



UNIVERSIDAD DE MÁLAGA

PROGRAMA DE DOCTORADO EN MATEMÁTICAS

FACULTAD DE CIENCIAS

DEPARTAMENTO DE ANÁLISIS MATEMÁTICO, ESTADÍSTICA E
INVESTIGACIÓN OPERATIVA Y MATEMÁTICA APLICADA

High-order Approximate Lax-Wendroff methods for systems of conservation laws

HUGO ALFREDO CARRILLO SERRANO

PHD THESIS

ADVISOR:

CARLOS MARÍA PARÉS MADROÑAL

UNIVERSIDAD DE MÁLAGA Mayo 2020


UNIVERSIDAD
DE MÁLAGA





UNIVERSIDAD
DE MÁLAGA

AUTOR: Hugo Alfredo Carrillo Serrano

 <http://orcid.org/0000-0002-7569-2270>

EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional:

<http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode>

Cualquier parte de esta obra se puede reproducir sin autorización pero con el reconocimiento y atribución de los autores.

No se puede hacer uso comercial de la obra y no se puede alterar, transformar o hacer obras derivadas.

Esta Tesis Doctoral está depositada en el Repositorio Institucional de la Universidad de Málaga (RIUMA): riuma.uma.es



D. Carlos María Parés Madroñal, Catedrático del Departamento de Análisis Matemático, Estadística e Investigación Operativa, y Matemática Aplicada de la Universidad de Málaga

Certifica:

Que D. Hugo Alfredo Carrillo Serrano ha realizado en dicho Departamento bajo mi dirección, el trabajo de investigación correspondiente a su Tesis Doctoral, titulado:

High order Approximate Lax-Wendroff methods for systems of conservation laws

Revisado el presente trabajo, estimo que puede ser presentado al Tribunal que ha de juzgarlo.

Y para que conste a efectos de lo establecido en el artículo octavo del Real Decreto 99/2011, autorizo la presentación de este trabajo en la Universidad de Málaga.

En Málaga, a 11 de mayo de 2020

PARES
MADROÑAL
CARLOS
MARIA -

Firmado digitalmente por PARES MADROÑAL CARLOS MARIA -
Fecha: 2020.05.11 09:35:26 +02'00'

Dr. Carlos María Pares Madroñal.



UNIVERSIDAD
DE MÁLAGA



Escuela de Doctorado

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR

D. HUGO ALFREDO CARRILLO SERRANO

Estudiante del programa de doctorado Matemáticas de la Universidad de Málaga, autor de la tesis, presentada para la obtención del título de doctor por la Universidad de Málaga, titulada:

High order Approximate Lax-Wendroff methods for systems of conservation laws

Realizada bajo la tutorización y dirección de Carlos Parés Madroñal

DECLARO QUE:

La tesis presentada es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, conforme al ordenamiento jurídico vigente (Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), modificado por la Ley 2/2019, de 1 de marzo.

Igualmente asumo, ante a la Universidad de Málaga y ante cualquier otra instancia, la responsabilidad que pudiera derivarse en caso de plagio de contenidos en la tesis presentada, conforme al ordenamiento jurídico vigente.

En Málaga, a 6 de mayo de 2020

Fdo.: Hugo Alfredo Carrillo Serrano

UNIVERSIDAD
DE MÁLAGA



EFQM AENOR



Edificio Pabellón de Gobierno. Campus El Ejido.
29071

Tel.: 952 13 10 28 / 952 13 14 61 / 952 13 71 10

E-mail: doctorado@uma.es



Dedicatoria

A mi esposa Bernice y mis hijos Clarissa, Andrea y Rodhac.

Agradecimientos

Es mi deber y sincero deseo agradecer a Carlos María Parés Madroñal por su asesoría, dedicación, paciencia, esfuerzo y sobre todo al valioso tiempo que dedico en esta memoria y a mi persona.

En particular agradezco a Manuel Castro, José María Gallardo y Cipriano Escalante por sus aportes y consejos. Al excelente equipo de profesionistas: Tomás Morales, Jorge Macias, Mari Luz Muñoz, José Manuel González, Sergio Ortega, Marc de la Asunción y Carlos Sánchez. Un agradecimiento enorme a mis compañeros del programa de doctorado: Kleiton Schneider, Juan Carlos González, Ernesto Guerrero, Ernesto Pimentel e Irene Gómez Bueno. Gracias a todos por todo, en especial por permitirme ser parte de la gran familia EDANYA.

A David Zorío, Giovanni Russo y Emanuelle Macca, por sus aportes en los materiales que componen esta investigación, además agradezco el apoyo y las atenciones brindadas en la vistas y estadías académicas.

A todo el personal de la facultad de ciencias y otros departamentos de la Universidad de Málaga que se involucraron en este proyecto.

Y finalmente al programa de investigación e innovación de la Unión Europea Horizonte 2020 que bajo el acuerdo de subvención Marie Skłodowska-Curie No. 642768 financiaron esta investigación.

Contents

List of figures	v
Resumen	x
Abstract	xxviii
1 Introduction	1
1.1 Motivation	1
1.2 Scope of this thesis	2
1.3 Outline	2
2 Preliminaries	5
2.1 Hyperbolic conservation laws	5
2.2 Numerical methods	8
2.2.1 Computational grids	8
2.2.2 Conservative methods	9
2.2.3 High-resolution conservative methods	11
2.2.3.1 ENO and WENO reconstructions	11
2.2.3.2 ENO and WENO algorithms	12
2.2.3.3 WENO conservative methods	14
2.2.3.4 High-order TVD Runge-Kutta time discretization	15
2.2.3.5 Lax-Wendroff type time discretizations	16
2.2.3.6 Lax-Wendroff procedure for nonlinear problems	18
3 Compact Approximate Taylor Method	21
3.1 The high-order Lax-Wendroff method for linear problems	22
3.1.1 Formulas of numerical differentiation	23
3.1.2 Conservative form	27
3.1.3 Computation of the coefficients: an iterative algorithm	27
3.1.4 Order of accuracy	28
3.1.5 Modified equation and stability	30
3.2 Extension to nonlinear problems	34
3.2.1 Approximate Taylor method	34
3.2.2 Compact Approximate Taylor method	35
3.2.3 Examples of CAT schemes	40
3.2.3.1 Second-order compact approximate Taylor method	40
3.2.3.2 Fourth-order compact approximate Taylor method	41
3.3 Shock-capturing techniques	45
3.3.1 Flux limiter - CAT methods	45



3.3.2	WENO-CAT methods	45
3.3.3	Systems of conservation laws	46
3.4	Numerical Experiments	47
3.4.1	1D scalar equations	47
3.4.1.1	Test 3.1 Transport equation - Discontinuous solutions 1	47
3.4.1.2	Test 3.2 Transport equation - Discontinuous solutions 2	47
3.4.1.3	Test 3.3 Transport equation - Accuracy order	49
3.4.1.4	Test 3.4 Burgers equation - Discontinuous solutions 1	49
3.4.1.5	Test 3.5 Burgers equation - Discontinuous solutions 2	50
3.4.1.6	Test 3.6 Burgers equation - Order of accuracy	52
3.4.2	1D Euler equations	52
3.4.2.1	Test 3.7 Order of accuracy	53
3.4.2.2	Test 3.8 Sod shock tube problem	54
3.4.2.3	Test 3.9 Shu-Osher problem	54
3.4.3	1D MHD equations	55
3.4.3.1	Test 3.10 Brio-Wu shock tube problem	57
3.4.3.2	Test 3.11 High mach shock tube problem	58
4	Adaptive Compact Approximate Taylor Method	63
4.1	Adaptive Compact Approximate Taylor Method	64
4.1.1	FL-CAT2 numerical flux	65
4.1.2	Smoothness indicators	66
4.1.3	ACAT2P methods	71
4.1.4	Systems of conservation laws	72
4.2	Two-dimensional problems	73
4.3	Numerical experiments	77
4.3.1	1D Scalar equations	77
4.3.1.1	Test 4.1 Transport equation - Smooth solutions	77
4.3.1.2	Test 4.2 Transport equation - Discontinuous solutions	79
4.3.1.3	Test 4.3 Burgers equation - Discontinuous solutions	80
4.3.2	1D Euler equations	82
4.3.2.1	Test 4.4 Sod shock tube problem	82
4.3.2.2	Test 4.5 123 Einfeldt problem	84
4.3.2.3	Test 4.6 Right blast wave problem of Woodward & Colella	84
4.3.3	2D Equations	87
4.3.3.1	Test 4.7 Transport equation	87
4.3.3.2	Test 4.8 - 4.10 Euler equations	87
5	Approximate Taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions	95
5.1	Approximate Taylor Methods	96

5.1.1	Lax-Wendroff Approximate Taylor Methods	96
5.1.2	Compact Approximate Taylor methods	99
5.2	Fast and optimal WENO reconstructions	100
5.3	FOWENO-AT Methods	104
5.4	Numerical experiments	104
5.4.1	Scalar conservation laws	105
5.4.1.1	Test 5.1 Transport equation	105
5.4.1.2	Test 5.2 Burgers equation	107
5.4.2	1D Systems of conservation laws	109
5.4.2.1	Test 5.3 Sod shock tube problem	110
5.4.2.2	Test 5.4 123 Einfeldt problem	110
5.4.2.3	Test 5.5 Left half of the blast wave problem	111
5.4.2.4	Test 5.6 Right half of the blast wave problem	111
5.4.2.5	Test 5.7 Blast wave problem	111
5.4.3	2D Systems of conservation laws	120
5.4.3.1	Test 5.8 - 5.13 Euler equations	122
5.5	Comparison of errors and efficiency	124
5.5.1	1D Scalar equations	125
5.5.1.1	Test 5.14. Linear transport equation	125
5.5.1.2	Test 5.15. Burgers Equation	126
5.5.2	Test 5.16. 1D Euler equations: the Sod shock tube problem	129
5.5.3	Test 5.17. 2D Euler equations: Lax configuration 6	129
6	Conclusions and future work	143
	Bibliography	147

