnam
  common gam1,gam12,speed,uin,vin,pin,rhoin,bigen,bighn
```

```

        common /chemistry/ y2in,qheat,theta,m02
c
200  format(2(e25.18,2x),e25.18)
220  format(4(e14.7,1x),e14.7)
c
        gam1 = gamma - 1.0d0
        gam12 = gam1/2.0d0
c Read in data file or start from scratch
        rerun = .true.
c even number better
        ncount = 500
        dimens = .true.
        topwal = .false.
        init = 1
c choose whether to write data files at intervals, after how many steps
c choose even number of intervals
        intrvl = .false.
        intcnt = 500
c
c Read dimensional grid file for cell vertex locations
        open(12,file='grid.dat',status='old',form='unformatted')
        do 90 i=1,nx
            do 90 j=1,ny
                read(12) xv(i,j),yv(i,j)
90    continue
        close(12)
c
c dimensional inflow conditions
        rho0 = 1.225d0
        p0 = 101325.0d0
        m0 = 4.0d0
        u0 = m0*sqrt(gamma*p0/rho0)
        v0 = 0.0d0
        bige0 = p0/(rho0*gam1) + 0.5d0*(u0**2 + v0**2)
        bigh0 = bige0 + p0/rho0
c (dimensional) arrhenius prefactor, activation energy, and heat
c release (for chemistry)
        bigb0 = 1.0d6
        qheat0 = 0.0d0
        acte = 4135714.28571d0
c nondimensionalize (write out just to indicate method of nondim.)
        m02 = m0*m0
        const1 = m0*sqrt(p0/rho0)
        const2 = m02*p0
        rhoin = rho0/rho0
        pin = p0/const2
        uin = u0/const1
        vin = v0/const1
        bigen = bige0*rho0/const2
        bighn = bigh0*rho0/const2
        y2in = 0.0d0
        qheat = qheat0*rho0/const2
        theta = acte*rho0/const2
c nondimensionalize grid vertices

```

```

do 100 j=1,ny
  do 100 i=1,nx
    xv(i,j) = bigb0*xv(i,j)/const1
    yv(i,j) = bigb0*yv(i,j)/const1
100  continue
c
c calculate cell centers
  call center(nx,ny,xv,yv,xc,yc)
c determine K factor from Poinot and Lele for subsonic outflow
  bigk = 0.025d0
c write out centers data file
  open(15,file='centers.dat',status='unknown')
  write(15,*) nx-1,ny-1
  if(dimens) then
    write(15,200) ((xc(i,j)*const1/bigb0,i=1,nx-1),j=1,ny-1)
    write(15,200) ((yc(i,j)*const1/bigb0,i=1,nx-1),j=1,ny-1)
  close(15)
  else
    write(15,200) ((xc(i,j),i=1,nx-1),j=1,ny-1)
    write(15,200) ((yc(i,j),i=1,nx-1),j=1,ny-1)
  close(15)
  endif
c calculate ghost cell centers and associated metrics
  call ghost(nx,ny,xv,yv,xc,yc,xixc,xiyc,etaxc,etayc,jacc)
c calculate metrics at centers
  call metrcc(nx,ny,xv,yv,xixc,xiyc,etaxc,etayc,jacc)
c calculate metrics at interfaces in i an j directions, respectively
  call metrci(nx,ny,xv,yv,xc,yc,xixii,xiyii,jacii)
  call metrcj(nx,ny,xv,yv,xc,yc,etaxij,etayij,jacij)
c
c Read in previous run or initialize grid
  if(rerun) then
    open(16,file='roe2d.dat',status='old')
    read(16,*) junk1,junk2
    if((junk1+1).ne.nx.or.(junk2+1).ne.ny) pause 'Problem
+   w/dimensions.'
    read(16,*) ((rhoc(i,j),i=1,nx-1),j=1,ny-1)
    read(16,*) ((uc(i,j),i=1,nx-1),j=1,ny-1)
    read(16,*) ((vc(i,j),i=1,nx-1),j=1,ny-1)
    read(16,*) ((bigec(i,j),i=1,nx-1),j=1,ny-1)
    read(16,*) ((pc(i,j),i=1,nx-1),j=1,ny-1)
    read(16,*) ((y2c(i,j),i=1,nx-1),j=1,ny-1)
    read(16,*) nstep,t
    n0 = nstep
  close(16)
c dimensional data must be nondimensionalized
  if(dimens) then
    do 103 j=1,ny-1
      do 103 i=1,nx-1
        rhoc(i,j) = rhoc(i,j)/rho0
        uc(i,j) = uc(i,j)/const1
        vc(i,j) = vc(i,j)/const1
        bigec(i,j) = bigec(i,j)*rho0/const2
        pc(i,j) = pc(i,j)/const2

```

```

103     continue
      endif
      do 105 j=1,ny-1
        do 105 i=1,nx-1
          bighc(i,j) = bigec(i,j) + pc(i,j)/rhoc(i,j)
105     continue
      else
c Initialize grid
c Choose from among different ways with init variable
c init = 1, set constant state throughout domain
      if(init.eq.1) then
        do 110 j=1,ny-1
          do 110 i=1,nx-1
c primitive variables
          rhoc(i,j) = rhoin
          pc(i,j) = pin
          uc(i,j) = uin
          vc(i,j) = vin
          bigec(i,j) = bigen
          bighc(i,j) = bighn
          y2c(i,j) = y2in
110     continue
c init = 2, set zero velocity, constant state throughout domain
      elseif(init.eq.2) then
        do 111 j=1,ny-1
          do 111 i=1,nx-1
          rhoc(i,j) = rhoin
          pc(i,j) = pin
          uc(i,j) = 0.0d0
          vc(i,j) = 0.0d0
          bigec(i,j) = pin/(rhoin*gam1)
          bighc(i,j) = bigec(i,j) + pin/rhoin
          y2c(i,j) = 0.0d0
111     continue
c init = 3 or more, set linearly varying u initial state
      else
        dm = (m0 - 1.01d0)/(nx-1)
        do 112 j=1,ny-1
          do 112 i=1,nx-1
          rhoc(i,j) = rhoin
          pc(i,j) = pin
          xm = m0 - dm*dble(i)
          uc(i,j) = sqrt(gamma*pin*xm*xm/rhoin - vin*vin)
          vc(i,j) = vin
          bigec(i,j) = pin/(rhoin*gam1) + 0.5d0*(uc(i,j)*uc(i,j) +
+          vc(i,j)*vc(i,j))
          bighc(i,j) = bigec(i,j) + pin/rhoin
          y2c(i,j) = 0.0d0
112     continue
        endif
        t = 0.0d0
        nstep = 0
        n0 = 0
      endif

```

```

c
c dependent variable vector
  do 115 j=1,ny-1
    do 115 i=1,nx-1
      jac = jacc(i,j)
      rho = rhoc(i,j)
      q(i,j,1) = rho/jac
      q(i,j,2) = rho*uc(i,j)/jac
      q(i,j,3) = rho*vc(i,j)/jac
      q(i,j,4) = rho*bigec(i,j)/jac
      q(i,j,5) = rho*y2c(i,j)/jac
115    continue
c
c Set inflow flux based upon inflow quantities
c assumes supersonic flow, so all characteristics point in
  do 120 j=1,ny-1
    fflux(1,j,1) = rhoin*(xixc(0,j)*uin + xiyc(0,j)*vin)
    +
    /jacc(0,j)
    fflux(1,j,2) = fflux(1,j,1)*uin + xixc(0,j)*pin/jacc(0,j)
    fflux(1,j,3) = fflux(1,j,1)*vin + xiyc(0,j)*pin/jacc(0,j)
    fflux(1,j,4) = fflux(1,j,1)*bighn
    fflux(1,j,5) = fflux(1,j,1)*y2in
c Need ghost cells to contain inflow quantities for higher order
c terms in flxxi subroutine
  rhoc(0,j) = rhoin
  pc(0,j) = pin
  uc(0,j) = uin
  vc(0,j) = vin
  bigec(0,j) = bigen
  bighc(0,j) = bighn
  y2c(0,j) = y2in
c
  jac = jacc(0,j)
  rho = rhoc(0,j)
  q(0,j,1) = rho/jac
  q(0,j,2) = rho*uc(0,j)/jac
  q(0,j,3) = rho*vc(0,j)/jac
  q(0,j,4) = rho*bigec(0,j)/jac
  q(0,j,5) = rho*y2c(0,j)/jac
120  continue
c make sure unused outflow interface quantities are zero
  do 300 k=1,5
    eigvlx(nx,k) = 0.0d0
    dwx(nx,k) = 0.0d0
c
    eigvly(1,k) = 0.0d0
    dwy(1,k) = 0.0d0
    eigvly(ny,k) = 0.0d0
    dwy(ny,k) = 0.0d0
300  continue
cccccccccccccccccccccccccccccccccccc
c Begin main iteration loop
c Use fractional stepping to get entire solution
c Use seond-order Runge Kutta time integration for each step,

```

```

c and then combine steps symmetrically to maintain second-order
c accuracy with splitting. One complete step will advance solution
c 2*dt in time.
cccccccccccccccccccccccccccccccccccccccc
      total = dble((nx-1)*(ny-1))
c keep old values of variables for calculating residuals
50   do 122 j=1,ny-1
      do 122 i=1,nx-1
        rhoold(i,j) = rhoc(i,j)
        uold(i,j) = uc(i,j)
        vold(i,j) = vc(i,j)
        polold(i,j) = pc(i,j)
        beold(i,j) = bigec(i,j)
122   continue
c get timestep
      call timest(nx,ny,rhoc,pc,uc,vc,xixc,xiyc,etaxc,etayc,dt)
      dt2 = 0.5d0*dt
cccccccccccc
c Fractional step in xi direction
cccccccccccc
c Integrate forward in time dt/2
c determine fluxes at xi interfaces using Roe scheme
      call flxxi(nx,ny,rhoc,pc,uc,vc,bighc,y2c,q,xixc,xiyc,jacc,
+             xixii,xiyii,eigvlx,eigvcx,dwx,fflux)
c update outflow boundary cells using Poinot and Lele idea
c use 2nd order runge kutta here also, so advance half step
      call boundx(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,fflux,
+             xixc,xiyc,jacc,dt2,.true.)
c determine estimate qbar after half timestep
c counting cell centers
      do 130 k=1,5
        do 130 j=1,ny-1
          do 130 i=1,nx-2
            qbar(i,j,k) = q(i,j,k)-dt2*(fflux(i+1,j,k) - fflux(i,j,k))
130      continue
c decode qbar into primitive variables
      call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,qbar,jacc)
c determine new fluxes based upon qbar
      call flxxi(nx,ny,rhoc,pc,uc,vc,bighc,y2c,qbar,xixc,xiyc,jacc,
+             xixii,xiyii,eigvlx,eigvcx,dwx,fflux)
c determine new q based up intermediate qbar values at boundary
      call boundx(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,fflux,
+             xixc,xiyc,jacc,dt,.false.)
c use qbar fluxes to determine new q after dt
c counting cell centers
      do 135 k=1,5
        do 135 j=1,ny-1
          do 135 i=1,nx-2
            q(i,j,k) = q(i,j,k)-dt*(fflux(i+1,j,k) - fflux(i,j,k))
135      continue
c decode q into primitive variables for next step
      call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,q,jacc)
cccccccccccccccc
c fractional step in eta direction

```