List of Figures

3.1	Function $h_2(c)$ for $p = 1, \dots, 4$	33
3.2	Test 3.1. Transport equation with initial condition ((3.4.1)), CFL= 0.9 and $t = 1$. Solutions using CAT2 p methods with $p = 1, 2, 3, 4, 5$	48
3.3	Test 3.2. Transport equation with initial condition (3.4.1), CFL= 0.5 and $t = 1$. Left-top: general view. a, b, c and d : enlarged view of interest areas.	48
3.4	Test 3.4. Burgers equation with initial condition (3.4.1), CFL= 0.8, 0.4, 0.2, 0.1, 0.05 and $t = 1.2$. Solutions for CAT2 p , $p = 1, 2, 3, 4$. Left: general view. Right: enlarged view.	50
3.5	Test 3.5. Burgers equation with initial condition (3.4.1), CFL= 0.5 and $t = 1.2$. Top: general view. Bottom: flux limiter function $\varphi_{i+1/2}$ for FL-CAT2.	51
3.6	Test 3.5. Burgers equation with initial condition (3.4.1), CFL= 0.5 and CFL= 0.9, $t = 1.2$ and $t = 12$: enlarged view of the numerical results. Left-top: CFL= 0.5 and $t = 1.2$. Left-bottom: CFL= 0.5 and $t = 12$. Right-top: CFL= 0.9 and $t = 1.2$. Right-bottom: CFL= 0.9 and $t = 12$	52
3.7	Test 3.8. The Sod shock tube problem, CFL= 0.5 and $t = 0.25$. Left-top: general view of numerical solutions for density ρ and φ_{i+2}^ρ . Left-bottom: general view of numerical solutions for the internal energy and φ_{i+2}^E . Right-top: general view of numerical solutions for velocity u and φ_{i+2}^{pu} FL-CAT2. Right-down: general view of numerical solutions for the pressure p	55
3.8	Test 3.8. The Sod shock tube problem, CFL= 0.5 and $t = 0.25$. General view and enlarge view of the numerical results for ρ close to regions a, b, c, d	56
3.9	Test 3.8. The Sod shock tube problem, CFL= 0.5 and $t = 0.25$. General view and enlarge view of the numerical solutions for internal energy e close to regions a, b, c, d	56
3.10	Test 3.9. The Shu-Osher problem, CFL= 0.5 and $t = 1$. Left: general view of numerical solutions for density. Right-top: enlarged view. Right-bottom: enlarged view.	57
3.11	Test 3.10. The Brio-Wu shock tube problem, CFL= 0.8 and $t = 0.2$. Numerical solutions for ρ, v_x, B_y, p	58
3.12	Test 3.10. The Brio-Wu shock tube problem, CFL= 0.8 and $t = 0.2$. Enlarged view for ρ	59



3.13	Test 3.10. The Brio-Wu shock tube problem, CFL= 0.8 and $t = 0.2$. Enlarged view for v_x	59
3.14	Test 3.11. The high mach shock problem, CFL= 0.5 and $t = 0.012$. Numerical solutions for ρ, v_x, B_y, p	60
3.15	Test 3.11. The high mach shock problem, CFL= 0.8 and $t = 0.012$. General view of the numerical solutions provided by WENO5-CAT4 and WENO5- RK3 for ρ, v_x, B_y and p	61
3.16	Test 3.11. 1D MHD equations with the High mach shock tube problem, CFL= 0.8 and $t = 0.012$. Left: enlarged views for ρ . Right: enlarged views for B_y	61
4.1	Stencil S_2 centered in $\mathbf{x}_{1/2} = (0.5\Delta x, 0.5\Delta y)$	74
4.2	Test 4.1. Transport equation with initial condition (4.3.2). Numerical solution at $t = 4$: general view (<i>left-top</i>); local order of accuracy for ACAT6 (<i>sub-frame</i>); consecutive zooms close to the local maximum (<i>left-bottom</i> , <i>right-top</i> and <i>right-bottom</i>).	78
4.3	Test 4.1 Transport equation with initial condition (4.3.2). Numerical solution at $t = 40$: general view (<i>left-top</i>); local order of accuracy for ACAT6 (<i>sub-frame</i>); consecutive zooms close to the local maximum (<i>left- bottom</i> , <i>right-top</i> and <i>right-bottom</i>).	78
4.4	Test 4.1. Transport equation with initial condition (4.3.3). Solution obtained with ACAT6 at time 4 (<i>top</i>) and graphs of the smoothness indicators ψ_{sb} , ψ^2 and ψ^3 (<i>bottom</i>).	79
4.5	Test 4.2. Transport equation with initial condition (4.3.4). Numerical solutions at $t = 2$ (<i>a</i>) and $t = 20$ (<i>b</i>). Zoom of the numerical solutions at time $t = 2$ (<i>c</i>) and $t = 20$ (<i>d</i>). Sub-frames: local order of accuracy for ACAT6.	80
4.6	Test 4.3. Burgers equation with initial condition (4.3.2). Numerical solutions obtained at times $t = 0.25$ (<i>left-top</i>), $t = 0.5$ (<i>right-top</i>), $t = 1$ (<i>left-bottom</i>), and $t = 10$ (<i>right-bottom</i>). Sub-frames: local order of accuracy for ACAT6.	81
4.7	Test 4.3. Burgers equation with initial condition (4.3.2). Zoom of the numerical solutions obtained at times $t = 0.25$ (<i>a</i>), $t = 0.5$ (<i>b</i>), $t = 0.1$ (<i>c</i>), and $t = 10$ (<i>d</i>). Sub-frames: local accuracy order for ACAT6.	81
4.8	Test 4.4. 1D Euler equations: the Sod problem. Numerical solutions at $t = 0.25$ using CFL= 0.8 and 200 points: density (<i>left-top</i>), velocity (<i>right- top</i>), internal energy (<i>left-bottom</i>), pressure (<i>right-bottom</i>). Sub-frames: local order of accuracy for ACAT6.	83
4.9	Test 4.4. 1D Euler equations: the Sod problem. Numerical density at $t = 0.25$ using CFL= 0.8 and 200 points: general view and zooms close to the points <i>a, b, c</i> and <i>d</i>	83



4.10	Test 4.4 1D Euler equations: the Sod problem. Numerical internal energy at $t = 0.25$ using CFL= 0.8 and 200 points: general view and zooms close to the points a, b, c and d . 1D Euler equations.	84
4.11	Test 4.5. 1D Euler equations: the 123 Einfeldt problem using CFL= 0.8 and 200 points. Density obtained with ACAT6 and graph of the smoothness indicator ψ^3 for $t = t_s/4$ (<i>left-top</i>), $t_s/2$ (<i>right-top</i>), $3t_s/4$ (<i>left-bottom</i>), t_s (<i>right-bottom</i>), with $t_s = 0.15$	85
4.12	Test 4.5. 1D Euler equations: the 123 Einfeldt problem using CFL= 0.8 and 200 points. Numerical densities at time $t = 0.15$: general view (<i>left-top</i>) and zooms close to the points a (<i>left-bottom</i>), b (<i>right-top</i>), and c (<i>right-bottom</i>).	85
4.13	Test 4.6. 1D Euler equations: right blast wave of the Woodward & Colella problem. Numerical densities at time $t = 0.012$, using CFL= 0.8 (<i>left</i>) and zooms close to the shocks (<i>center</i> and <i>right</i>).	86
4.14	Test 4.7. 2D Transport equation: solution obtained with ACAT2, ACAT4, WENO3 RK3 and WENO5 RK3 at time $t = 1$: cut with a vertical plane passing through the line $y = x$. Subplot: zoom close to the discontinuity	88
4.15	Test 4.8. 2D Euler equations: contour plots of the density at time $t = 0.25$ obtained with ACAT2 (<i>left-top</i>) and ACAT4 (<i>right-top</i>). Contour plots of the smoothness indicators ψ_x^1 (<i>left-center</i>), ψ_x^2 (<i>right-center</i>), ψ_y^1 (<i>left-bottom</i>) and ψ_y^2 (<i>right-bottom</i>).	91
4.16	Test 4.9. 2D Euler equations: contour plots of the density at time $t = 0.3$ obtained with ACAT2 (<i>left-top</i>) and ACAT4 (<i>right-top</i>). Contour plots of the smoothness indicators ψ_x^1 (<i>left-center</i>), ψ_x^2 (<i>right-center</i>), ψ_y^1 (<i>left-bottom</i>) and ψ_y^2 (<i>right-bottom</i>).	92
4.17	Test 4.10. 2D Euler equations: contour plots of the density at time $t = 0.25$ obtained with ACAT2 (<i>left-top</i>) and ACAT4 (<i>right-top</i>). Contour plots of the smoothness indicators ψ_x^1 (<i>left-center</i>), ψ_x^2 (<i>right-center</i>), ψ_y^1 (<i>left bottom</i>) and ψ_y^2 (<i>right-bottom</i>).	93
4.18	Test 4.10 2D Euler equations: contour plots of the density at time $t = 0.25$ obtained with ACAT2 (<i>left-top</i>), ACAT4 (<i>right-top</i>), WENO3 RK3 (<i>left-bottom</i>) and WENO5 RK3 (<i>right-bottom</i>).	94
5.1	Test 5.1. Transport equation with initial conditions (5.4.2), CFL= 0.5 and $t = 2s$. Methods based on 3rd-order reconstructions: general view (<i>top</i>) and zoom of the areas of interest (<i>bottom</i>).	106
5.2	Test 5.1. Transport equation with initial conditions (5.4.2), CFL= 0.5 and $t = 2s$. Methods based on 5th-order reconstructions: general view (<i>top</i>) and zoom of the areas of interest (<i>bottom</i>).	107



5.3	Test 5.1 Transport equation with initial conditions (5.4.2), and $t = 2s$. Methods based on 7th-order reconstructions with CFL= 0.5: general view (<i>top</i>) and zoom of the areas of interest (<i>bottom</i>).	108
5.4	Test 5.1. Transport equation with initial conditions (5.4.2), and $t = 2s$. Methods based on 5th-order reconstructions with CFL= 0.9: general view (<i>top</i>) and zoom of the areas of interest (<i>bottom</i>).	109
5.5	Test 5.2. Burgers equation with initial conditions (5.4.3), CFL= 0.5 and $t = 2s$. Row 1: methods based on 5th-order reconstructions: general view. Rows 2-4: zooms of an area of interest.	110
5.6	Test 5.3. 1D Euler equations. Sod problem: density. Row 1: exact solution (<i>left</i>), methods using 3rd-order (<i>center</i>) and 5th-order (<i>right</i>) reconstruction operators. Rows 2-4: zooms corresponding to areas a , b and c . CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	112
5.7	Test 5.3. 1D Euler equations. Sod problem: internal energy. Methods using 3rd order (<i>row 1</i>), 5th order (<i>row 2</i>), and 7th order (<i>row 3</i>) reconstruction operators. Left: general view. Right: zoom of an area of interest. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th order reconstructions respectively.	113
5.8	Test 5.4. 1D Euler equations. 123 Einfeldt problem: density. Row 1: exact solution (<i>left</i>), methods using 3rd (<i>center</i>) and 5th order (<i>right</i>) reconstruction operators. Rows 2-4: zooms corresponding to areas a , b and c . CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	114
5.9	Test 5.4. 1D Euler equations. 123 Einfeldt problem: internal energy. Methods using 3rd-order (<i>row 1</i>), 5th-order (<i>row 2</i>), and 7th-order (<i>row 3</i>) reconstruction operators. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	115
5.10	Test 5.5. 1D Euler equations. Left half of the blast wave problem of Woodward and Colella: density. Row 1: exact solution (<i>left</i>), methods using 3rd (<i>center</i>) and 5th-order (<i>right</i>) reconstruction operators. Rows 2-4: zooms corresponding to areas a , b and c . CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	116
5.11	Test 5.5. 1D Euler equations. Left half of the blast wave problem of Woodward and Colella: internal energy. Methods using 3d-order (<i>row 1</i>), 5th-order (<i>row 2</i>), and 7th-order (<i>row 3</i>) reconstruction operators. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	117

5.12	Test 5.6. 1D Euler equations. Right half of the blast wave problem of Woodward and Colella: density. Row 1: exact solution (<i>left</i>), methods using 3rd-order (<i>center</i>) and 5th-order (<i>right</i>) reconstruction operators. Rows 2-4: zooms corresponding to areas <i>a</i> , <i>b</i> and <i>c</i> . CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	118
5.13	Test 5.6. 1D Euler equations. Right half of the blast wave problem of Woodward and Colella: internal energy. Methods using 3rd-order (<i>row 1</i>), 5th-order (<i>row 2</i>) and 7th-order (<i>row 3</i>) reconstruction operators. Left: general view. Right: zoom of an area of interest. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	119
5.14	Test 5.7. 1D Euler equations. Woodward and Colella problem: density. Row 1: exact solution (<i>left</i>), methods using 3rd-order (<i>center</i>) and 5th-order (<i>right</i>) reconstruction operators. Rows 2-4: zooms corresponding to areas <i>a</i> , <i>b</i> and <i>c</i> . CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	120
5.15	Test 5.7. 1D Euler equations. Woodward and Colella problem: internal energy. Methods using 3rd-order (<i>row 1</i>), 5th-order (<i>row 2</i>) and 7th-order (<i>row 3</i>) reconstruction operators. Left: general view. Right: zoom of an area of interest. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.	121
5.16	Test 5.8. 2D Euler equations. Lax configuration 3: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.	133
5.17	Test 5.9. 2D Euler equations. Lax configuration 6: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.	134
5.18	Test 5.9. 2D Euler equations. Lax configuration 6: density computed with WENO-RK, WENO-CAT and WENO-LAT.	135
5.19	Test 5.10. 2D Euler equations. Lax configuration 11: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.	136
5.20	Test 5.11. 2D Euler equations. Lax configuration 13: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.	137
5.21	Test 5.12. 2D Euler equations. Lax configuration 17: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.	138
5.22	Test 5.13. 2D Euler equations. Lax configuration 19: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.	139
5.23	Test 5.14. Linear transport equation with initial condition 5.5.2: efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 4$ and CFL= 0.5. Second and third-order methods (<i>left</i>) and fourth or fifth-order methods (<i>right</i>).	140



5.24	Test 5.14. Linear transport equation with initial condition 5.5.2: efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 4$ and CFL= 0.9. Second or third-order methods (<i>left</i>) and fourth or fifth-order methods (<i>right</i>).	140
5.25	Test 5.15. Burgers equation with initial conditions (5.5.3): efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 1.25$ and CFL= 0.5. Second and third-order methods (<i>left</i>) and fourth or fifth-order methods (<i>right</i>).	141
5.26	Test 5.15. Burgers equation with initial conditions (5.5.3): efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 1.25$ and CFL= 0.9. Second and third-order methods (<i>left</i>) and fourth or fifth-order methods (<i>right</i>).	141

Resumen

Métodos Lax-Wendroff aproximados para sistemas de leyes de conservación

Capítulo 1. Introducción

Motivación

Los métodos Lax-Wendroff para sistemas lineales de leyes de conservación avanzan en el tiempo mediante desarrollos de Taylor en el que las derivadas temporales se transforman en derivadas espaciales usando la ecuación: ver [1], [2], [3], [4]. Las derivadas espaciales son entonces discretizadas por medio de fórmulas centradas de diferenciación de alto orden. Este procedimiento permite derivar métodos numéricos de orden arbitrario.

Esta tesis se centra en la extensión de los métodos Lax-Wendroff a sistemas no lineales de leyes de conservación. La principal dificultad para extender los métodos Lax-Wendroff a problemas no lineales proviene de la transformación de las derivadas temporales en derivadas espaciales. Una primera estrategia para hacerlo es el procedimiento de Cauchy-Kovalevsky (CK). El principal inconveniente de este proceso es que conduce a expresiones cuyo número de términos crece exponencialmente, tiene alto costo computacional y es difícil de implementar. En el contexto de los métodos ADER introducidos por Toro y colaboradores (véase [5], [6], [7]), esta dificultad ha sido salvada mediante la sustitución del procedimiento CK por la resolución de problemas espacio-temporales locales usando un método de Galerkin: ver [8], [9].

Una alternativa al procedimiento CK y a la resolución de problemas locales ha sido introducida recientemente en [10]), en el que se sigue una estrategia basada en métodos de Taylor aproximados(AT): las derivadas temporales se aproximan utilizando fórmulas de diferenciación numéricas centradas combinadas con desarrollos de Taylor en el tiempo que se calculan de manera recursiva. Sin embargo, los esquemas AT no son generalizaciones genuinas de los métodos Lax-Wendroff, ya que son necesarios stencils de $(4p + 1)$ -puntos para conseguir métodos de orden p en lugar de $(2p + 1)$ como ocurre en el caso lineal. Este aumento en el tamaño del stencil conlleva además peores propiedades de estabilidad que

los métodos Lax-Wendroff originales.

No obstante, los métodos AT pueden ser estabilizados mediante el uso de reconstrucciones WENO en el cálculo de las derivadas en tiempo de primer orden, como en [11]. Los métodos resultantes suelen dar buenos resultados bajo una condición de tipo $CFL = 0.5$. Estos métodos son fáciles de implementar en mallas uniformes cartesianas.

Objetivos de la tesis

Los objetivos principales de este trabajo son los siguientes:

- Desarrollar una familia de métodos numéricos de alto orden para sistemas no lineales de leyes de conservación basados en un procedimiento aproximado de Taylor (AT) que constituya una generalización genuina de los métodos Lax-Wendroff, i.e. que se reduzcan a los métodos de alto orden Lax-Wendroff cuando el flujo es lineal.
- Combinar este nuevo procedimiento AT con técnicas de captura de choque conocidas y/o desarrollar técnicas propias y adecuadas a este nuevo método.

Organización de la memoria

Estos objetivos se han cumplido satisfactoriamente en tres artículos. El primero, titulado *Métodos Aproximados de Taylor Compactos*, fue publicado en 2019 en la revista *Journal of Scientific Computing*, véase [12]. Este artículo introduce una variante de los procedimientos AT que constituye una extensión genuina de alto orden de los métodos Lax-Wendroff a los sistemas no lineales de leyes de conservación.