```

cccccccccccccccc
c determine fluxes at eta interfaces using Roe scheme
    call flxeta(nx,ny,rhoc,pc,uc,vc,bighc,y2c,q,etaxc,etayc,
+         jacc,etaxij,etayij,jacij,eigvly,eigvcy,dwy,gflux,topwal)
c
c If topwal=true, then use wall condition at top of domain
c (inside flxeta), else use outflow condition of Poinot and Lele
    if(topwal) then
c wall boundary conditions handled inside flxeta
c determine intermediate qbar over dt/2
    do 140 k=1,5
        do 140 j=1,ny-1
            do 140 i=1,nx-1
                qbar(i,j,k) = q(i,j,k)-dt2*(gflux(i,j+1,k) - gflux(i,j,k))
140            continue
c
c decode qbar into primitive variables
    call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,qbar,jacc)
c determine new fluxes based upon qbar
    call flxeta(nx,ny,rhoc,pc,uc,vc,bighc,y2c,qbar,etaxc,etayc,
+         jacc,etaxij,etayij,jacij,eigvly,eigvcy,dwy,gflux,topwal)
c determine new q after dt using intermediate fluxes from qbar
    do 142 k=1,5
        do 142 j=1,ny-1
            do 142 i=1,nx-1
                q(i,j,k) = q(i,j,k) - dt*(gflux(i,j+1,k) - gflux(i,j,k))
142            continue
cccccccc
        else
cccccccc
c use Poinot and Lele outflow condition at top of domain
c determine intermediate qbar along top of domain
    call boundn(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,gflux,
+         etaxc,etayc,jacc,etaxij,etayij,jacij,xixc,xiyc,
+         dt2,.true.)
c determine intermediate values qbar over dt/2
    do 145 k=1,5
        do 145 j=1,ny-2
            do 145 i=1,nx-1
                qbar(i,j,k) = q(i,j,k)-dt2*(gflux(i,j+1,k) - gflux(i,j,k))
145            continue
c
c decode qbar into primitive variables
    call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,qbar,jacc)
c calculate intermediate flux values based upon qbar
    call flxeta(nx,ny,rhoc,pc,uc,vc,bighc,y2c,qbar,etaxc,etayc,
+         jacc,etaxij,etayij,jacij,eigvly,eigvcy,dwy,gflux,topwal)
c Determine new q along top of domain
    call boundn(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,gflux,
+         etaxc,etayc,jacc,etaxij,etayij,jacij,xixc,xiyc,
+         dt,.false.)
c determine new q using intermediate flux values
    do 147 k=1,5
        do 147 j=1,ny-2

```

```

                do 147 i=1,nx-1
                    q(i,j,k) = q(i,j,k) - dt*(gflux(i,j+1,k) - gflux(i,j,k))
147            continue
                endif
c decode into new primitive variables for next fractional step
                call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,q,jacc)
cccccccccccccccc
c fractional step for chemistry (species equation)
cccccccccccccccc
                do 150 j=1,ny-1
                    do 150 i=1,nx-1
                        rho = rhoc(i,j)
                        y2c(i,j) = 1.0d0+(y2c(i,j)-1.0d0)*exp(-dt*
+                            exp(-theta*rho/pc(i,j)))
c update primitive variables, but only need to do p and H because
c all other variables stay fixed (except Y2 above)
                        pc(i,j) = rho*gam1*(bigec(i,j) + y2c(i,j)*qheat-0.5d0*
+                            (uc(i,j)*uc(i,j) + vc(i,j)*vc(i,j)))
                        bighc(i,j) = bigec(i,j) + pc(i,j)/rho
                        q(i,j,5) = rho*y2c(i,j)/jacc(i,j)
150            continue
cccccccccccccccc
c First timestep done, but reverse to get second order accuracy
c for splitting. do chemistry first and then flow
c When done have advanced two delta t
cccccccccccccccc
                do 155 j=1,ny-1
                    do 155 i=1,nx-1
                        rho = rhoc(i,j)
                        y2c(i,j) = 1.0d0+(y2c(i,j)-1.0d0)*exp(-dt*
+                            exp(-theta*rho/pc(i,j)))
                        pc(i,j) = rho*gam1*(bigec(i,j) + y2c(i,j)*qheat-0.5d0*
+                            (uc(i,j)*uc(i,j) + vc(i,j)*vc(i,j)))
                        bighc(i,j) = bigec(i,j) + pc(i,j)/rho
                        q(i,j,5) = rho*y2c(i,j)/jacc(i,j)
155            continue
cccccccccccccccc
c fractional step in eta direction
cccccccccccccccc
                call flxeta(nx,ny,rhoc,pc,uc,vc,bighc,y2c,q,etaxc,etayc,
+                    jacc,etaxij,etayij,jacij,eigvly,eigvcy,dwy,gflux,topwal)
                if(topwal) then
                    do 160 k=1,5
                        do 160 j=1,ny-1
                            do 160 i=1,nx-1
                                qbar(i,j,k) = q(i,j,k)-dt2*(gflux(i,j+1,k) - gflux(i,j,k))
160            continue
                call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,qbar,jacc)
                call flxeta(nx,ny,rhoc,pc,uc,vc,bighc,y2c,qbar,etaxc,etayc,
+                    jacc,etaxij,etayij,jacij,eigvly,eigvcy,dwy,gflux,topwal)
                    do 165 k=1,5
                        do 165 j=1,ny-1
                            do 165 i=1,nx-1
                                q(i,j,k) = q(i,j,k) - dt*(gflux(i,j+1,k) - gflux(i,j,k))

```



```

165     continue
cccccccc
      else
cccccccc
      call boundn(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,gflux,
+          etaxc,etayc,jacc,etaxij,etayij,jacij,xixc,xiyc,
+          dt2,.true.)
      do 175 k=1,5
      do 175 j=1,ny-2
      do 175 i=1,nx-1
      qbar(i,j,k) = q(i,j,k)-dt2*(gflux(i,j+1,k) - gflux(i,j,k))
175     continue
      call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,qbar,jacc)
      call flxeta(nx,ny,rhoc,pc,uc,vc,bighc,y2c,qbar,etaxc,etayc,
+          jacc,etaxij,etayij,jacij,eigvly,eigvcy,dwy,gflux,topwal)
      call boundn(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,gflux,
+          etaxc,etayc,jacc,etaxij,etayij,jacij,xixc,xiyc,
+          dt,.false.)
      do 180 k=1,5
      do 180 j=1,ny-2
      do 180 i=1,nx-1
      q(i,j,k) = q(i,j,k) - dt*(gflux(i,j+1,k) - gflux(i,j,k))
180     continue
      endif
      call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,q,jacc)
cccccccc
c Fractional step in xi direction
cccccccc
      call flxxi(nx,ny,rhoc,pc,uc,vc,bighc,y2c,q,xixc,xiyc,jacc,
+          xixii,xiyii,eigvlx,eigvcx,dwx,fflux)
      call boundx(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,fflux,
+          xixc,xiyc,jacc,dt2,.true.)
      do 190 k=1,5
      do 190 j=1,ny-1
      do 190 i=1,nx-2
      qbar(i,j,k) = q(i,j,k)-dt2*(fflux(i+1,j,k) - fflux(i,j,k))
190     continue
      call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,qbar,jacc)
      call flxxi(nx,ny,rhoc,pc,uc,vc,bighc,y2c,qbar,xixc,xiyc,jacc,
+          xixii,xiyii,eigvlx,eigvcx,dwx,fflux)
      call boundx(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,fflux,
+          xixc,xiyc,jacc,dt,.false.)
      do 195 k=1,5
      do 195 j=1,ny-1
      do 195 i=1,nx-2
      q(i,j,k) = q(i,j,k)-dt*(fflux(i+1,j,k) - fflux(i,j,k))
195     continue
      call decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,q,jacc)
c determine residuals for output
c determine average difference of variables over grid
      bigdr = 0.0d0
      bigdu = 0.0d0
      bigdv = 0.0d0
      bigdp = 0.0d0

```

```

bigde = 0.0d0
do 170 j=1,ny-1
  do 170 i=1,nx-1
    drho = abs(rhoc(i,j) - rhoold(i,j))
    du = abs(uc(i,j) - uold(i,j))
    dv = abs(vc(i,j) - vold(i,j))
    dp = abs(pc(i,j) - pold(i,j))
    de = abs(bigec(i,j) - beold(i,j))
  c
    bigdr = bigdr + drho
    bigdu = bigdu + du
    bigdv = bigdv + dv
    bigdp = bigdp + dp
    bigde = bigde + de
170  continue
    bigdr = bigdr/total
    bigdu = bigdu/total
    bigdv = bigdv/total
    bigdp = bigdp/total
    bigde = bigde/total
  c
    nstep = nstep + 2
    t = t + 2.0d0*dt
  c
c write out data at intervals
  if(intrvl.and.mod(nstep,intcnt).eq.0) then
c figure out data file name
    jcnt = nstep/intcnt
    if(jcnt.lt.10) then
      intnam = 'roe2d_int'//char(48+jcnt)//'.dat'
    elseif(jcnt.lt.100) then
      jcnt10 = jcnt/10
      jcnt1 = jcnt - jcnt10*10
      intnam = 'roe2d_int'//char(48+jcnt10)//char(48+jcnt1)//'.dat'
    elseif(jcnt.lt.1000) then
      jcnt100 = jcnt/100
      jcnt10 = (jcnt - jcnt100*100)/10
      jcnt1 = (jcnt - jcnt100*100 - jcnt10*10)
      intnam = 'roe2d_int'//char(48+jcnt100)//char(48+jcnt10)
+      //char(48+jcnt1)//'.dat'
    endif
c write data
    open(19,file=intnam,status='new')
    write(19,*) nx-1,ny-1
    if(dimens) then
      write(19,220) ((rhoc(i,j)*rho0,i=1,nx-1),j=1,ny-1)
      write(19,220) ((uc(i,j)*const1,i=1,nx-1),j=1,ny-1)
      write(19,220) ((vc(i,j)*const1,i=1,nx-1),j=1,ny-1)
      write(19,220) ((bigec(i,j)*const2/rho0,i=1,nx-1),j=1,ny-1)
      write(19,220) ((pc(i,j)*const2,i=1,nx-1),j=1,ny-1)
      write(19,220) ((y2c(i,j),i=1,nx-1),j=1,ny-1)
    else
      write(19,220) ((rhoc(i,j),i=1,nx-1),j=1,ny-1)
      write(19,220) ((uc(i,j),i=1,nx-1),j=1,ny-1)

```

```

        write(19,220) ((vc(i,j),i=1,nx-1),j=1,ny-1)
        write(19,220) ((bigec(i,j),i=1,nx-1),j=1,ny-1)
        write(19,220) ((pc(i,j),i=1,nx-1),j=1,ny-1)
        write(19,220) ((y2c(i,j),i=1,nx-1),j=1,ny-1)
    endif
    write(19,*) nstep,t
    close(19)
endif
c
if(nstep-n0.ge.ncount) then
    write(*,'(i5,2x,5(e12.5,2x))') nstep,bigdr,bigdu,bigdv,
+                                     bigdp,bigde
    open(14,file='roe2d.dat',status='unknown')
    write(14,*) nx-1,ny-1
    if(dimens) then
        write(14,200) ((rhoc(i,j)*rho0,i=1,nx-1),j=1,ny-1)
        write(14,200) ((uc(i,j)*const1,i=1,nx-1),j=1,ny-1)
        write(14,200) ((vc(i,j)*const1,i=1,nx-1),j=1,ny-1)
        write(14,200) ((bigec(i,j)*const2/rho0,i=1,nx-1),j=1,ny-1)
        write(14,200) ((pc(i,j)*const2,i=1,nx-1),j=1,ny-1)
        write(14,200) ((y2c(i,j),i=1,nx-1),j=1,ny-1)
    else
        write(14,200) ((rhoc(i,j),i=1,nx-1),j=1,ny-1)
        write(14,200) ((uc(i,j),i=1,nx-1),j=1,ny-1)
        write(14,200) ((vc(i,j),i=1,nx-1),j=1,ny-1)
        write(14,200) ((bigec(i,j),i=1,nx-1),j=1,ny-1)
        write(14,200) ((pc(i,j),i=1,nx-1),j=1,ny-1)
        write(14,200) ((y2c(i,j),i=1,nx-1),j=1,ny-1)
    endif
    write(14,*) nstep,t
    close(14)
else
    write(*,'(i5,2x,5(e12.5,2x))') nstep,bigdr,bigdu,bigdv,
+                                     bigdp,bigde
    goto 50
endif
end

```

A.2 center.f

```

subroutine center(nx,ny,xv,yv,xc,yc)
c calculates center of cell using cell vertices. Simplistic approach:
c find intersection of lines connecting opposite face centers.
    implicit double precision (a-h,o-z)
    dimension xv(nx,ny),yv(nx,ny),xc(0:nx,0:ny),yc(0:nx,0:ny)
    double precision m1,m2
c
c xv and yv are the cell vertices, while xc and yc are the center
c
c i,j+1 ----4----- i+1,j+1
c   |                   |
c   1                   2

```

```

c      |          |
c  i,j ----3----- i+1,j
c
      do 100 j=1,ny-1
      do 100 i=1,nx-1
c face 1
      xc1 = 0.5d0*(xv(i,j) + xv(i,j+1))
      yc1 = 0.5d0*(yv(i,j) + yv(i,j+1))
c face 2
      xc2 = 0.5d0*(xv(i+1,j) + xv(i+1,j+1))
      yc2 = 0.5d0*(yv(i+1,j) + yv(i+1,j+1))
c face 3
      xc3 = 0.5d0*(xv(i,j) + xv(i+1,j))
      yc3 = 0.5d0*(yv(i,j) + yv(i+1,j))
c face 4
      xc4 = 0.5d0*(xv(i,j+1) + xv(i+1,j+1))
      yc4 = 0.5d0*(yv(i,j+1) + yv(i+1,j+1))
c line connecting centers of face 1 and 2
c y = m*x + b
      m1 = (yc2 - yc1)/(xc2 -xc1)
      b1 = yc2 - m1*xc2
c line connecting centers of face 3 and 4
c x = m*y + b (avoids problem with vertical lines)
      m2 = (xc4 - xc3)/(yc4 - yc3)
      b2 = xc4 - m2*yc4
c intersection = centers
      yc(i,j) = (m1*b2 + b1)/(1.0d0 - m1*m2)
      xc(i,j) = m2*yc(i,j) + b2
100  continue
      return
      end

```

A.3 ghost.f

```

      subroutine ghost(nx,ny,xv,yv,xc,yc,xixc,xiyc,etaxc,etayc,jacc)
c Determines locations of vertices and cell centers of "ghost" cells.
c These ghost cells will be used to (hopefully) smooth the metrics
c near the boundaries of the domain. Ghost cells also postulated for
c determining the proper boundary conditions at walls, etc.
c This routine kind of assumes that the grid lines are orthogonal at
c the boundaries. This is not really a restriction, since the grid
c lines should be nearly orthogonal everywhere for the best performance
c of the code.
      implicit double precision (a-h,o-z)
      dimension xv(nx,ny),yv(nx,ny),xc(0:nx,0:ny),yc(0:nx,0:ny),
+             xixc(0:nx,0:ny),xiyc(0:nx,0:ny),etaxc(0:nx,0:ny),
+             etayc(0:nx,0:ny)
      double precision jacc(0:nx,0:ny),jacob
c
c Determine vertices for ghost cells before i=1 and after i=nx
c Do this by finding the line that connects i=1 vertices with
c i=2 vertices, and then extending it behind i=1. Similarly for

```

```

c i=nx. Extend behind the cell based upon the percent change between
c the first two real vertices.
c first ghost vertice
c i=0
      slope1 = (yv(2,1) - yv(1,1))/(xv(2,1) - xv(1,1))
      b1 = yv(1,1) - slope1*xv(1,1)
      gxv1 = xv(1,1) - (xv(2,1)-xv(1,1))*2/(xv(3,1)-xv(2,1))
      gyv1 = slope1*gxv1 + b1
c
      do 100 j=2,ny
c second ghost vertice
      slope2 = (yv(2,j) - yv(1,j))/(xv(2,j) - xv(1,j))
      b2 = yv(1,j) - slope2*xv(1,j)
      gxv2 = xv(1,j) - (xv(2,j)-xv(1,j))*2/(xv(3,j)-xv(2,j))
      gyv2 = slope2*gxv2 + b2
c get center of cell by finding intersection of lines connecting
c interface centers, as done in center subroutine
      xc1 = 0.5d0*(gxv1 + gxv2)
      yc1 = 0.5d0*(gyv1 + gyv2)
      xc2 = 0.5d0*(xv(1,j) + xv(1,j-1))
      yc2 = 0.5d0*(yv(1,j) + yv(1,j-1))
      xc3 = 0.5d0*(gxv1 + xv(1,j-1))
      yc3 = 0.5d0*(gyv1 + yv(1,j-1))
      xc4 = 0.5d0*(gxv2 + xv(1,j))
      yc4 = 0.5d0*(gyv2 + yv(1,j))
      slope1 = (yc2 - yc1)/(xc2 - xc1)
      slope2 = (xc4 - xc3)/(yc4 - yc3)
      b1 = yc2 - slope1*xc2
      b2 = xc4 - slope2*yc4
      yc(0,j-1) = (slope1*b2 + b1)/(1.0d0 - slope1*slope2)
      xc(0,j-1) = slope2*yc(0,j-1) + b2
c determine ghost center metric quantities (needed for BC's)
c to do this find grid line centers ("interface" centers) and finite
c difference
      xxi = 0.5d0*(xv(1,j-1)+xv(1,j)) - 0.5d0*(gxv1+gxv2)
      yxi = 0.5d0*(yv(1,j-1)+yv(1,j)) - 0.5d0*(gyv1+gyv2)
      xeta = 0.5d0*(gxv2+xv(1,j)) - 0.5d0*(gxv1+xv(1,j-1))
      yeta = 0.5d0*(gyv2+yv(1,j)) - 0.5d0*(gyv1+yv(1,j-1))
      jacob = 1.0d0/(xxi*yeta - xeta*yxi)
      jacc(0,j-1) = jacob
      xixc(0,j-1) = yeta*jacob
      xiyc(0,j-1) = -xeta*jacob
      etaxc(0,j-1) = -yxi*jacob
      etayc(0,j-1) = xxi*jacob
c get ready for next loop through
      gxv1 = gxv2
      gyv1 = gyv2
100 continue
c
c i=nx
      slope1 = (yv(nx,1) - yv(nx-1,1))/(xv(nx,1) - xv(nx-1,1))
      b1 = yv(nx,1) - slope1*xv(nx,1)
      gxv1 = xv(nx,1) + (xv(nx,1)-xv(nx-1,1))*2
+      / (xv(nx-1,1)-xv(nx-2,1))

```

```

        gyv1 = slope1*gxv1 + b1
c
    do 110 j=2,ny
        slope2 = (yv(nx,j) - yv(nx-1,j))/(xv(nx,j)-xv(nx-1,j))
        b2 = yv(nx,j) - slope2*xv(nx,j)
        gxv2 = xv(nx,j) + (xv(nx,j)-xv(nx-1,j))*2
+      / (xv(nx-1,j)-xv(nx-2,j))
        gyv2 = slope2*gxv2 + b2
c
c get center of cell by finding intersection of lines connecting
c interface centers, as done in center subroutine
        xc2 = 0.5d0*(gxv1 + gxv2)
        yc2 = 0.5d0*(gyv1 + gyv2)
        xc1 = 0.5d0*(xv(nx,j) + xv(nx,j-1))
        yc1 = 0.5d0*(yv(nx,j) + yv(nx,j-1))
        xc3 = 0.5d0*(gxv1 + xv(nx,j-1))
        yc3 = 0.5d0*(gyv1 + yv(nx,j-1))
        xc4 = 0.5d0*(gxv2 + xv(nx,j))
        yc4 = 0.5d0*(gyv2 + yv(nx,j))
        slope1 = (yc2 - yc1)/(xc2 - xc1)
        slope2 = (xc4 - xc3)/(yc4 - yc3)
        b1 = yc2 - slope1*xc2
        b2 = xc4 - slope2*yc4
        yc(nx,j-1) = (slope1*b2 + b1)/(1.0d0 - slope1*slope2)
        xc(nx,j-1) = slope2*yc(nx,j-1) + b2
c determine ghost center metric quantities (needed for BC's)
c to do this find grid line centers ("interface" centers) and finite
c difference
        xxi = 0.5d0*(gxv1+gxv2) - 0.5d0*(xv(nx,j-1)+xv(nx,j))
        yxi = 0.5d0*(gyv1+gyv2) - 0.5d0*(yv(nx,j-1)+yv(nx,j))
        xeta = 0.5d0*(gxv2+xv(nx,j)) - 0.5d0*(gxv1+xv(nx,j-1))
        yeta = 0.5d0*(gyv2+yv(nx,j)) - 0.5d0*(gyv1+yv(nx,j-1))
        jacob = 1.0d0/(xxi*yeta - xeta*yxi)
        jacc(nx,j-1) = jacob
        xixc(nx,j-1) = yeta*jacob
        xiyc(nx,j-1) = -xeta*jacob
        etaxc(nx,j-1) = -yxi*jacob
        etayc(nx,j-1) = xxi*jacob
c
        gxv1 = gxv2
        gyv1 = gyv2
110    continue
c
c Determine ghost cells for j=1 and j=ny, same way as before.
c j=0
        slope1 = (xv(1,2) - xv(1,1))/(yv(1,2) - yv(1,1))
        b1 = xv(1,1) - slope1*yv(1,1)
        gyv1 = yv(1,1) - (yv(1,2) - yv(1,1))*2/(yv(1,3) - yv(1,2))
        gxv1 = slope1*gyv1 + b1
c
    do 120 i=2,nx
        slope2 = (xv(i,2) - xv(i,1))/(yv(i,2)-yv(i,1))
        b2 = xv(i,1) - slope2*yv(i,1)
        gyv2 = yv(i,1) - (yv(i,2) - yv(i,1))*2/(yv(i,3) - yv(i,2))

```

```

      gxv2 = slope2*gyv2 + b2
c get center of cell by finding intersection of lines connecting
c interface centers, as done in center subroutine
      xc3 = 0.5d0*(gxv1 + gxv2)
      yc3 = 0.5d0*(gyv1 + gyv2)
      xc4 = 0.5d0*(xv(i,1) + xv(i-1,1))
      yc4 = 0.5d0*(yv(i,1) + yv(i-1,1))
      xc1 = 0.5d0*(gxv1 + xv(i-1,1))
      yc1 = 0.5d0*(gyv1 + yv(i-1,1))
      xc2 = 0.5d0*(gxv2 + xv(i,1))
      yc2 = 0.5d0*(gyv2 + yv(i,1))
      slope1 = (yc2 - yc1)/(xc2 -xc1)
      slope2 = (xc4 - xc3)/(yc4 - yc3)
      b1 = yc2 - slope1*xc2
      b2 = xc4 - slope2*yc4
      yc(i-1,0) = (slope1*b2 + b1)/(1.0d0 - slope1*slope2)
      xc(i-1,0) = slope2*yc(i-1,0) + b2
c
      xxi = 0.5d0*(xv(i,1)+gxv2) - 0.5d0*(xv(i-1,1)+gxv1)
      yxi = 0.5d0*(yv(i,1)+gyv2) - 0.5d0*(yv(i-1,1)+gyv1)
      xeta = 0.5d0*(xv(i,1)+xv(i-1,1)) - 0.5d0*(gxv1+gxv2)
      yeta = 0.5d0*(yv(i,1)+yv(i-1,1)) - 0.5d0*(gyv1+gyv2)
      jacob = 1.0d0/(xxi*yeta - xeta*yxi)
      jacc(i-1,0) = jacob
      xixc(i-1,0) = yeta*jacob
      xiyc(i-1,0) = -xeta*jacob
      etaxc(i-1,0) = -yxi*jacob
      etayc(i-1,0) = xxi*jacob
c
      gxv1 = gxv2
      gyv1 = gyv2
120 continue
c
c
c j=ny
      slope1 = (xv(1,ny) - xv(1,ny-1))/(yv(1,ny)-yv(1,ny-1))
      b1 = xv(1,ny) - slope1*yv(1,ny)
      gyv1 = yv(1,ny) + (yv(1,ny)-yv(1,ny-1))*2/
+      (yv(1,ny-1) - yv(1,ny-2))
      gxv1 = slope1*gyv1 + b1
c
      do 130 i=2,nx
      slope2 = (xv(i,ny) - xv(i,ny-1))/(yv(i,ny)-yv(i,ny-1))
      b2 = xv(i,ny) - slope2*yv(i,ny)
      gyv2 = yv(i,ny) + (yv(i,ny)-yv(i,ny-1))*2/
+      (yv(i,ny-1) - yv(i,ny-2))
      gxv2 = slope2*gyv2 + b2
c get center of cell by finding intersection of lines connecting
c interface centers, as done in center subroutine
      xc4 = 0.5d0*(gxv1 + gxv2)
      yc4 = 0.5d0*(gyv1 + gyv2)
      xc3 = 0.5d0*(xv(i,ny) + xv(i-1,ny))
      yc3 = 0.5d0*(yv(i,ny) + yv(i-1,ny))
      xc1 = 0.5d0*(gxv1 + xv(i-1,ny))

```