Una vez derivados y probados los nuevos métodos, se abordó el segundo objetivo: encontrar formas efectivas y apropiadas de evitar oscilaciones cercanas a discontinuidades o choques. En colaboración con G. Russo, E. Macca (Universidad de Catania, Italia) y D. Zorío (Universidad de Concepción, Chile) presentamos una nueva familia de métodos numéricos que se basan en el uso de una adaptación local del orden del esquema en función de la suavidad de la solución numérica en función de una nueva familia de indicadores de suavidad. El documento resultante, titulado *Métodos Aproximados de Taylor Compactos Adaptativos*, está disponible en el repositorio *arXiv* y será enviado en breve para su publicación [13].

El último trabajo, titulado *Métodos Aproximados de Taylor con reconstrucciones de tipo WENO rápidas y optimizadas* (ver [14]) fue desarrollado en colaboración con D. Zorío (Universidad de Concepción, Chile). Este artículo también está disponible en el repositorio *arXiv* y fue enviado en Febrero de 2020 a la revista científica *Journal of Scientific Computing*.

El contenido de esta tesis se compone principalmente de las tres publicaciones mencionadas anteriormente y esta organizado como sigue:

El Capítulo 2 contiene los conceptos preliminares y la notación que consideramos importantes y/o necesarios para entender los capítulos subsiguientes (en este resumen no se incluyen). El Capítulo 3 se compone esencialmente del contenido del artículo *Métodos Aproximados de Taylor Compactos*. El capítulo 4 está dedicado al texto del artículo *Métodos Aproximados de Taylor Compactos Adaptativos* o CAT. El Capítulo 5 se centra en el texto del artículo *Métodos Aproximados de Taylor con reconstrucciones de tipo WENO rápidas y optimizadas*. Finalmente las conclusiones y trabajo futuro se presentan en el Capítulo 6.

Capítulo 2. Preliminares

En este capítulo, revisamos hechos básicos sobre los sistemas hiperbólicos de leyes de conservación y algunos métodos numéricos de alto orden para resolverlos. En particular, nos centramos en los métodos de alto orden que proporcionan antecedentes pertinentes para los capítulos siguientes.

Capítulo 3. Métodos Aproximados de Taylor Compactos

En este capítulo se presenta una nueva familia de métodos numéricos (en diferencias finitas) para la solución de sistemas de conservación no lineales. Dichos métodos son una variante compacta de los métodos de tipo Taylor Aproximados AT, denominados Métodos Aproximados de Taylor Compactos (CAT).

Con el fin de simplificar la explicación de los métodos nos centraremos en su aplicación a leyes de conservación escalares:

$$u_t + f(u)_x = 0, \quad u(x, t), x \in \mathcal{R}, t \geq 0. \quad (0.0.1)$$

La forma genérica de los métodos de tipo Taylor es la siguiente:

$$u_i^{n+1} = u_i^n + \sum_{k=1}^m \frac{\Delta t^k}{k!} u_i^{(k)}, \quad (0.0.2)$$

donde $u_i^{(k)}$ representa una aproximación de la derivada temporal $\partial_t^k u(x_i, t_n)$. Esta expresión se obtiene al aplicar un desarrollo de Taylor de grado m en el tiempo. En los métodos de tipo Lax-Wendroff el procedimiento seguido para aproximar las derivadas temporales se basa en su sustitución por derivadas espaciales usando la ecuación seguida de una aproximación de las derivadas espaciales usando fórmulas de derivación numérica de alto orden.

Métodos de alto orden Lax Wendroff

Cuando el problema (0.0.1) es lineal, es decir si $f(u) = au$, siendo a un número real, es fácil sustituir las derivadas temporales por derivadas espaciales derivando reiteradamente la ecuación:

$$\partial_t^k u = (-1)^k a^k \partial_x^k u, \quad k = 1, 2, \dots$$

A continuación, las derivadas espaciales son aproximadas con la fórmula de derivación numérica interpoladora centrada de $(2p + 1)$ puntos

$$f^{(k)}(x_i) \simeq D_{p,i}^k(f, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f(x_{i+j}), \quad (0.0.3)$$

donde $f^{(k)}$ representa la k -ésima derivada de la función f (con el convenio $f^{(0)} = f$) y $\delta_{p,j}^k$ son los coeficientes de la fórmula de derivación. La expresión del método numérico resultante es

$$u_i^{n+1} = u_i^n + \sum_{k=1}^m \frac{(-1)^k c^k}{k!} \sum_{j=-p}^p \delta_{p,j}^k u_{i+j}^n, \quad (0.0.4)$$

donde $\{x_i\}$ son los nodos de una malla uniforme de tamaño de paso Δx ; u_i^n es una aproximación de los valores puntuales x_i en el tiempo $n\Delta t$, donde Δt es el paso del tiempo; $p \geq 1$ es un número natural y $c = a\Delta t/\Delta x$.

Formulas de derivación numérica

Antes de extender el método (0.0.4) a problemas no lineales, se estudian en el Capítulo 2 algunas propiedades importantes de los coeficientes de las fórmulas de derivación numérica que serán utilizadas:

- Además de (0.0.3), usaremos la siguiente familia de fórmulas interpolatorias de derivación numérica de $2p$ puntos:

$$f^{(k)}(x_i + q\Delta x) \simeq A_{p,i}^{k,q}(f, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} f(x_{i+j}),$$

i.e. $A_{p,i}^{k,q}(f, \Delta x)$ es la fórmula interpoladora que aproxima la k -ésima derivada en el punto $x_i + q\Delta x$ usando los valores de la función en los $2p$ puntos $x_{i-p+1}, \dots, x_{i+p}$. Obsérvese que los coeficientes, como en (0.0.3), no dependen de i .

- Dada una variable w , la siguiente notación será usada:

$$D_{p,i}^k(w_*, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k w_{i+j},$$

$$A_{p,i}^{k,q}(w_*, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} w_{i+j},$$

para indicar que las fórmulas son aplicadas a las aproximaciones de w , w_i , y no a sus valores puntuales exactos $w(x_i)$. El símbolo $*$ se usara para indicar si se deriva respecto al tiempo ó al espacio. Esto es:

$$\partial_x^k u(x_i + q\Delta x, t_n) \simeq A_{p,i}^{k,q}(u_*^n, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} u_{i+j}^n, \quad (0.0.5)$$

$$\partial_t^k u(x_i, t_n + q\Delta t) \simeq A_{p,n}^{k,q}(u_i^*, \Delta t) = \frac{1}{\Delta t^k} \sum_{r=-p+1}^p \gamma_{p,r}^{k,q} u_i^{n+r}. \quad (0.0.6)$$

Con esta notación el algoritmo (0.0.4) se escribe como sigue:

$$u_i^{n+1} = u_i^n + \sum_{k=1}^m \frac{(-1)^k a^k \Delta t^k}{k!} D_{p,i}^k(u_*^n, \Delta x). \quad (0.0.7)$$

El método numérico (0.0.4) puede ser escrito en forma conservativa usando la igualdad

$$D_{p,i}^k(f, \Delta x) = \frac{1}{\Delta x} \left(A_{p,i}^{k-1,1/2}(f, \Delta x) - A_{p,i-1}^{k-1,1/2}(f, \Delta x) \right),$$

que se demuestra en [12]. En efecto, el uso de esta igualdad permite reescribir el método en la forma

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} \left(F_{i-1/2}^p - F_{i+1/2}^p \right), \quad (0.0.8)$$

con

$$F_{i+1/2}^p = \sum_{k=1}^{2p} (-1)^{k-1} \frac{a^k \Delta t^{k-1}}{k!} A_{p,i}^{k-1,1/2}(u_*^n, \Delta x). \quad (0.0.9)$$

En [12] se estudia con detalle el orden del método (0.0.4) y se ve que la combinación óptima entre el grado del desarrollo de Taylor en tiempo y el número de puntos de las fórmulas de derivación numérica se obtiene con la elección $m = 2p$, que conduce a métodos de orden $2p$. Se estudia además la estabilidad de estos métodos óptimos mediante técnicas de Fourier, llegándose a que son estables con la condición CFL-1, es decir si:

$$|a| \frac{\Delta t}{\Delta x} \leq 1.$$

Método Compacto Taylor Aproximado

Pasamos a la extensión de (0.0.4) a leyes de conservación no lineales. En los métodos aproximados de Taylor (LAT) introducidos en [10], la sustitución de las derivadas temporales por derivadas espaciales se basa en las identidades:

$$\partial_t^k u = -\partial_x^1 \partial_t^{k-1} f(u), \quad k = 1, 2 \dots m. \quad (0.0.10)$$

Así, para aproximar las derivadas temporales, se obtienen en primer lugar aproximaciones

$$\tilde{f}_i^{(k-1)} \cong \partial_t^{k-1} f(u)(x_i, t_n),$$

a las que se aplica, en segundo lugar, una fórmula centrada de $(2p+1)$ puntos, obteniéndose así las aproximaciones de las derivadas temporales

$$\tilde{u}_i^{(k)} = -D_{p,i}^1(\tilde{f}_*^{(k-1)}, \Delta x)$$

que permiten avanzar en el tiempo usando (0.0.2).