```

yc1 = 0.5d0*(gyv1 + yv(i-1,ny))
xc2 = 0.5d0*(gxv2 + xv(i,ny))
yc2 = 0.5d0*(gyv2 + yv(i,ny))
slope1 = (yc2 - yc1)/(xc2 -xc1)
slope2 = (xc4 - xc3)/(yc4 - yc3)
b1 = yc2 - slope1*xc2
b2 = xc4 - slope2*yc4
yc(i-1,ny) = (slope1*b2 + b1)/(1.0d0 - slope1*slope2)
xc(i-1,ny) = slope2*yc(i-1,ny) + b2
c
xxi = 0.5d0*(xv(i,ny)+gxv2) - 0.5d0*(xv(i-1,ny)+gxv1)
yxi = 0.5d0*(yv(i,ny)+gyv2) - 0.5d0*(yv(i-1,ny)+gyv1)
xeta = 0.5d0*(gxv1+gxv2) - 0.5d0*(xv(i,ny)+xv(i-1,ny))
yeta = 0.5d0*(gyv1+gyv2) - 0.5d0*(yv(i,ny)+yv(i-1,ny))
jacob = 1.0d0/(xxi*yeta - xeta*yxi)
jacc(i-1,ny) = jacob
xixc(i-1,ny) = yeta*jacob
xiyc(i-1,ny) = -xeta*jacob
etaxc(i-1,ny) = -yxi*jacob
etayc(i-1,ny) = xxi*jacob
c
gxv1 = gxv2
gyv1 = gyv2
130 continue
return
end

```

A.4 metrcc.f

```

subroutine metrcc(nx,ny,xv,yv,xix,xiy,etax,etay,jac)
c determine metrics at cell centers using finite differencing.
implicit double precision (a-h,o-z)
dimension xv(nx,ny),yv(nx,ny),xix(0:nx,0:ny),
+ xiy(0:nx,0:ny),etax(0:nx,0:ny),etay(0:nx,0:ny)
double precision jac(0:nx,0:ny),jacob
c
c use (2nd order) central differencing to determine metrics on interior
c Do these first because they are easy. Assume delta xi and delta eta
c are 1 for simplicty.
do 100 j=1,ny-1
xvci1 = 0.5d0*(xv(1,j+1) + xv(1,j))
yvc11 = 0.5d0*(yv(1,j+1) + yv(1,j))
do 100 i=1,nx-1
xvci2 = 0.5d0*(xv(i+1,j+1) + xv(i+1,j))
yvc12 = 0.5d0*(yv(i+1,j+1) + yv(i+1,j))
xvcj1 = 0.5d0*(xv(i,j) + xv(i+1,j))
yvcj1 = 0.5d0*(yv(i,j) + yv(i+1,j))
xvcj2 = 0.5d0*(xv(i,j+1) + xv(i+1,j+1))
yvcj2 = 0.5d0*(yv(i,j+1) + yv(i+1,j+1))
c
xxi = xvci2 - xvci1
yeta = yvcj2 - yvcj1

```



```

        xeta = xvcj2 - xvcj1
        yxi = yvci2 - yvci1
        jacob = 1.0d0/(xxi*yeta - xeta*yxi)
c
        jac(i,j) = jacob
        xix(i,j) = yeta*jacob
        xiy(i,j) = -xeta*jacob
        etax(i,j) = -yxi*jacob
        etay(i,j) = xxi*jacob
c
        xvci1 = xvci2
        yvci1 = yvci2
100    continue
    return
end

```

A.5 metrci.f

```

    subroutine metrci(nx,ny,xv,yv,x,y,xixi,xiyi,jaci)
c determine metrics at cell interfaces using finite differencing
c for interfaces along the i direction only.
    implicit double precision (a-h,o-z)
    dimension xv(nx,ny),yv(nx,ny),x(0:nx,0:ny),y(0:nx,0:ny),
+           xixi(nx,ny-1),xiyi(nx,ny-1)
    double precision jaci(nx,ny-1),jacob
c
c use (2nd order) central differencing with cell centers to determine
c metrics on interior cell interfaces. Assume delta xi and delta eta
c are one, and interface is halfway between. Thus, h for central
c difference=0.5, c and 2h = 1.0.
    do 100 j=1,ny-1
    do 100 i=1,nx
c this is central differencing, even though it looks like one sided.
        xxi = x(i,j) - x(i-1,j)
        yeta = yv(i,j+1) - yv(i,j)
        xeta = xv(i,j+1) - xv(i,j)
        yxi = y(i,j) - y(i-1,j)
        jacob = 1.0d0/(xxi*yeta - xeta*yxi)
c
        jaci(i,j) = jacob
        xixi(i,j) = yeta*jacob
        xiyi(i,j) = -xeta*jacob
100    continue
c
    return
end

```

A.6 metrcj.f

```

    subroutine metrcj(nx,ny,xv,yv,x,y,etaxj,etayj,jacj)
c determine metrics at cell interfaces using finite differencing

```

```

c for interfaces along the j direction only.
  implicit double precision (a-h,o-z)
  dimension xv(nx,ny),yv(nx,ny),x(0:nx,0:ny),y(0:nx,0:ny),
+      etaxj(nx-1,ny),etayj(nx-1,ny)
  double precision jacj(nx-1,ny),jacob
c
c use (2nd order) central differencing to determine metrics on
c interior cell interfaces. Assume delta xi and delta eta are one,
c and interface is halfway between. Thus, h for central difference=0.5,
c and 2h = 1.0.
  do 100 j=1,ny
    do 100 i=1,nx-1
c this is central differencing, even though it looks like one sided.
      xxi = xv(i+1,j) - xv(i,j)
      yeta = y(i,j) - y(i,j-1)
      xeta = x(i,j) - x(i,j-1)
      yxi = yv(i+1,j) - yv(i,j)
      jacob = 1.0d0/(xxi*yeta - xeta*yxi)
c
      jacj(i,j) = jacob
      etaxj(i,j) = -yxi*jacob
      etayj(i,j) = xxi*jacob
100  continue
c
  return
end

```

A.7 timest.f

```

  subroutine timest(nx,ny,rhoc,pc,uc,vc,xixc,xiyc,etaxc,etayc,dt)
c Determines timestep by considering magnitudes of eigenvalues at cell
c centers.
  implicit double precision (a-h,o-z)
  dimension rhoc(0:nx,0:ny),pc(0:nx,0:ny),uc(0:nx,0:ny),
+      vc(0:nx,0:ny),xixc(0:nx,0:ny),xiyc(0:nx,0:ny),
+      etaxc(0:nx,0:ny),etayc(0:nx,0:ny)
  double precision m02
  common gam1,gam12,speed,uin,vin,pin,rhoin,bigen,bighn
  common /chemistry/ y2in,qheat,theta,m02
c
c Determine max eigenvalues at cells, which corresponds to the
c fastest travelling wave. Determine how long it would take this wave
c to cover half of a cell.
  gamma = gam1 + 1.0d0
  speed = 0.0d0
  do 100 j=1,ny-1
    do 100 i=0,nx-1
      xix = xixc(i,j)
      xiy = xiyc(i,j)
      etax = etaxc(i,j)
      etay = etayc(i,j)
      xi2 = sqrt(xix*xix + xiy*xiy)

```

```

        eta2 = sqrt(etax*etax + etay*etay)
        c = sqrt(gamma*pc(i,j)/rhoc(i,j))
        ucc = xix*uc(i,j) + xiy*vc(i,j)
        vcc = etax*uc(i,j) + etay*vc(i,j)
c absolute eigenvalues
        ei1 = abs(ucc - c*xix)
        ei2 = abs(ucc + c*xix)
        ej1 = abs(vcc - c*eta2)
        ej2 = abs(vcc + c*eta2)
c max eigenvalues in each direction
        e1 = max(ei1,ei2)
        e2 = max(ej1,ej2)
c take magnitude of two max eigenvalues as greatest wave speed
        e = sqrt(e1*e1 + e2*e2)
c finally, get maximum over entire grid
        speed = max(speed,e)
100    continue
c thus, the time to cross half a cell (1 by 1 in xi-eta space)
c plus a safety factor.
        dt = (0.5d0/speed)*0.8d0
        return
        end

```

A.8 flxxi.f

```

        subroutine flxxi(nx,ny,rhoc,pc,uc,vc,bighc,y2c,q,xixc,xiyc,
+           jacc,xixi,xiyi,eigval,eigvec,dw,fflux)
c Determines interface fluxes in the xi direction using the Roe
c scheme.
        implicit double precision (a-h,o-z)
        dimension rhoc(0:nx,0:ny),pc(0:nx,0:ny),
+           uc(0:nx,0:ny),vc(0:nx,0:ny),y2c(0:nx,0:ny),
+           bighc(0:nx,0:ny),q(0:nx,0:ny,5),xixc(0:nx,0:ny),
+           xiyc(0:nx,0:ny)
        dimension xixi(nx,ny-1),xiyi(nx,ny-1),fflux(nx,ny-1,5)
        dimension eigval(nx,5),eigvec(nx,5,5),dw(nx,5)
        double precision metsq,inveig(5,5),jacc(0:nx,0:ny),m02
        common gam1,gam12,speed,uin,vin,pin,rhoin,bigen,bighn
        common /chemistry/ y2in,qheat,theta,m02
c
c loop through xi interfaces of entire grid
        do 100 j=1,ny-1
c Determine Roe averaged variables
c i counting interfaces
c count from 1 to nx so that roe quantities are calculated at inflow
c for use in the higher order flux terms below.
c Assume the ghost cells have been properly set for this to work.
        do 110 i=1,nx-1
            rr = sqrt(rhoc(i,j)/jacc(i,j))/sqrt(rhoc(i-1,j)/jacc(i-1,j))
            ua = (rr*uc(i,j) + uc(i-1,j))/(rr+1.0d0)
            va = (rr*vc(i,j) + vc(i-1,j))/(rr+1.0d0)
            bigha = (rr*bighc(i,j) + bighc(i-1,j))/(rr+1.0d0)

```

```

y2a = (rr*y2c(i,j) + y2c(i-1,j))/(rr+1.0d0)
ca = sqrt(gam1*(bigha + y2a*qheat - 0.5d0*(ua*ua + va*va)))
c other useful quantities
xix = xixi(i,j)
xiy = xiyi(i,j)
uacont = xix*ua + xiy*va
sqrmet = sqrt(xix*xix + xiy*xiy)
metsq = xix*xix + xiy*xiy
ca2 = ca*ca
usq = ua*ua + va*va
con1 = gam1/(xix*ca2)
con2 = gam1/(2.0d0*ca2)
con3 = 2.0d0*ca*sqrmet
c eigenvalues of generalized flux jacobian
eigval(i,1) = uacont
eigval(i,2) = uacont
eigval(i,3) = uacont - ca*sqrmet
eigval(i,4) = uacont + ca*sqrmet
eigval(i,5) = uacont
c eigenvectors of generalized flux jacobian
c number 1
eigvec(i,1,1) = xix
eigvec(i,2,1) = uacont
eigvec(i,3,1) = 0.0d0
eigvec(i,4,1) = (ua*ua - va*va)*xix/2.0d0 + xiy*ua*va
eigvec(i,5,1) = 0.0d0
c number 2
eigvec(i,1,2) = 0.0d0
eigvec(i,2,2) = xiy
eigvec(i,3,2) = -xix
eigvec(i,4,2) = xiy*ua - xix*va
eigvec(i,5,2) = 0.0d0
c number 3
eigvec(i,1,3) = 1.0d0
eigvec(i,2,3) = ua - xix*ca/sqrmet
eigvec(i,3,3) = va - xiy*ca/sqrmet
eigvec(i,4,3) = bigha - ca*uacont/sqrmet
eigvec(i,5,3) = y2a
c number 4
eigvec(i,1,4) = 1.0d0
eigvec(i,2,4) = ua + xix*ca/sqrmet
eigvec(i,3,4) = va + xiy*ca/sqrmet
eigvec(i,4,4) = bigha + ca*uacont/sqrmet
eigvec(i,5,4) = y2a
c number 5
eigvec(i,1,5) = 0.0d0
eigvec(i,2,5) = 0.0d0
eigvec(i,3,5) = 0.0d0
eigvec(i,4,5) = -qheat
eigvec(i,5,5) = 1.0d0
c inverse eigenvector array for obtaining Roe coefficients
inveig(1,1) = (1.0d0/xix) - 0.5d0*con1*usq
inveig(1,2) = con1*ua
inveig(1,3) = con1*va

```

```

    inveig(1,4) = -con1
    inveig(1,5) = -con1*qheat
c
    inveig(2,1) = 0.5d0*con1*usq*va - xiy*uacont/(xix*metsq)
    inveig(2,2) = xiy/metsq - con1*ua*va
    inveig(2,3) = -(con1*va*va + xix/metsq)
    inveig(2,4) = con1*va
    inveig(2,5) = con1*va*qheat
c
    inveig(3,1) = 0.5d0*con2*usq + uacont/con3
    inveig(3,2) = -(con2*ua + xix/con3)
    inveig(3,3) = -(con2*va + xiy/con3)
    inveig(3,4) = con2
    inveig(3,5) = con2*qheat
c
    inveig(4,1) = 0.5d0*con2*usq - uacont/con3
    inveig(4,2) = xix/con3 - con2*ua
    inveig(4,3) = xiy/con3 - con2*va
    inveig(4,4) = con2
    inveig(4,5) = con2*qheat
c
    inveig(5,1) = -con2*usq*y2a
    inveig(5,2) = con2*2.0d0*ua*y2a
    inveig(5,3) = con2*2.0d0*va*y2a
    inveig(5,4) = -con2*2.0d0*y2a
    inveig(5,5) = 1.0d0 - con2*2.0d0*qheat*y2a
c determine Roe coefficients using inverse eigenvector array and
c dependent variables
    qrql1 = q(i,j,1) - q(i-1,j,1)
    qrql2 = q(i,j,2) - q(i-1,j,2)
    qrql3 = q(i,j,3) - q(i-1,j,3)
    qrql4 = q(i,j,4) - q(i-1,j,4)
    qrql5 = q(i,j,5) - q(i-1,j,5)
    do 120 k=1,5
        dw(i,k) = inveig(k,1)*qrql1 + inveig(k,2)*qrql2 +
+               inveig(k,3)*qrql3 + inveig(k,4)*qrql4 +
+               inveig(k,5)*qrql5
120    continue
110    continue
c
c Determine fluxes at xi interfaces using Roe scheme
c i counting interfaces
    do 130 i=2,nx-1
        sum1 = 0.0d0
        sum2 = 0.0d0
        sum3 = 0.0d0
        sum4 = 0.0d0
        sum5 = 0.0d0
        suml1 = 0.0d0
        suml2 = 0.0d0
        suml3 = 0.0d0
        suml4 = 0.0d0
        suml5 = 0.0d0
    do 140 k=1,5

```

```

    ajm1 = eigval(i-1,k)
    aj = eigval(i,k)
    ajp1 = eigval(i+1,k)
    dwjm1 = dw(i-1,k)
    dwj = dw(i,k)
    dwjp1 = dw(i+1,k)
c first-order flux terms
    sum1 = sum1 + abs(aj)*dwj*eigvec(i,1,k)
    sum2 = sum2 + abs(aj)*dwj*eigvec(i,2,k)
    sum3 = sum3 + abs(aj)*dwj*eigvec(i,3,k)
    sum4 = sum4 + abs(aj)*dwj*eigvec(i,4,k)
    sum5 = sum5 + abs(aj)*dwj*eigvec(i,5,k)
c higher-order flux terms
    if(aj.gt.0.0d0.and.ajm1.gt.0.0d0.and.dwj.ne.0.0d0) then
        biglp = (ajm1*dwjm1*aj*dwj + abs(ajm1*dwjm1*aj*dwj))/
+             (ajm1*dwjm1 + aj*dwj)
        suml1 = suml1 + biglp*eigvec(i,1,k)
        suml2 = suml2 + biglp*eigvec(i,2,k)
        suml3 = suml3 + biglp*eigvec(i,3,k)
        suml4 = suml4 + biglp*eigvec(i,4,k)
        suml5 = suml5 + biglp*eigvec(i,5,k)
    elseif(aj.lt.0.0d0.and.ajp1.lt.0.0d0.and.dwj.ne.0.0d0) then
        biglm = (aj*dwj*ajp1*dwjp1 + abs(aj*dwj*ajp1*dwjp1))/
+             (aj*dwj + ajp1*dwjp1)
        suml1 = suml1 - biglm*eigvec(i,1,k)
        suml2 = suml2 - biglm*eigvec(i,2,k)
        suml3 = suml3 - biglm*eigvec(i,3,k)
        suml4 = suml4 - biglm*eigvec(i,4,k)
        suml5 = suml5 - biglm*eigvec(i,5,k)
    endif
140    continue
c calculate interface fluxes according to complicated Roe flux
c equation in case need sonic correction
    const1 = rhoc(i-1,j)*(xixc(i-1,j)*uc(i-1,j) +
+             xiyc(i-1,j)*vc(i-1,j))/jacc(i-1,j)
    const2 = rhoc(i,j)*(xixc(i,j)*uc(i,j) +
+             xiyc(i,j)*vc(i,j))/jacc(i,j)
c
    fflux(i,j,1) = 0.5d0*(const1 + const2 - sum1 + suml1)
    fflux(i,j,2) = 0.5d0*(const1*uc(i-1,j) + xixc(i-1,j)*pc(i-1,j)
+             /jacc(i-1,j) + const2*uc(i,j) + xixc(i,j)*
+             pc(i,j)/jacc(i,j) - sum2 + suml2)
    fflux(i,j,3) = 0.5d0*(const1*vc(i-1,j) + xiyc(i-1,j)*pc(i-1,j)
+             /jacc(i-1,j) + const2*vc(i,j) + xiyc(i,j)*
+             pc(i,j)/jacc(i,j) - sum3 + suml3)
    fflux(i,j,4) = 0.5d0*(const1*bighc(i-1,j) + const2*bighc(i,j)
+             - sum4 + suml4)
    fflux(i,j,5) = 0.5d0*(const1*y2c(i-1,j) + const2*y2c(i,j) -
+             sum5 + suml5)
130    continue
100    continue
    return
end

```

A.9 boundx.f

```

      subroutine boundx(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,
+
+          fflux,xixc,xiyc,jacc,dt,intmed)
c Calculates values for boundary cells at exit.
c For use after xi direction integration step
  implicit double precision (a-h,o-z)
  dimension rhoc(0:nx,0:ny),pc(0:nx,0:ny),uc(0:nx,0:ny),
+      vc(0:nx,0:ny),bighc(0:nx,0:ny),qbar(0:nx,0:ny,5),
+      q(0:nx,0:ny,5),fflux(nx,ny-1,5),y2c(0:nx,0:ny)
  dimension xixc(0:nx,0:ny),xiyc(0:nx,0:ny)
  double precision jacc(nx,ny),jac,metsq,m02
  logical intmed
  common gam1,gam12,speed,uin,vin,pin,rhoin,bigen,bighn
  common /chemistry/ y2in,qheat,theta,m02
c
  gamma = gam1+1.0d0
c Do outlet conditions first
c setup conditions to handle both supersonic and subsonic outflow,
c following Poinot and Lele's suggestions for nonreflecting subsonic
  nn = nx-1
  do 100 j=1,ny-1
c Calculate xi derivatives using one sided differencing (d xi=1)
  drodx = 0.5d0*(3.0d0*rhoc(nn,j) - 4.0d0*rhoc(nx-2,j) +
+      rhoc(nx-3,j))
  dudx = 0.5d0*(3.0d0*uc(nn,j)- 4.0d0*uc(nx-2,j) + uc(nx-3,j))
  dvdx = 0.5d0*(3.0d0*vc(nn,j)- 4.0d0*vc(nx-2,j) + vc(nx-3,j))
  dpdx = 0.5d0*(3.0d0*pc(nn,j)- 4.0d0*pc(nx-2,j) + pc(nx-3,j))
  dy2dx = 0.5d0*(3.0d0*y2c(nn,j) - 4.0d0*y2c(nx-2,j) +
+      y2c(nx-3,j))
c build script H values (assume Uc is positive)
  xix = xixc(nn,j)
  xiy = xiyc(nn,j)
  jac = jacc(nn,j)
  metsq = xix*xix + xiy*xiy
  sqrmet = sqrt(metsq)
  ucont = xix*uc(nn,j) + xiy*vc(nn,j)
  c = sqrt(gamma*pc(nn,j)/rhoc(nn,j))
c
  scrh1 = ucont*(-drodx + xiy*dudx/metsq - xix*dvdx/metsq +
+      dpdx/(c*c) - dy2dx)
  scrh2 = ucont*(drodx - dpdx/(c*c))
  scrh3 = ucont*dy2dx
  scrh4 = (ucont + c*sqrmet)*((xix*dudx + xiy*dvdx)/(2.0d0*
+      c*metsq) + dpdx/(2.0d0*gamma*pc(nn,j)*sqrmet))
  if((ucont-c*sqrmet).gt.0.0d0) then
    scrh5 = (ucont - c*sqrmet)*(-(xix*dudx + xiy*dvdx)/(2.0d0*
+      c*metsq) + dpdx/(2.0d0*gamma*pc(nn,j)*sqrmet))
  else
c K factor from Poinot and Lele
    scrh5 = bigk*(pc(nn,j) - pin)
  endif
c use script H values in integration to get new q's at exit
  con = rhoc(nn,j)*sqrmet

```

```

        xixr = xix*rhoc(nn,j)
        xiyr = xiy*rhoc(nn,j)
        if(intmed) then
qbar(nn,j,1) = q(nn,j,1)-dt*(scrh2 + con*(scrh4+scrh5))/jac
qbar(nn,j,2) = q(nn,j,2)-dt*(xiyr*(scrh1+scrh2+scrh3) +
+         uc(nn,j)*scrh2 + xixr*c*(scrh4-scrh5) +
+         con*uc(nn,j)*(scrh4+scrh5))/jac
qbar(nn,j,3) = q(nn,j,3)-dt*(vc(nn,j)*scrh2 - xixr*(scrh1+
+         scrh2+scrh3) + xiyr*c*(scrh4-scrh5) + con*
+         vc(nn,j)*(scrh4+scrh5))/jac
qbar(nn,j,4) = q(nn,j,4)-dt*((xiyr*uc(nn,j)-xixr*vc(nn,j))*
+         (scrh1+scrh2+scrh3) + ((uc(nn,j)*uc(nn,j)+
+         vc(nn,j)*vc(nn,j))/2.0d0 - qheat*y2c(nn,j))*scrh2-
+         rhoc(nn,j)*qheat*scrh3 + rhoc(nn,j)*c*ucont*
+         (scrh4-scrh5) + con*bighc(nn,j)*(scrh4+scrh5))/jac
qbar(nn,j,5) = q(nn,j,5)-dt*(y2c(nn,j)*scrh2 + rhoc(nn,j)*
+         scrh3 + con*y2c(nn,j)*(scrh4+scrh5))/jac
        else
q(nn,j,1) = q(nn,j,1) - dt*(scrh2 + con*(scrh4+scrh5))/jac
q(nn,j,2) = q(nn,j,2) - dt*(xiyr*(scrh1+scrh2+scrh3) +
+         uc(nn,j)*scrh2 + xixr*c*(scrh4-scrh5) +
+         con*uc(nn,j)*(scrh4+scrh5))/jac
q(nn,j,3) = q(nn,j,3) - dt*(vc(nn,j)*scrh2 - xixr*(scrh1+
+         scrh2+scrh3) + xiyr*c*(scrh4-scrh5) + con*
+         vc(nn,j)*(scrh4+scrh5))/jac
q(nn,j,4) = q(nn,j,4) - dt*((xiyr*uc(nn,j)-xixr*vc(nn,j))*
+         (scrh1+scrh2+scrh3) + ((uc(nn,j)*uc(nn,j)+
+         vc(nn,j)*vc(nn,j))/2.0d0 - qheat*y2c(nn,j))*scrh2-
+         rhoc(nn,j)*qheat*scrh3 + rhoc(nn,j)*c*ucont*
+         (scrh4-scrh5) + con*bighc(nn,j)*(scrh4+scrh5))/jac
q(nn,j,5) = q(nn,j,5) - dt*(y2c(nn,j)*scrh2 + rhoc(nn,j)*
+         scrh3 + con*y2c(nn,j)*(scrh4+scrh5))/jac
        endif
100 continue
c
        return
        end

```

A.10 decode.f

```

        subroutine decode(nx,ny,rhoc,pc,uc,vc,bigec,bighc,y2c,q,jacc)
c Decodes the dependent variable vector into primitive variables.
        implicit double precision (a-h,o-z)
        dimension rhoc(0:nx,0:ny),pc(0:nx,0:ny),uc(0:nx,0:ny),
+         vc(0:nx,0:ny),bigec(0:nx,0:ny),bighc(0:nx,0:ny),
+         q(0:nx,0:ny,5),y2c(0:nx,0:ny)
        double precision jacc(0:nx,0:ny),jac,m02
        common gam1,gam12,speed,uin,vin,pin,rhoin,bigen,bighn
        common /chemistry/ y2in,qheat,theta,m02
c
        do 100 j=1,ny-1
            do 100 i=1,nx-1

```