El cálculo de las aproximaciones $\tilde{u}_i^{(k)}$ y $\tilde{f}_i^{(k-1)}$ se lleva a cabo de forma recursiva: una vez conocidas $u_i^{(l)}$, $l = 0, \dots, k-1$, se utiliza un desarrollo de Taylor en tiempo de grado $k-1$ para obtener aproximaciones de $f(u(x_i, (n+r)\Delta t))$, $r = -p, \dots, p$; se aplica la fórmula centrada de diferenciación numérica con $(2p+1)$ puntos para obtener $\tilde{f}_i^{(k-1)}$; finalmente se aplica la fórmula $D_{p,i}^1$ a las aproximaciones obtenidas $\tilde{f}_{i+j}^{(k-1)}$, $j = -p, \dots, p$ para calcular $\tilde{u}_i^{(k)}$.

A diferencia de este procedimiento, los métodos CAT se basan en la escritura conservativa de los métodos de Lax-Wendroff (0.0.8): el flujo numérico CAT $F_{i+1/2}^p$ se calcula usando solo las aproximaciones

$$u_{i-p+1}^n, \dots, u_{i+p}^n, \quad (0.0.11)$$

Esto asegura que los valores usados para actualizar u_i^{n+1} son únicamente aquellos contenidos en el stencil centrado de $(2p+1)$, como ocurre en el caso lineal. De hecho, esta propiedad les permite ser una generalización genuina del método Lax-Wendroff para los problemas lineales.

Para poder calcular los flujos numéricos usando únicamente (0.0.11), para cada i se calculan aproximaciones *locales* de

$$\partial_t^{k-1} f(u(x_{i-p+1}, t^n), \dots, \partial_t^{k-1} f(u(x_{i+p}, t^n),$$

que serán representadas por

$$\tilde{f}_{i,j}^{(k-1)} \cong \partial_t^{k-1} f(u)(x_{i+j}, t_n), \quad j = -p+1, \dots, p.$$

Estas aproximaciones son locales en el sentido que $i_1 + j_1 = i_2 + j_2$, no implica que $\tilde{f}_{i_1, j_1}^{(k-1)} = \tilde{f}_{i_2, j_2}^{(k-1)}$. Una vez estas aproximaciones han sido calculadas, el flujo numérico viene dado por

$$F_{i+1/2}^p = \sum_{k=1}^{2p} \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x),$$

con

$$A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x) = \sum_{j=-p+1}^p \gamma_{p,j}^{0,1/2} \tilde{f}_{i,j}^{(k-1)}.$$

El cálculo de las aproximaciones $\tilde{f}_{i,j}^{(k-1)}$ se lleva a cabo nuevamente mediante un procedimiento recursivo basado en el uso de desarrollos de Taylor de grado creciente:

- $k = 1$: calcule $\tilde{f}_{i,j}^{(0)} = f(u_{i+j}^n)$, $j = -p + 1, \dots, p$.

- Para $k = 2 \dots 2p$:

- Calcule

$$\tilde{u}_{i,j}^{(k-1)} = -A_{p,0}^{1,j}(\tilde{f}_{i,*}^{(k-2)}, \Delta x).$$

- Calcule

$$\tilde{f}_{i,j}^{k-1,n+r} = f\left(u_{i+j}^n + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} \tilde{u}_{i,j}^{(l)}\right), \quad j, r = -p + 1, \dots, p.$$

- Calcule

$$\tilde{f}_{i,j}^{(k-1)} = A_{p,n}^{k-1,0}(\tilde{f}_{i,j}^{k-1,*}, \Delta t), \quad j = -p + 1, \dots, p.$$

Obsérvese que, mientras que en los métodos LAT todas las derivadas son aproximadas usando una formula centrada de $(2p + 1)$ puntos, en este algoritmo el stencil $x_{i-p+1}, \dots, x_{i+p}$ es usado para las derivadas en el espacio y el stencil $t_{n-p+1}, \dots, t_{n+p}$ para las derivadas en el tiempo.

En el Capítulo 3 se demuestra que, si se toma $m = 2p$, el método correspondiente CAT $2p$

- se reduce a (0.0.4) cuando $f(u) = au$;
- es linealmente L^2 -estable bajo la condición CFL

$$\max_i (|f'(u_i)|) \frac{\Delta t}{\Delta x} \leq 1;$$

- es de orden de precision $2p$.

Extender el procedimiento anterior a sistemas de conservación es simple, basta con repetir el proceso anterior variable a variable.

Aunque los métodos CAT son linealmente L^2 -estables bajo la condición usual $CFL-1$, pueden producir fuertes oscilaciones en las proximidades de una discontinuidad de la solución. Dos técnicas son introducidas en este capítulo para eliminar las oscilaciones espúreas. En primer lugar se introduce el método FL-CAT2 que combina el método CAT2 con un método robusto de primer orden usando un limitador de flujo estándar. En segundo lugar se introducen los métodos CAT-WENO, en los que, siguiendo a los autores de [10], se utilizan Weighted Essentially Non-Oscillatory (WENO) reconstrucciones espaciales (ver [15], [16]) para calcular la primera derivada en tiempo.

Capítulo 4. Métodos Aproximados de Taylor Compactos Adaptativos

En este capítulo se extiende a cualquier orden la estrategia seguida en el anterior para evitar oscilaciones espúreas combinando los flujos numéricos CAT con uno de primer orden robusto. La estrategia consiste en seleccionar automáticamente el stencil utilizado para calcular $F_{i+1/2}$ a fin de que su longitud sea máxima entre aquellos para los que la solución es suave. Específicamente, supongamos que las soluciones en el tiempo $n\Delta t$ $\{u_i^n\}$ han sido ya calculado. La longitud máxima del stencil para calcular $F_{i+1/2}$ se establece en, digamos, $2P$, donde P es un número natural. Por tanto, los stencils candidatos para calcular $F_{i+1/2}$ son

$$S_p = \{x_{i-p+1}, \dots, x_{i+p}\}, \quad p = 1, \dots, P.$$

Para seleccionar el stencil, se introducen una nueva familia de indicadores de suavidad $\psi_{i+1/2}^p$, $p = 1, \dots, P$ tales que:

$$\psi_{i+1/2}^p \approx \begin{cases} 1 & \text{si } \{u_i^n\} \text{ es 'suave' en } S_p, \\ 0 & \text{otro caso.} \end{cases}$$

Definimos entonces:

$$\mathcal{A} = \{p \in \{1, \dots, P\} \text{ s.t. } \psi_{i+1/2}^p \cong 1\}.$$

La idea sería calcular el flujo numérico como sigue:

$$F_{i+1/2}^A = \begin{cases} F_{i+1/2}^{lo} & \text{si } \mathcal{A} = \emptyset; \\ F_{i+1/2}^{p_s} & \text{donde } p_s = \max(\mathcal{A}) \text{ otro caso;} \end{cases}$$

donde $F_{i+1/2}^{p_s}$ es el flujo numérico CAT2 p_s y $F_{i+1/2}^{lo}$ es un flujo de algún método robusto de primer orden. Sin embargo, no es posible determinar si la solución es suave o no en el

stencil S_1 donde únicamente se dispone de los valores u_i^n, u_{i+1}^n . Por lo tanto, lo que se hará en la práctica es definir:

$$\mathcal{A} = \{p \in \{2, \dots, P\} \text{ s.t. } \psi_{i+1/2}^p \cong 1\}. \quad (0.0.12)$$

y luego:

$$F_{i+1/2}^A = \begin{cases} F_{i+1/2}^* & \text{si } \mathcal{A} = \emptyset; \\ F_{i+1/2}^{p_s} & \text{donde } p_s = \max(\mathcal{A}) \text{ otro caso;} \end{cases} \quad (0.0.13)$$

donde $F_{i+1/2}^*$ es el flujo numérico FL-CAT2 introducido en el capítulo anterior (que también usa el stencil S_2).

Flujo numérico FL-CAT2

La expresión del flujo numérico FL-CAT2 es la siguiente:

$$F_{i+1/2}^* = \psi_{i+1/2}^1 F_{i+1/2}^1 + (1 - \psi_{i+1/2}^1) F_{i+1/2}^{lo}, \quad (0.0.14)$$

donde $F_{i+1/2}^1$ esta dado por

$$F_{i+1/2}^1 = \frac{1}{4}(\tilde{f}_{i,1}^{1,n+1} + \tilde{f}_{i,0}^{1,n+1} + f_{i+1}^n + f_i^n),$$

y

$$\tilde{f}_{i,j}^{1,n+1} = f \left(u_{i+j}^n - \frac{\Delta t}{\Delta x} (f(u_{i+1}^n) - f(u_i^n)) \right), \quad j = \{0, 1\}.$$

$F_{i+1/2}^{lo}$ es un flujo numérico de algún método robusto de primer orden; y $\psi_{i+1/2}^1$ es un limitador de flujo usual, véase [3]:

$$\psi_{i+1/2}^1 = \psi^1(r_{i+1/2}), \quad (0.0.15)$$

donde

$$r_{i+1/2} = \frac{\Delta upw}{\Delta loc} = \begin{cases} r_{i+1/2}^- := \frac{u_i^n - u_{i-1}^n}{u_{i+1}^n - u_i^n} & \text{si } a_{i+1/2} > 0, \\ r_{i+1/2}^+ := \frac{u_{i+2}^n - u_{i+1}^n}{u_{i+1}^n - u_i^n} & \text{si } a_{i+1/2} < 0; \end{cases}$$

y $a_{i+1/2}$ es una estimación de la velocidad de onda.

Indicadores de suavidad

Se introduce una nueva familia de indicadores locales de suavidad $\psi_{i+1/2}^p$, $p \geq 2$, para sistemas de leyes de conservación definida como sigue: dada una aproximación nodal f_i

de una función f en el stencil S_p , $p \geq 2$, centrado en $x_{i+1/2}$, se definen en primer lugar los pesos laterales

$$I_{p,L} := \sum_{j=-p+1}^{-1} (f_{i+1+j} - f_{i+j})^2 + \varepsilon, \quad I_{p,R} := \sum_{j=1}^{p-1} (f_{i+1+j} - f_{i+j})^2 + \varepsilon,$$

donde ε es un número positivo pequeño que se añade para evitar que los pesos se anulen cuando la función es constante. A continuación, se calcula:

$$I_p := \frac{I_{p,L} I_{p,R}}{I_{p,L} + I_{p,R}}. \quad (0.0.16)$$

Finalmente se define el indicador de suavidad para el stencil S_p :

$$\psi_{i+1/2}^p := \left(\frac{I_p}{I_p + \tau_p} \right), \quad (0.0.17)$$

donde

$$\tau_p := (\Delta_{i-p+1}^{2p-1} f)^2.$$

Aquí, $\Delta_{i-p+1}^{2p-1} f$ representa las diferencias divididas de $\{f_{i-p+1}, \dots, f_{i+p}\}$, que pueden ser calculadas de forma recursiva ó usando los coeficientes $\gamma_{p,j}^{2p-1,1/2}$ de la formula de diferenciación numérica $A_{p,i}^{2p-1,1/2}(f)$ como en (0.0.5).

Se demuestra el siguiente resultado:

Proposition 0.0.1 Sean $f_j = f(x_j)$, $j = i - p + 1, \dots, i + p$ los valores de una función f en el stencil S_p , con $p > 2$. Se tiene:

$$\psi_{i+1/2}^p = \begin{cases} 1 - \mathcal{O}(\Delta x^{4(p-1)-2k}) & \text{si } f \in \mathcal{C}^{\max(2p-1, k+2)}, \\ \bar{\mathcal{O}}(\Delta x^{2(k+1)}) & \text{si } f \text{ es } \mathcal{C}^{k+2} \text{ a trozos y } S_p \text{ contiene una discontinuidad de salto aislada} \end{cases}$$

donde $k = 0$ si f no tiene puntos críticos en el stencil o k es el orden del punto crítico cuando hay uno, siendo el orden de un punto crítico el de la primera derivada que no se anula.

En el caso $p = 2$ el indicador puede fallar cuando el stencil S_2 contiene un punto crítico situado exactamente entre dos de los nodos y tal que $f^{(3)}(x^*) = 0$. Aunque es un caso poco probable, se propone una modificación del indicador para solucionar este posible fallo.

ACAT2P methods

La expresión del Métodos Aproximados de Taylor Compactos Adaptativo (ACAT2P) de orden máximo $2P$ para una ley de conservación escalar viene dada por:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} (\mathcal{F}_{i-1/2}^A - \mathcal{F}_{i+1/2}^A). \quad (0.0.18)$$

Los flujos numéricos $\mathcal{F}_{i+1/2}^A$ son definidos por (0.0.12)-(0.0.13) donde $F_{i+1/2}^*$ es el flujo numérico FL-CAT2 (0.0.14) y los indicadores de suavidad son dados por (0.0.15), (0.0.17).

Observe que, por definición, $\mathcal{F}_{i+1/2}^A$ se reduce a:

- un flujo de primer orden si $\psi_{i+1/2}^1 = 0$ y $\psi_{i+1/2}^p = 0$ para todo $p = 2, \dots, P$;
- un flujo de segundo orden si $\psi_{i+1/2}^1 = 1$ y $\psi_{i+1/2}^p \approx 0$ para todo $p = 2, \dots, P$;
- un flujo de orden $2p_s$ si $\psi_{i+1/2}^{p_s} \approx 1$.

Además, si $p_s = P$, entonces ACAT2P coincide con CAT2P que tiene una precisión de $2P$ -orden y es L^2 -estable bajo $CFL \leq 1$.

ACAT para sistemas de leyes de conservación

Para sistemas de leyes de conservación la expresión del método ACAT2P es la misma que en el caso escalar: la única diferencia es el computo de los indicadores de suavidad. En el caso de sistemas, los indicadores de suavidad son calculados primero para cada variable:

$$\psi_{i+1/2}^{j,p}, \quad p = 1, \dots, P,$$

donde

- $\psi_{i+1/2}^{j,1}$ es obtenido al aplicar los indicadores de suavidad (0.0.15) a el j -ésimo componente de las soluciones numéricas $\{u_i^{j,n}\}$.
- $\psi_{i+1/2}^{j,p}$, $p > 2$ es obtenido al aplicar los indicadores de suavidad (0.0.17) a el j -ésimo componente de las soluciones numéricas $\{u_i^{j,n}\}$.
- $\psi_{i+1/2}^{j,2}$ es obtenido al aplicar los indicadores de suavidad (0.0.17) a el j -ésimo componente de las soluciones numéricas $\{u_i^{j,n}\}$.

Una vez calculados estos indicadores de suavidad escalar, definimos

$$\psi_{i+1/2}^p = \min_{j=1, \dots, m} \psi_{i+1/2}^{j,p},$$

para que el stencil seleccionado sea el de longitud máxima entre aquellos en las que todas las variables sean suaves.

Capítulo 5. Métodos Taylor aproximados con reconstrucciones tipo WENO rápidas y optimizadas

El uso de las reconstrucciones espaciales *Weighted Essentially Non-Oscillatory* WENO para calcular la primera derivada en el tiempo en los métodos tipo Taylor (ó Lax- Wendroff en el caso lineal) previene la aparición de oscilaciones cerca de las discontinuidades u ondas de choque (véase [15], [16]). Esta técnica también es usada en los métodos Taylor Aproximados LAT [10] y los métodos Compactos Taylor Aproximados CAT [12], conocidos como WENO-LAT y WENO-CAT respectivamente, como se vió en el Capítulo 2.

El objetivo de este capítulo es explorar la potencialidad de dichos métodos cuando se implementan en las reconstrucciones WENO las siguientes mejoras:

- Los indicadores de suavidad introducidos en [17] que requieren un menor número de operaciones que los propuestos originalmente por Jiang y Shu en [16].
- Los métodos optimizados introducido en [18] y en [19], para evitar la pérdida de precisión cerca de los puntos críticos de las soluciones.

Así, en este sentido, las nuevas reconstrucciones WENO serán óptimas y rápidas: nos referiremos a ellas como reconstrucciones FOWENO.

Reconstrucciones FOWENO

Dados los valores puntuales de la función f en un stencil de $2p + 1$ puntos:

$$S_i = \{f_{i-p}, \dots, f_{i+p}\},$$

donde $f_j = f(x_j)$, los operadores WENO proveen reconstrucciones de f en

$$x_{i+1/2} = x_i + \frac{h}{2},$$

donde h es el paso de la malla (asumiendo que es constante). Esta reconstrucción está basada en los polinomios de Lagrange $p_s(x)$, $0 \leq s \leq p$ que interpolan los valores puntuales en los $p + 1$ sub-stencils

$$S_{p,s} = \{f_{i-p+s}, \dots, f_{i+s}\}, \quad s = 0, \dots, p.$$

Siendo más precisos, la estrategia de WENO consiste en definir las reconstrucciones como una combinación convexa

$$q(x_{i+1/2}) = \sum_{s=0}^p w_s p_s(x_{i+1/2}),$$

donde los pesos w_0, \dots, w_p satisfacen $w_s \cong c_s$ en zonas suaves, siendo c_0, \dots, c_p los pesos para los que se verifica que

$$P(x_{i+1/2}) = \sum_{s=0}^p c_s p_s(x_{i+1/2}),$$

es el polinomio que interpola todos los valores puntuales del stencil total S_i . A estos pesos c_0, \dots, c_p se les denomina pesos ideales. Los pesos w_i han de ser función de algunos indicadores de suavidad. En los métodos FWENO se proponen los siguientes:

$$I_s := \sum_{j=1}^p (f_{-p+i+s} - f_{-p-1+i+s})^2, \quad 0 \leq s \leq p, \quad (0.0.19)$$

Resumamos ahora los métodos FOWENO. La expresión de FOWENO3, (i.e. OWENO3) es la siguiente: dados i y $\varepsilon > 0$,

1. Se incrementa en primer lugar el stencil:

$$\bar{S} = \{f_{i-1}, f_i, f_{i+1}, f_{i+2}\},$$

con $f_i = f(x_i)$.

2. Se calculan los correspondientes polinomios de interpolación evaluados en $x_{i+1/2}$, que, tanto en caso de reconstrucciones a partir de valores puntuales como de promedios en las celdas, vienen dados por

$$p_0(x_{i+1/2}) = -\frac{1}{2}f_{i-1} + \frac{3}{2}f_i, \quad p_1(x_{i+1/2}) = \frac{1}{2}f_i + \frac{1}{2}f_{i+1}.$$

3. Se calculan los indicadores Jiang-Shu correspondientes I_0, I_1 y I_2 (incluyendo el que considera el nodo de más) de la forma

$$I_0 = (f_i - f_{i-1})^2, \quad I_1 = (f_{i+1} - f_i)^2, \quad I_2 = (f_{i+2} - f_{i+1})^2.$$

4. Se calculan los pesos preliminares $\tilde{\omega}_0$ y $\tilde{\omega}_1$:

$$\tilde{\omega}_s := \frac{I_s + \varepsilon}{I_0 + I_1 + 2\varepsilon}, \quad s = 0, 1$$

5. Se define τ como

$$\tau := dI, \quad d := (-f_{i-1} + 3f_i - 3f_{i+1} + f_{i+2})^2, \quad I := I_0 + I_1 + I_2.$$

6. Se calculan los pesos correctores ω :

$$\omega = \frac{J}{J + \tau + \varepsilon}, \quad \text{with } J = I_0(I_1 + I_2) + (I_0 + I_1)I_2.$$

7. Se calculan los pesos corregidos ω_0 y ω_1 :

$$\omega_0 := \omega c_0 + (1 - \omega)\tilde{\omega}_0, \quad \omega_1 := \omega c_1 + (1 - \omega)\tilde{\omega}_1,$$

donde c_0, c_1 son los pesos ideales.