```

        jac = jacc(i,j)
        rho = q(i,j,1)*jac
c
        rhoc(i,j) = rho
        uc(i,j) = q(i,j,2)*jac/rho
        vc(i,j) = q(i,j,3)*jac/rho
        bigec(i,j) = q(i,j,4)*jac/rho
        y2c(i,j) = q(i,j,5)*jac/rho
c
        pc(i,j) = rho*gam1*(bigec(i,j) + y2c(i,j)*qheat - 0.5d0*
+          (uc(i,j)*uc(i,j) + vc(i,j)*vc(i,j)))
        bighc(i,j) = bigec(i,j) + pc(i,j)/rho
100  continue
        return
        end

```

A.11 flxeta.f

```

        subroutine flxeta(nx,ny,rhoc,pc,uc,vc,bighc,y2c,q,etaxc,etayc,
+          jacc,etaxi,etayi,jaci,eigval,eigvec,dw,gflux,topwal)
c Determines interface fluxes in the eta direction using the Roe
c scheme.
        implicit double precision (a-h,o-z)
        dimension rhoc(0:nx,0:ny),pc(0:nx,0:ny),
+          uc(0:nx,0:ny),vc(0:nx,0:ny),y2c(0:nx,0:ny),
+          bighc(0:nx,0:ny),q(0:nx,0:ny,5),etaxc(0:nx,0:ny),
+          etayc(0:nx,0:ny)
        dimension etaxi(nx-1,ny),etayi(nx-1,ny),gflux(nx-1,ny,5)
        dimension eigval(ny,5),eigvec(ny,5,5),dw(ny,5)
        double precision inveig(5,5),jacc(0:nx,0:ny),jaci(nx-1,ny),
+          metsq,m02,jac
        logical topwal
        common gam1,gam12,speed,uin,vin,pin,rhoin,bigen,bighn
        common /chemistry/ y2in,qheat,theta,m02
c
c loop through eta interfaces of entire grid
        do 100 i=1,nx-1
c Determine Roe averaged variables
c j counting interfaces
        do 110 j=2,ny-1
            rr = sqrt(rhoc(i,j)/jacc(i,j))/sqrt(rhoc(i,j-1)/jacc(i,j-1))
            ua = (rr*uc(i,j) + uc(i,j-1))/(rr+1.0d0)
            va = (rr*vc(i,j) + vc(i,j-1))/(rr+1.0d0)
            bigha = (rr*bighc(i,j) + bighc(i,j-1))/(rr+1.0d0)
            y2a = (rr*y2c(i,j) + y2c(i,j-1))/(rr+1.0d0)
            ca = sqrt(gam1*(bigha + y2a*qheat - 0.5d0*(ua*ua + va*va)))
c other useful quantities
            etax = etaxi(i,j)
            etay = etayi(i,j)
            vacont = etax*ua + etay*va
            sqrmet = sqrt(etax*etax + etay*etay)
            metsq = etax*etax + etay*etay

```

```

ca2 = ca*ca
usq = ua*ua + va*va
con1 = gam1/(etay*ca2)
con2 = gam1/(2.0d0*ca2)
con3 = 2.0d0*ca*sqrmet
c eigenvalues of generalized flux jacobian
eigval(j,1) = vacont
eigval(j,2) = vacont
eigval(j,3) = vacont - ca*sqrmet
eigval(j,4) = vacont + ca*sqrmet
eigval(j,5) = vacont
c eigenvectors of generalized flux jacobian
c number 1
eigvec(j,1,1) = etay
eigvec(j,2,1) = 0.0d0
eigvec(j,3,1) = vacont
eigvec(j,4,1) = (va*va-ua*ua)*etay/2.0d0 + etax*ua*va
eigvec(j,5,1) = 0.0d0
c number 2
eigvec(j,1,2) = 0.0d0
eigvec(j,2,2) = etay
eigvec(j,3,2) = -etax
eigvec(j,4,2) = etay*ua - etax*va
eigvec(j,5,2) = 0.0d0
c number 3
eigvec(j,1,3) = 1.0d0
eigvec(j,2,3) = ua - etax*ca/sqrmet
eigvec(j,3,3) = va - etay*ca/sqrmet
eigvec(j,4,3) = bigha - ca*vacont/sqrmet
eigvec(j,5,3) = y2a
c number 4
eigvec(j,1,4) = 1.0d0
eigvec(j,2,4) = ua + etax*ca/sqrmet
eigvec(j,3,4) = va + etay*ca/sqrmet
eigvec(j,4,4) = bigha + ca*vacont/sqrmet
eigvec(j,5,4) = y2a
c number 5
eigvec(j,1,5) = 0.0d0
eigvec(j,2,5) = 0.0d0
eigvec(j,3,5) = 0.0d0
eigvec(j,4,5) = -qheat
eigvec(j,5,5) = 1.0d0
c inverse eigenvector array for obtaining Roe coefficients
inveig(1,1) = (1.0d0/etay) - 0.5d0*con1*usq
inveig(1,2) = con1*ua
inveig(1,3) = con1*va
inveig(1,4) = -con1
inveig(1,5) = -con1*qheat
c
inveig(2,1) = etax*vacont/(etay*metsq) - 0.5d0*con1*usq*ua
inveig(2,2) = etay/metsq + con1*ua*ua
inveig(2,3) = con1*ua*va - etax/metsq
inveig(2,4) = -con1*ua
inveig(2,5) = -con1*ua*qheat

```

```

c
    inveig(3,1) = 0.5d0*con2*usq + vacont/con3
    inveig(3,2) = -(con2*ua + etax/con3)
    inveig(3,3) = -(con2*va + etay/con3)
    inveig(3,4) = con2
    inveig(3,5) = con2*qheat

c
    inveig(4,1) = 0.5d0*con2*usq - vacont/con3
    inveig(4,2) = etax/con3 - con2*ua
    inveig(4,3) = etay/con3 - con2*va
    inveig(4,4) = con2
    inveig(4,5) = con2*qheat

c
    inveig(5,1) = -con2*usq*y2a
    inveig(5,2) = con2*2.0d0*ua*y2a
    inveig(5,3) = con2*2.0d0*va*y2a
    inveig(5,4) = -con2*2.0d0*y2a
    inveig(5,5) = 1.0d0 - con2*2.0d0*qheat*y2a
c determine Roe coefficients using inverse eigenvector array and
c dependent variables
    qrql1 = q(i,j,1) - q(i,j-1,1)
    qrql2 = q(i,j,2) - q(i,j-1,2)
    qrql3 = q(i,j,3) - q(i,j-1,3)
    qrql4 = q(i,j,4) - q(i,j-1,4)
    qrql5 = q(i,j,5) - q(i,j-1,5)
    do 120 k=1,5
        dw(j,k) = inveig(k,1)*qrql1 + inveig(k,2)*qrql2 +
+               inveig(k,3)*qrql3 + inveig(k,4)*qrql4 +
+               inveig(k,5)*qrql5
120    continue
110    continue
c
c Determine values of variables at walls for wall flux and eigvals
c and dw at walls by using fact that contravariant velocity is zero
c at the wall. Then use riemann invariants to estimate p and c at wall
c bottom wall
    etay = etayc(i,1)
    etax = etaxc(i,1)
    sqrmet = sqrt(etax*etax + etay*etay)
    c = sqrt((gam1+1.0d0)*pc(i,1)/rhoc(i,1))
    vcont = etax*uc(i,1) + etay*vc(i,1)
    p = pc(i,1) - rhoc(i,1)*c*vcont/sqrmet
c integrated riemann invariant to estimate c
    con = vcont - sqrmet*(2.0d0*c)/gam1
    sqrmtw = sqrt(etaxi(i,1)**2 + etayi(i,1)**2)
    cwalsq = -gam1*con/2.0d0
c set flux at wall and rest of variables
    gflux(i,1,1) = 0.0d0
    gflux(i,1,2) = etaxi(i,1)*p/jaci(i,1)
    gflux(i,1,3) = etayi(i,1)*p/jaci(i,1)
    gflux(i,1,4) = 0.0d0
    gflux(i,1,5) = 0.0d0
c estimate dw using Roe method and knowing Vc=0 and p and c
    eigval(1,4) = cwalsq

```

```

        dw(1,4) = (rhoc(i,1)*vcont/jacc(i,1))/eigval(1,4)
c
c top wall
c if topwal=true then setup fluxes for wall condition, otherwise
c set to zero as boundn subroutine will take care of this
        gflux(i,ny,1) = 0.0d0
        gflux(i,ny,4) = 0.0d0
        gflux(i,ny,5) = 0.0d0
        if(topwal) then
            etay = etayc(i,ny-1)
            etax = etaxc(i,ny-1)
            sqrmet = sqrt(etax*etax + etay*etay)
            c = sqrt((gam1+1.0d0)*pc(i,ny-1)/rhoc(i,ny-1))
            vcont = etax*uc(i,ny-1) + etay*vc(i,ny-1)
            p = pc(i,ny-1) + rhoc(i,ny-1)*c*vcont/sqrmet
c integrated riemann invariant to estimate c
            con = vcont + sqrmet*(2.0d0*c)/gam1
            sqrrtw = sqrt(etaxi(i,1)**2 + etayi(i,1)**2)
            cwalsq = gam1*con/2.0d0
c set flux at wall and rest of variables
            gflux(i,ny,2) = etaxi(i,ny)*p/jaci(i,ny)
            gflux(i,ny,3) = etayi(i,ny)*p/jaci(i,ny)
c estimate dw using Roe method and knowing Vc=0 and p and c
            eigval(ny,3) = -cwalsq
            dw(ny,3) = (rhoc(i,ny-1)*vcont/jacc(i,ny-1))/(-eigval(ny,3))
        else
            gflux(i,ny,2) = 0.0d0
            gflux(i,ny,3) = 0.0d0
            eigval(ny,3) = 0.0d0
            dw(ny,3) = 0.0d0
        endif
c
c Determine fluxes at xi interfaces using Roe scheme
c Skip boundary fluxes, because higher order terms will cause problems
c i counting interfaces
        do 130 j=2,ny-1
            sum1 = 0.0d0
            sum2 = 0.0d0
            sum3 = 0.0d0
            sum4 = 0.0d0
            sum5 = 0.0d0
            suml1 = 0.0d0
            suml2 = 0.0d0
            suml3 = 0.0d0
            suml4 = 0.0d0
            suml5 = 0.0d0
            do 140 k=1,5
                ajm1 = eigval(j-1,k)
                aj = eigval(j,k)
                ajp1 = eigval(j+1,k)
                dwjm1 = dw(j-1,k)
                dwj = dw(j,k)
                dwjp1 = dw(j+1,k)
c first-order flux terms

```

```

        sum1 = sum1 + abs(aj)*dwj*eigvec(j,1,k)
        sum2 = sum2 + abs(aj)*dwj*eigvec(j,2,k)
        sum3 = sum3 + abs(aj)*dwj*eigvec(j,3,k)
        sum4 = sum4 + abs(aj)*dwj*eigvec(j,4,k)
        sum5 = sum5 + abs(aj)*dwj*eigvec(j,5,k)
c higher-order flux terms
        if(aj.gt.0.0d0.and.ajm1.gt.0.0d0.and.dwj.ne.0.0d0) then
            biglp = (ajm1*dwjml*aj*dwj + abs(ajm1*dwjml*aj*dwj))/
+                (ajm1*dwjml + aj*dwj)
            suml1 = suml1 + biglp*eigvec(j,1,k)
            suml2 = suml2 + biglp*eigvec(j,2,k)
            suml3 = suml3 + biglp*eigvec(j,3,k)
            suml4 = suml4 + biglp*eigvec(j,4,k)
            suml5 = suml5 + biglp*eigvec(j,5,k)
        elseif(aj.lt.0.0d0.and.ajp1.lt.0.0d0.and.dwj.ne.0.0d0) then
            biglm = (aj*dwj*ajp1*dwj*ajp1 + abs(aj*dwj*ajp1*dwj*ajp1))/
+                (aj*dwj + ajp1*dwj*ajp1)
            suml1 = suml1 - biglm*eigvec(j,1,k)
            suml2 = suml2 - biglm*eigvec(j,2,k)
            suml3 = suml3 - biglm*eigvec(j,3,k)
            suml4 = suml4 - biglm*eigvec(j,4,k)
            suml5 = suml5 - biglm*eigvec(j,5,k)
        endif
140     continue
c calculate interface fluxes according to complicated Roe flux
c equation in case need sonic correction
        const1 = rhoc(i,j-1)*(etaxc(i,j-1)*uc(i,j-1) +
+                etayc(i,j-1)*vc(i,j-1))/jacc(i,j-1)
        const2 = rhoc(i,j)*(etaxc(i,j)*uc(i,j) +
+                etayc(i,j)*vc(i,j))/jacc(i,j)
c
        gflux(i,j,1) = 0.5d0*(const1 + const2 - sum1 + suml1)
        gflux(i,j,2) = 0.5d0*(const1*uc(i,j-1) + etaxc(i,j-1)*pc(i,j-1)
+                /jacc(i,j-1) + const2*uc(i,j) + etaxc(i,j)*
+                pc(i,j)/jacc(i,j) - sum2 + suml2)
        gflux(i,j,3) = 0.5d0*(const1*vc(i,j-1) + etayc(i,j-1)*pc(i,j-1)
+                /jacc(i,j-1) + const2*vc(i,j) + etayc(i,j)*
+                pc(i,j)/jacc(i,j) - sum3 + suml3)
        gflux(i,j,4) = 0.5d0*(const1*bighc(i,j-1) + const2*bighc(i,j)
+                - sum4 + suml4)
        gflux(i,j,5) = 0.5d0*(const1*y2c(i,j-1) + const2*y2c(i,j) -
+                sum5 + suml5)
130     continue
100     continue
        return
        end

```

A.12 boundn.f

```

subroutine boundn(nx,ny,bigk,rhoc,uc,vc,pc,y2c,bighc,q,qbar,
+                gflux,etaxc,etayc,jacc,etaxi,etayi,jaci,
+                xixc,xiyc,dt,intmed)

```

```

c Calculates values for boundary cells at top.
c For use after eta direction integration step
  implicit double precision (a-h,o-z)
  dimension rhoc(0:nx,0:ny),pc(0:nx,0:ny),uc(0:nx,0:ny),
+   vc(0:nx,0:ny),bighc(0:nx,0:ny),qbar(0:nx,0:ny,5),
+   q(0:nx,0:ny,5),gflux(nx-1,ny,5),y2c(0:nx,0:ny)
  dimension etaxc(0:nx,0:ny),etayc(0:nx,0:ny),etaxi(nx-1,ny),
+   etayi(nx-1,ny),xixc(0:nx,0:ny),xiyc(0:nx,0:ny)
  double precision jacc(nx,ny),jaci(nx-1,ny),jac,metsq,m02
  logical intmed
  common gam1,gam12,speed,uin,vin,pin,rhoin,bigen,bighn
  common /chemistry/ y2in,qheat,theta,m02
c
  gamma = gam1+1.0d0
c Do outlet conditions first
c setup conditions to handle both supersonic and subsonic outflow,
c following Poinot and Lele's suggestions for nonreflecting subsonic
  nn = ny-1
  do 100 i=1,nx-1
c Calculate eta derivatives using one sided differencing (d eta=1)
  drodn = 0.5d0*(3.0d0*rhoc(i,nn) - 4.0d0*rhoc(i,ny-2) +
+   rhoc(i,ny-3))
  dudn = 0.5d0*(3.0d0*uc(i,nn)- 4.0d0*uc(i,ny-2) + uc(i,ny-3))
  dvdn = 0.5d0*(3.0d0*vc(i,nn)- 4.0d0*vc(i,ny-2) + vc(i,ny-3))
  dpdn = 0.5d0*(3.0d0*pc(i,nn)- 4.0d0*pc(i,ny-2) + pc(i,ny-3))
  dy2dn = 0.5d0*(3.0d0*y2c(i,nn) - 4.0d0*y2c(i,ny-2) +
+   y2c(i,ny-3))
c
  etax = etaxc(i,nn)
  etay = etayc(i,nn)
  jac = jacc(i,nn)
  metsq = etax*etax + etay*etay
  sqrmet = sqrt(metsq)
  vcont = etax*uc(i,nn) + etay*vc(i,nn)
  c = sqrt(gamma*pc(i,nn)/rhoc(i,nn))
c
  if(vcont.le.0.0d0) then
c If Vc <= 0 then set top interface flux equal to freestream
c conditions and integrate q forward in time
  con1 = rhoin*(uin*etaxi(i,ny) + vin*etayi(i,ny))/jaci(i,ny)
  gflux(i,ny,1) = con1
  gflux(i,ny,2) = con1*uin + etaxi(i,ny)*pin/jaci(i,ny)
  gflux(i,ny,3) = con1*vin + etayi(i,ny)*pin/jaci(i,ny)
  gflux(i,ny,4) = con1*bighn
  gflux(i,ny,5) = con1*y2in
  if(intmed) then
  do 150 k=1,5
    qbar(i,nn,k) = q(i,nn,k)-dt*(gflux(i,ny,k)-gflux(i,ny-1,k))
150  continue
  else
  do 145 k=1,5
    q(i,nn,k) = q(i,nn,k) - dt*(gflux(i,ny,k)-gflux(i,ny-1,k))
145  continue

```

```

        endif
    else
c else build script H values (assume Vc is positive)
c and use one sided integration for outflow boundary
    scrh1 = vcont*(-drodn + etay*dudn/metsq - etax*dvdn/metsq +
+           dpdn/(c*c) - dy2dn)
    scrh2 = vcont*(drodn - dpdn/(c*c))
    scrh3 = vcont*dy2dn
    scrh4 = (vcont + c*sqrmet)*((etax*dudn + etay*dvdn)/(2.0d0*
+           c*metsq) + dpdn/(2.0d0*gamma*pc(i,nn)*sqrmet))
    if((vcont-c*sqrmet).gt.0.0d0) then
        scrh5 = (vcont - c*sqrmet)*(-(etax*dudn + etay*dvdn)/(2.0d0*
+           c*metsq) + dpdn/(2.0d0*gamma*pc(i,nn)*sqrmet))
    else
c K factor from Poinsot and Lele
        scrh5 = bigk*(pc(i,nn) - pin)
    endif
c use script H values in integration to get new q's at exit
    con = rhoc(i,nn)*sqrmet
    etaxr = etax*rhoc(i,nn)
    etayr = etay*rhoc(i,nn)
    if(intmed) then
qbar(i,nn,1) = q(i,nn,1)-dt*(scrh2 + con*(scrh4+scrh5))/jac
qbar(i,nn,2) = q(i,nn,2)-dt*(etayr*(scrh1+scrh2+scrh3) +
+           uc(i,nn)*scrh2 + etaxr*c*(scrh4-scrh5) +
+           con*uc(i,nn)*(scrh4+scrh5))/jac
qbar(i,nn,3) = q(i,nn,3)-dt*(vc(i,nn)*scrh2 - etaxr*(scrh1+
+           scrh2+scrh3) + etayr*c*(scrh4-scrh5) + con*
+           vc(i,nn)*(scrh4+scrh5))/jac
qbar(i,nn,4) = q(i,nn,4)-dt*((etayr*uc(i,nn)-etaxr*vc(i,nn))*
+           (scrh1+scrh2+scrh3) + ((uc(i,nn)*uc(i,nn)+
+           vc(i,nn)*vc(i,nn))/2.0d0 - qheat*y2c(i,nn))*scrh2-
+           rhoc(i,nn)*qheat*scrh3 + rhoc(i,nn)*c*ucont*
+           (scrh4-scrh5) + con*bighc(i,nn)*(scrh4+scrh5))/jac
qbar(i,nn,5) = q(i,nn,5)-dt*(y2c(i,nn)*scrh2 + rhoc(i,nn)*
+           scrh3 + con*y2c(i,nn)*(scrh4+scrh5))/jac
    else
q(i,nn,1) = q(i,nn,1) - dt*(scrh2 + con*(scrh4+scrh5))/jac
q(i,nn,2) = q(i,nn,2) - dt*(etayr*(scrh1+scrh2+scrh3) +
+           uc(i,nn)*scrh2 + etaxr*c*(scrh4-scrh5) +
+           con*uc(i,nn)*(scrh4+scrh5))/jac
q(i,nn,3) = q(i,nn,3) - dt*(vc(i,nn)*scrh2 - etaxr*(scrh1+
+           scrh2+scrh3) + etayr*c*(scrh4-scrh5) + con*
+           vc(i,nn)*(scrh4+scrh5))/jac
q(i,nn,4) = q(i,nn,4) - dt*((etayr*uc(i,nn)-etaxr*vc(i,nn))*
+           (scrh1+scrh2+scrh3) + ((uc(i,nn)*uc(i,nn)+
+           vc(i,nn)*vc(i,nn))/2.0d0 - qheat*y2c(i,nn))*scrh2-
+           rhoc(i,nn)*qheat*scrh3 + rhoc(i,nn)*c*ucont*
+           (scrh4-scrh5) + con*bighc(i,nn)*(scrh4+scrh5))/jac
q(i,nn,5) = q(i,nn,5) - dt*(y2c(i,nn)*scrh2 + rhoc(i,nn)*
+           scrh3 + con*y2c(i,nn)*(scrh4+scrh5))/jac
    endif
    endif
100 continue

```

c