8. Se obtienen las reconstrucciones en $x_{i+1/2}$:

$$q(x_{i+1/2}) = \omega_0 p_0(x_{i+1/2}) + \omega_1 p_1(x_{i+1/2}).$$

A diferencia de FOWENO3, FOWENO($2p + 1$) para $p \geq 2$ no requieren aumentar artificialmente el stencil. Su expresión, combinada con los indicadores de suavidad (0.0.19), puede resumirse como sigue:

Dado i , el stencil S_i y $\varepsilon > 0$.

1. Se calculan los polinomios interpoladores p_j , $j = 0 \leq j \leq p$.
2. Se calculan los indicadores rápidos (0.0.19).
3. Se calcula el discriminante

$$D_p = |B_p - 4A_p C_p|,$$

con

$$A_p = \frac{1}{2} \sum_{j=-p}^p \delta_{p,j}^{2p} f_{i+j}, \quad B_p = \sum_{j=-p}^p \delta_{p,j}^{2p-1} f_{i+j}, \quad C_p = \sum_{j=-p}^p \delta_{p,j}^{2p-2} f_{i+j}.$$

para $j = -p, \dots, p$.

4. Se obtiene el cuadrado de las diferencias divididas de orden $2p$:

$$\tau_p = (2A_p)^2.$$

5. Se calcula

$$d_p := \frac{\tau_p^{a_1} D_p^{a_1}}{\tau_p^{a_1} + D_p^{a_1} + \varepsilon}$$

para algún a_1 a elegir tal que $a_1 \geq 1$, como en [19].

6. Se calcula

$$\alpha_s = c_s \left(1 + \frac{d_p}{I_s^{a_1} + \varepsilon} \right)^{a_2}, \quad 0 \leq s \leq p,$$

donde c_s son los pesos lineales ideales. a_2 ha de satisfacer $a_2 \geq \frac{p+1}{2a_1}$, que es condición suficiente para lograr precisión $(p+1)$ óptima cerca de las discontinuidades [17].

7. Se generan los pesos FOWENO:

$$\omega_s = \frac{\alpha_s}{\alpha_0 + \dots + \alpha_p}, \quad s = 0, \dots, p.$$

8. Se obtiene la reconstrucción en $x_{i+1/2}$:

$$q_p(x_{i+1/2}) = \sum_{s=0}^p \omega_s p_s(x_{i+1/2}).$$

Métodos FOWENO-AT

En los métodos FOWENO-AT se usa la ecuación, $u_t = -f(u)_x$, para aproximar la derivada primera en tiempo: se usan las reconstrucciones FOWENO de los valores puntuales del flujo para aproximar su derivada primera en espacio. Más concretamente, en los métodos LAT se aproxima la derivada primera como sigue:

$$\tilde{u}_{t,i}^{(1)} = -\frac{\hat{f}_{i+1/2} - \hat{f}_{i-1/2}}{\Delta x}.$$

donde $\hat{f}_{i+1/2}$ denota el flujo FOWENO de orden $(2p+1)$ reconstruido en $x_{i+1/2}$. En los métodos CAT la expresión conservativa del flujo se reemplaza por:

$$F_{i+1/2}^p = \hat{f}_{i+1/2} + \sum_{k=2}^m \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x).$$

Las reconstrucciones FOWENO son calculadas en variables conservadas usando el procedimiento descrito en [20], así que su extensión a sistemas es directo.

Capítulo 6. Conclusiones y trabajo futuro

En esta tesis se han introducido nuevas familias de métodos numéricos de alto orden para los sistemas de leyes de conservación basados en técnicas de Taylor Aproximadas que se han descrito en el Capítulo 3. En dicho capítulo:

- Se han revisado los métodos Lax-Wendroff de alto orden para problemas hiperbólicos lineales incluyendo el estudio del orden, la L^2 -estabilidad y el cálculo y las propiedades de los coeficientes.
- A continuación, se ha introducido una extensión a las leyes de conservación no lineales con un orden par arbitrario $2p$ de precisión, los llamados métodos Compact Approximate Taylor (CAT). A diferencia de aplicaciones anteriores de los métodos de Taylor a las leyes de conservación, estos métodos usan stenciles centrados de $(2p + 1)$ -puntos al igual que los métodos de Lax-Wendroff para problemas lineales. Además son linealmente L^2 -estable bajo una condición CFL-1.

Además, se han aplicado dos formas originales y apropiadas de mantener bajo control las oscilaciones espúreas generadas por los métodos CAT en las proximidades de las discontinuidades u ondas de choques. En primer lugar, se ha introducido el método Adaptive Compact Taylor ACAT en el capítulo 4. En dicho capítulo:

- Se ha presentado una versión adaptativa de los métodos CAT que incorpora la técnica de limitación de flujo (FL-CAT2 ó ACAT2) para el esquema de orden bajo.
- Se ha introducido y analizado una nueva familia de indicadores de suavidad de alto orden que son capaces de detectar la suavidad de la solución numérica en stenciles centrados.
- Se ha presentado la ampliación a los problemas 2D de los métodos CAT y ACAT.

Y finalmente, la combinación de métodos LAT y CAT con reconstrucciones WENO rápidas y óptimas ha sido estudiada en el Capítulo 5. En dicho capítulo:

- Se han considerado dos operadores diferentes de reconstrucción espacial de alto orden: los operadores estándar WENO y FOWENO. Este último combina el uso de indicadores de suavidad rápidos (que coinciden con los indicadores de suavidad originales en el caso de tercer orden) y el cálculo de pesos óptimos que permiten preservar la precisión de las reconstrucciones cercanas al punto crítico independientemente de su orden. En nuestro conocimiento, esta es la primera vez que estas dos técnicas se han combinado.

Trabajo futuro:

- Implementación optimizada de métodos CAT en arquitecturas GPU.
- Combinación de métodos CAT con la estrategia *MOOD* (véase [21], [22], [23],...) para remediar las oscilaciones espúreas.

- Extensión a los sistemas de leyes de equilibrio, incluyendo el análisis de la propiedad *well-balanced*.
- Extensión a sistemas hiperbólicos no conservativos. En particular, los métodos de CAT se pueden combinar con los métodos *Well-controlled dissipation* WCD .(véase [24]).

Abstract

In this thesis a new family of high-order methods for systems of conservation laws is introduced: the Compact Approximate Taylor (CAT) methods. As in the Approximate Taylor methods proposed by Zorío, Baeza, and Mulet in [10] the Cauchy-Kovalevsky procedure is circumvented by using Taylor approximations in time that are computed in a recursive way. The difference is that here this strategy is applied *locally* to compute the numerical fluxes what leads to methods that have $(2p + 1)$ -point stencil and order of accuracy $2p$, where p is an arbitrary integer. Moreover we prove that they reduce to the high-order Lax-Wendroff methods for linear problems and hence they are linearly L^2 -stable under the usual CFL condition. Although CAT methods present an extra computational cost due to the local character, this extra cost is compensated by the fact that they still give good solutions with CFL values close to 1.

In order to prevent the spurious oscillations that appear close to discontinuities two shock-capturing methods have been considered: first a new family of high-order numerical methods, the Adaptive Compact Approximate Taylor Methods, based on the use of a local adaptation of the order of the scheme according to the smoothness of the numerical solution that is measured using a new family of smoothness indicators. Next, the Approximate Taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions, which is an original variant of WENO combined with the high-order Approximate Taylor Methods. Both methods are compared with standard WENO methods combined with TVDRK methods or the Approximate Taylor methods proposed in [10] in a number of test cases ranging from linear scalar conservation laws to the 2D Euler system of gas dynamics.

Chapter 1

Introduction

1.1 Motivation

Lax-Wendroff methods for linear systems of conservation laws are based on Taylor expansions in time in which the time derivatives are transformed into spatial derivatives using the equations [1], [2], [3], [4]. The spatial derivatives are then discretized by means of centered high-order differentiation formulas. This procedure allows to derive numerical methods of order $2p$, where p is an arbitrary integer, using a centered $(2p+1)$ -point stencil that are L^2 -stable under the usual CFL condition.

This thesis focuses on the extension of Lax-Wendroff methods to nonlinear systems of conservation laws. Many authors have developed numerical methods that use this approach for the time discretization as an alternative to multistep or multistage one-step methods like the SSP Runge-Kutta schemes (see [25]): this is the case of the original finite volume ENO schemes (see [26]). This approach was also followed by E.F. Toro and collaborators in the design of the so-called ADER (arbitrary high-order schemes utilizing higher order derivatives) methods: see [5], [6], [7]). . . The computation of time derivatives in these methods is based on the modified generalized Riemann problem introduced by Toro in [27]. A Lax-Wendroff, second order evolution, Galerkin method for multidimensional hyperbolic systems was also introduced in [28]. More recently, in [11] this procedure has been used together with WENO reconstructions for the spatial discretization. The main benefit, compared to RK time discretizations, is that only one WENO reconstruction is needed at each spatial cell per time step.