```
return  
end
```


Bibliography

- [1] J. K. Ahuja and S. N. Tiwari. A parametric study of shock-induced combustion in a hydrogen-air system. AIAA Paper 94-0674, January 1994.
- [2] J. K. Ahuja, S. N. Tiwari, and D. J. Singh. Investigation of hypersonic shock-induced combustion in a hydrogen-air system. AIAA Paper 92-0339, January 1992.
- [3] D. A. Anderson, J. C. Tannehill, and R. H. Pletcher. *Computational Fluid Mechanics and Heat Transfer*. Hemisphere Publishing Corporation, New York, 1984.
- [4] J. D. Anderson. *Hypersonic and High Temperature Gas Dynamics*. McGraw-Hill, New York, 1989.
- [5] Christopher A. Atwood. Computation of viscous blast wave flowfields. Final Report 91-002, MCAT Institute, San Jose, March 1991.
- [6] H. Behrens, W. Struth, and F. Weckens. Studies of hypervelocity firings into mixtures of hydrogen with air or with oxygen. In *Tenth Symposium (International) on Combustion*, pages 245–252, Pittsburgh, PA, 1965. The Combustion Institute.
- [7] D. W. Bogdanoff. Ram accelerator direct space launch system: New concepts. *Journal of Propulsion and Power*, 8(2):481–490, March-April 1992.
- [8] A. Bourlioux and A. J. Majda. Theoretical and numerical structure for unstable two-dimensional detonations. *Combustion and Flame*, 90:211–229, 1992.
- [9] A. Bourlioux, A. J. Majda, and V. Roytburd. Theoretical and numerical structure for unstable one-dimensional detonations. *SIAM Journal on Applied Mathematics*, 51(2):303–343, April 1991.
- [10] D. C. Brackett and D. W. Bogdanoff. Computational investigation of oblique detonation ramjet-in-tube concepts. *Journal of Propulsion and Power*, 5(3):276–281, May-June 1989.
- [11] J. Buckmaster. The structural stability of oblique detonation waves. *Combustion Science and Technology*, 72:283–296, August 1990.
- [12] J. Buckmaster and C. J. Lee. Flow refraction by an uncoupled shock and reaction front. *AIAA Journal*, 28(7):1310–1312, July 1990.
- [13] J. D. Buckmaster and G. S. S. Ludford. *Theory of Laminar Flames*. Cambridge University Press, New York, 1982.

- [14] J.-L. Cambier, H. G. Adelman, and G. P. Menees. Numerical simulations of oblique detonations in supersonic combustion chambers. *Journal of Propulsion and Power*, 5(4):482–491, July-August 1989.
- [15] J.-L. Cambier, H. G. Adelman, and G. P. Menees. Numerical simulations of an oblique detonation wave engine. *Journal of Propulsion and Power*, 6(3):315–323, May-June 1990.
- [16] R. Capiiaux and M. Washington. Nonequilibrium flow past a wedge. *AIAA Journal*, 1(3):650–660, March 1963.
- [17] S. R. Chakravarthy and S. Osher. *Computing with High-Resolution Upwind Schemes for Hyperbolic Equations*, volume 22 of *Lectures in Applied Mathematics*, pages 57–86. American Mathematical Society, Providence, 1985.
- [18] G. G. Chernyi. Supersonic flow past bodies with formation of detonation and combustion fronts. In *Problems of Hydrodynamics and Continuum Mechanics*, pages 145–169. Society of Industrial and Applied Mathematics, Philadelphia, 1969.
- [19] J. F. Clarke, S. Karni, J. J. Quirk, P. L. Roe, L. G. Simmonds, and E. F. Toro. Numerical computation of two-dimensional unsteady detonation waves in high energy solids. *Journal of Computational Physics*, 106(2):215–233, June 1993.
- [20] P. Colella, A. Majda, and V. Roytburd. Theoretical and numerical structure for reacting shock waves. *SIAM Journal on Scientific and Statistical Computing*, 7(4):1059–1080, October 1986.
- [21] P. Colella and P. R. Woodward. The piecewise parabolic method (PPM) for gas-dynamical systems. *Journal of Computational Physics*, 54(1):174–201, April 1984.
- [22] R. Courant and K. O. Friedrichs. *Supersonic Flow and Shock Waves*. Interscience Publishers, Inc., New York, 1948.
- [23] A. Dadone and B. Grossman. Characteristic-based, rotated upwind scheme for the Euler equations. *AIAA Journal*, 30(9):2219–2226, September 1992.
- [24] W. Doering. On detonation processes in gases. *Annalen der Physik*, 43:421–436, 1943.
- [25] R. Dunlap, R. L. Brehm, and J. A. Nicholls. A preliminary study of the application of steady-state detonative combustion to a reaction engine. *Jet Propulsion*, 28(7):451–456, 1958.
- [26] B. Engquist and S. Osher. One-sided difference approximations for nonlinear conservation laws. *Mathematics of Computation*, 36(154):321–352, April 1981.
- [27] W. Fickett and W. C. Davis. *Detonation*. University of California Press, Berkeley, 1979.
- [28] W. Fickett and W. W. Wood. Flow calculations for pulsating one-dimensional detonations. *The Physics of Fluids*, 9(5):903–916, May 1966.
- [29] T. Fujiwara and A. Matsuo. Oxyhydrogen oblique detonation supported by two-dimensional wedge. *Memoirs of the Faculty of Engineering, Nagoya University*, 41(2):295–302, 1989.

- [30] T. Fujiwara, A. Matsuo, and H. Nomoto. A two-dimensional detonation supported by a blunt body or wedge. *AIAA Paper* 88-0098, January 1988.
- [31] S. K. Godunov. A difference scheme for numerical computation of discontinuous solution of hydrodynamic equations. *Sbornik Mathematics*, 47:271–306, 1959.
- [32] M. J. Grismer and J. M. Powers. Comparison of numerical oblique detonation solutions with an asymptotic benchmark. *AIAA Journal*, 30(12):2985–2987, December 1992.
- [33] M. J. Grismer and J. M. Powers. Calculations for steady propagation of a generic ram accelerator configuration. *Journal of Propulsion and Power*, 11(1), Jan.-Feb. 1994.
- [34] R. A. Gross. Oblique detonation waves. *AIAA Journal*, 1(5):1225–1227, May 1963.
- [35] R. A. Gross and W. Chinitz. A study of supersonic combustion. *Journal of the Aero/Space Sciences*, 27(7):517–524, July 1960.
- [36] A. Harten and J. M. Hyman. Self adjusting grid methods for one-dimensional hyperbolic conservation laws. *Journal of Computational Physics*, 50(2):235–269, May 1983.
- [37] B. W. Henderson. Ram accelerator demonstrates potential for hypervelocity research, light launch. *Aviation Week & Space Technology*, 135:50–51, September 30, 1991.
- [38] A. Hertzberg, A. P. Bruckner, and D. W. Bogdanoff. Ram accelerator: A new chemical method for accelerating projectiles to ultrahigh velocities. *AIAA Journal*, 26(2):195–203, February 1988.
- [39] A. Hertzberg, A. P. Bruckner, and C. Knowlen. Experimental investigation of ram accelerator propulsion modes. *Shock Waves*, 1:17–25, March 1991.
- [40] C. Hirsch. *Numerical Computation of Internal and External Flows - Vol. 2: Computational Methods for Inviscid and Viscous Flows*. John Wiley & Sons, New York, 1990.
- [41] Institute Saint-Louis. *Proceedings of the First International Workshop on Ram Accelerator*, Saint-Louis, France, September 1993.
- [42] T. L. Jackson, A. K. Kapila, and M. Y. Hussaini. Convection of a pattern of vorticity through a reacting shock wave. *Physics of Fluids A*, 2(7):1260–1268, July 1990.
- [43] D. L. Kruczynski and M. J. Nusca. Experimental and computational investigation of scaling phenomena in a large caliber ram accelerator. *AIAA Paper* 92-3245, July 1992.
- [44] E. Larisch. Interactions of detonation waves. *Journal of Fluid Mechanics*, 6(3):392–400, October 1959.
- [45] D. G. Lasseigne, T. L. Jackson, and M. Y. Hussaini. Nonlinear interaction of a detonation/vorticity wave. *Physics of Fluids A*, 3(8):1972–1979, August 1991.

- [46] C. L. Leck and J. C. Tannehill. A new rotated upwind difference scheme for the Euler equations. AIAA Paper 93-0066, January 1993.
- [47] H. I. Lee and D. S. Stewart. Calculation of linear detonation instability: One-dimensional instability of plane detonation. *Journal of Fluid Mechanics*, 216:103–132, July 1990.
- [48] R. S. Lee. A unified analysis of supersonic nonequilibrium flow over a wedge: I. vibrational nonequilibrium. *AIAA Journal*, 2(4):637–646, April 1964.
- [49] H. F. Lehr. Experiments on shock-induced combustion. *Astronautica Acta*, 17(4 & 5):589–597, 1972.
- [50] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser Verlag, Boston, 2nd edition, 1992.
- [51] C. Li, K. Kailasanath, and E. S. Oran. Stability of oblique detonations in ram accelerators. AIAA Paper 92-0089, January 1992.
- [52] C. Li, K. Kailasanath, and E. S. Oran. Effects of boundary layers on oblique detonation-structures. AIAA Paper 93-0450, January 1993.
- [53] C. Li, K. Kailasanath, and E. S. Oran. Detonation structures behind oblique shocks. *Physics of Fluids*, 6(4):1600–1611, April 1994.
- [54] Y. Liu and M. Vinokur. Nonequilibrium flow computations. i. an analysis of numerical formulations of conservation laws. *Journal of Computational Physics*, 83(2):373–397, August 1989.
- [55] U. Maas and S. B. Pope. Simplifying chemical kinetics: Intrinsic low-dimensional manifolds in composition space. *Combustion and Flame*, 88(3 & 4):239–264, March 1992.
- [56] R. B. Morrison. Evaluation of the oblique detonation wave ramjet. Contractor Report 145358, NASA, January 1978.
- [57] A. K. Oppenheim, J. J. Smolen, and L. J. Zajac. Vector polar method for the analysis of wave intersections. *Combustion and Flame*, 12(1):63–76, February 1968.
- [58] T. J. Poinso and S. K. Lele. Boundary conditions for direct simulations of compressible viscous flows. *Journal of Computational Physics*, 101(1):104–129, July 1992.
- [59] J. M. Powers. Oblique detonations: Theory and propulsion applications. In J. Buckmaster, T. L. Jackson, and A. Kumar, editors, *Combustion in High-Speed Flows*. Kluwer Academic Publishers, 1994.
- [60] J. M. Powers, D. R. Fulton, K. A. Gonthier, and M. J. Grismer. Analysis for steady propagation of a generic ram accelerator/oblique detonation wave engine configuration. AIAA Paper 93-0243, January 1993.
- [61] J. M. Powers and K. A. Gonthier. Methodology and analysis for determination of propagation speed of high-speed devices. In *Proceedings of the Central States Section Spring 1992 Technical Meeting of the Combustion Institute*, pages 1–6, April 1992.

- [62] J. M. Powers and K. A. Gonthier. Reaction zone structure for strong, weak overdriven, and weak underdriven oblique detonations. *Physics of Fluids A*, 4(9):2082–2089, September 1992.
- [63] J. M. Powers and D. S. Stewart. Approximate solutions for oblique detonations in the hypersonic limit. *AIAA Journal*, 30(3):726–736, March 1992.
- [64] D. T. Pratt, J. W. Humphrey, and D. E. Glenn. Morphology of a standing oblique detonation wave. *Journal of Propulsion and Power*, 7(5):837–845, September–October 1991.
- [65] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in FORTRAN: the Art of Scientific Computing*. Cambridge University Press, New York, 2nd edition, 1992.
- [66] P. L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43(2):357–372, October 1981.
- [67] P. M. Rubins and R. P. Rhodes. Shock-induced combustion with oblique shocks: Comparison of experiment and kinetic calculations. *AIAA Journal*, 1(12):2778–2784, December 1963.
- [68] F. W. Rugg and W. W. Dorsey. A missile technique for the study of detonation waves. *Journal of Research of the National Bureau of Standards – C. Engineering and Instrumentation*, 66C(1):51–58, January–March 1962.
- [69] J. Rutkowski and J. A. Nicholls. Considerations for the attainment of a standing detonation wave. In D. K. Fleming, editor, *Proceedings of the Gas Dynamics Symposium on Aerothermochemistry*, pages 243–254, Evanston, Illinois, August 1955. Northwestern University.
- [70] J.-S. Shuen and S. Yoon. Numerical study of chemically reacting flows using a lower-upper symmetric successive overrelaxation scheme. *AIAA Journal*, 27(12):1752–1760, December 1989.
- [71] R. Siestrunk, J. Fabri, E. Le Grives, and J. Fabri. Some properties of stationary detonation waves. In *Fourth Symposium (International) on Combustion*, pages 498–501, Baltimore, 1953. Williams and Wilkins Co.
- [72] N. Sinha, B. J. York, S. M. Dash, R. Drabczuk, and G. E. Rolader. Progress towards the development of transient ram accelerator simulation as part of the U. S. Air Force armament directorate research program. AIAA Paper 92-3248, July 1992.
- [73] J. Srulijes, G. Smeets, and F. Seiler. Expansion tube experiments for the investigation of ram-accelerator-related combustion and gasdynamic problems. AIAA Paper 92-3246, July 1992.
- [74] D. S. Stewart and J. B. Bdzil. Asymptotics and multi-scale simulation in a numerical combustion laboratory. In H. G. Kaper and M. Garbey, editors, *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters*. Kluwer Academic Publishers, Boston, 1993.
- [75] G. Strang. On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis*, 5(3):506–517, September 1968.

- [76] K. W. Thompson. Time dependent boundary conditions for hyperbolic systems. *Journal of Computational Physics*, 68(1):1–24, January 1987.
- [77] G. Tivanov and J. Rom. Investigation of hypersonic flow of a detonable gas mixture ahead of a forward facing step. AIAA Paper 93-0611, January 1993.
- [78] W. G. Vincenti. Linearized flow over a wedge in a nonequilibrium oncoming stream. *Journal de Mecanique*, 1(2):193–212, June 1962.
- [79] J. von Neumann. Theory of detonation waves. In A. H. Taub, editor, *John von Neumann Collected Works*, pages 203–218. Macmillan Press, New York, 1963.
- [80] Y. Wada, S. Ogawa, and T. Ishiguro. A generalized Roe’s approximate Riemann solver for chemically reacting flows. AIAA Paper 89-0202, January 1989.
- [81] Y. Y. Wang, T. Fujiwara, T. Aoki, et al. Three-dimensional standing oblique detonation waves in a hypersonic flow. AIAA Paper 88-0478, January 1988.
- [82] R. C. Weast, editor. *CRC Handbook of Chemistry and Physics*. CRC Press, Inc., Boca Raton, Florida, 66th edition, 1985.
- [83] R. J. Weber and J. S. MacKay. Analysis of ramjet engines using supersonic combustion. Technical Note 4386, NACA, September 1958.
- [84] D. Whitfield and L. Taylor. Discretized newton-relaxation solution of high resolution flux-difference split schemes. AIAA Paper 91-1539, June 1991.
- [85] F. A. Williams. *Combustion Theory*. Benjamin/Cummings Pub. Co., Menlo Park, California, 2nd edition, 1985.
- [86] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54(1):115–173, April 1984.
- [87] S. Yungster. Numerical study of shock-wave/boundary-layer interactions in premixed combustible gases. *AIAA Journal*, 30(10):2379–2387, October 1992.
- [88] S. Yungster and A. P. Bruckner. Computational studies of a superdetonative ram accelerator mode. *Journal of Propulsion and Power*, 8(2):457–463, March-April 1992.
- [89] S. Yungster, S. Eberhardt, and A. P. Bruckner. Numerical simulation of hypervelocity projectiles in detonable gases. *AIAA Journal*, 29(2):187–199, February 1991.
- [90] Ya. B. Zeldovich. On the theory of the propagation of detonation in gaseous systems. *Zhurnal Eksperimental’noi i Teoreticheskoi Fiziki*, 10:542–568, 1940. (English translation: NACA TM 1261, 1960.).