The main difficulty to extend Lax-Wendroff methods to nonlinear problems comes from the transformation of time derivatives into spatial derivatives. A first strategy to do this is given by the Cauchy-Kovalevskaya (CK) procedure, in which the PDE is used to replace time derivatives by spatial derivatives. The main drawback of this procedure comes from the fact that it leads to expressions whose number of terms grows exponentially, implying high computational costs and difficult implementations. In the context of ADER methods, this difficulty has been circumvented in ADER-WENO methods (see [8]) by replacing the

CK procedure by local space-time problems that are solved with a Galerkin method. The so-called $P_N P_M$ methods introduced in [9], that generalize ADER-WENO and DG methods, also follow this approach. These methods can be applied both on structured and unstructured meshes with $CFL = 1$ condition for stability.

An alternative to both CK and local space-time problems has been proposed recently in [10] based on an Approximate Taylor (AT) method: the time derivatives are approximated using high-order centered differentiation formulas combined with Taylor approximations in time that are computed in a recursive way. Nevertheless AT schemes are not proper generalizations of Lax-Wendroff methods: they have $(4p + 1)$ -point stencils and worse linear stability properties than the original Lax-Wendroff methods. Nevertheless, they can be stabilized by using one WENO reconstruction per spatial cell and time step, as shown in [11], and the resulting methods typically give good results under a $CFL = 0.5$ condition. These methods are easy to implement in Cartesian uniform meshes and perform well.

1.2 Scope of this thesis

This thesis has two main objectives:

- To develop a family of high order numerical methods for nonlinear systems of conservation laws based on an approximate Taylor (AT) procedure that constitute a proper generalization of Lax-Wendroff methods, i.e. that reduce to the standard high-order Lax-Wendroff methods when the flux is linear.
- To combine this new AT procedure with some well known shock capturing techniques and/or obtain a new, appropriate new one to cure the spurious oscillations generated close to discontinuities by the AT methods.

1.3 Outline

These objectives have been satisfactorily satisfied in three papers, the first one, titled *Compact Approximate Taylor Methods*, was published in 2019 by the *Journal of Scientific Computing*, see [12]. This article introduces a variant of the AT procedures that constitutes a proper high-order extension of Lax-Wendroff methods to non-linear systems of conservation laws. Two shock-capturing techniques were tested in this first paper: a combination of the second-order method of the family with a robust first-order methods based on the use of a flux limiter was first tested. Then, following [10], Weighted Essentially Non-Oscillatory (WENO) spatial reconstructions (see [15], [16]) were used to compute the first-order time derivatives.

Once the new methods were derived and tested, the second objective was faced: to find effective and appropriate ways of avoiding spurious oscillations close to discontinuities or shocks. In collaboration with G. Russo, E. Macca (University of Catania, Italy) and D. Zorío (University of Concepción, Chile) we introduced a new family of high-order numerical methods which based on the use of a local adaptation of the order of the scheme according to the smoothness of the numerical solution according to a new family of smoothness indicators. The resulting paper titled *Adaptive Compact Approximate Taylor Methods*, which is available in the *arXiv* repository and is expected to be published soon, see [13].

The last paper, titled *Approximate Taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions* (see [14]) was developed in collaboration with the D. Zorío (University of Concepcion, Chile). In this work an original variant of WENO combined with the high-order AT methods was introduced. This paper is also available in the *arXiv* repository and was submitted in February 2020 to *Journal of Scientific Computing*.

The content of this thesis consists mainly of the three publications mentioned above and the organization is as follows:

Chapter 2 contains the preliminary concepts and notation that we consider important and/or necessary to understand the subsequent chapters. In addition to the basic concepts related to hyperbolic systems of conservation laws, we include an introduction to WENO reconstructions that will play an important role in Chapters 2, 3, and 4: WENO reconstructions will be used in the design of some of the numerical methods and comparison with WENO methods will be used to test the new numerical schemes. WENO reconstructions are frequently combined with a time discretization based on the classical TVD Runge-Kutta (TVDRK) methods that will be also recalled. Approximate Taylor methods as an alternative to TVDRK methods for the time discretization have been used in several well-known methods, as it was mentioned above, but here we will focus on two of them that are at the basis of our work: the original second-order Lax-Wendroff method [1] and its extension to high-order methods for non-linear conservation law systems introduced by Qiu and Shu [11].

Chapter 3 takes over essentially the contents of the article *Compact Approximate Taylor Methods for systems of conservation laws*.

Chapter 4 is devoted to the paper *An order-adaptive compact approximation Taylor method for systems of conservation laws* or CAT.

Chapter 5 focuses on the article *Lax Wendroff approximate Taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions*.

Finally, conclusions and future work are presented in Chapter 6.

Chapter 2

Preliminaries

In this chapter, we review basic facts about hyperbolic system of conservation laws and some high-order finite differences numerical methods used for solve them. In particular, we focus on the high-order methods that give relevant background for the following chapters.

2.1 Hyperbolic conservation laws

Conservation laws are systems of first order partial differential equations that can be written as:

$$\frac{\partial u}{\partial t} + \sum_{i=1}^d \frac{\partial f^i(u)}{\partial x_i} = 0, \quad x \in \mathbb{R}^d, \quad t \in \mathbb{R}^+, \quad (2.1.1)$$

where $u = (u_1, \dots, u_m)^T : \mathbb{R}^d \times \mathbb{R}^+ \rightarrow \mathbb{R}^m$ is the vector of conserved variables and $f^i : \mathbb{R}^m \rightarrow \mathbb{R}^m$ are the flux functions, $i = 1, \dots, d$.

Equation (2.1.1) is provided with initial conditions

$$u(x, 0) = u_0(x), \quad x \in \mathbb{R}^d.$$

Solving this Cauchy problem allows one to find the state of the system at a certain time $t = T$, given the state at time $t = 0$. System (2.1.1) is hyperbolic if any linear combination of the Jacobian matrices of f^i , $\sum_{i=1}^d \alpha_i (f^i)'(v)$, is diagonalizable with real eigenvalues for each $v \in \mathbb{R}^m$. This conditions ensures the stability of Cauchy problems for systems linearized about constant states. Boundary conditions have to be specified when considering a bounded domain $\Omega \subseteq \mathbb{R}^d$. At the end of this chapter, we include some comments about to numerical treatment of boundary conditions for high-order methods.

System (2.1.1) can be written in quasi-linear form as:

$$\frac{\partial u}{\partial t} + \sum_{i=1}^d (f^i)'(u) \frac{\partial u}{\partial x_i} = \frac{\partial u}{\partial t} + \sum_{i=1}^d \sum_{j=1}^m \frac{\partial f^i(u)}{\partial u_j} \frac{\partial u_j}{\partial x_i} = 0.$$

The particular case $m = 1$, referred as scalar conservation law, will be used often in this work for the design and validation of numerical methods due to its simplicity. In 1D, i.e. if $d = 1$, this conservation law can be written as

$$u_t + f(u)_x = 0, \quad x \in \mathbb{R}, \quad t \in \mathbb{R}^+, \quad (2.1.2)$$

with the conserved variable $u : \mathbb{R} \times \mathbb{R}^+ \rightarrow \mathbb{R}$ and flux function $f : \mathbb{R} \rightarrow \mathbb{R}$.

Conservation laws regularly come from an integral relationship representing the conservation of a certain quantity u . Conservation means that the amount of quantity contained in a given volume can only change due to the flux of this quantity crossing the interfaces of the given volume. In one space dimension this is written as:

$$\int_{x_1}^{x_2} (u(x, t_2) - u(x, t_1)) dx = \int_{t_1}^{t_2} f(u(x_1, t)) dt - \int_{t_1}^{t_2} f(u(x_2, t)) dt, \quad (2.1.3)$$

where the control volume in the $x - t$ plane is $V = [x_1, x_2] \times [t_1, t_2] \subseteq \mathbb{R} \times \mathbb{R}$.

The characteristic structure of the hyperbolic conservation laws refers to the eigenstructure of the Jacobian matrix of the fluxes. The characteristic structure is important for exact and approximate solutions of the equations. The characteristic speeds are the eigenvalues of the Jacobian matrices. For one-dimensional systems of conservation laws, we will assume that there are smooth functions $\lambda_k : \mathbb{R}^m \rightarrow \mathbb{R}$, $k = 1, \dots, m$, such that $\lambda_k(u)$ is the k -th eigenvalue of $f'(u)$. For scalar conservation laws, these characteristic speeds are just the flux derivatives $f'(u)$. For one-dimensional systems of conservation laws, the characteristics for a solution u are curves $(t, x(t))$ satisfying $x'(t) = \lambda_k(u(x(t), t))$. For scalar equations, this reduces to $x'(t) = f'(u(x(t), t))$. In this case, it can be easily shown that the solution u is constant along these curves so that they are straight lines of slope $f'(u_0(x(0)))$.

A classical solution of (2.1.2) is a smooth function $u : \mathbb{R} \times \mathbb{R}^+ \rightarrow \mathbb{R}$ that satisfies the equations pointwise. As pointed out in the previous section, an essential feature of this problem is that, given an initial condition, in general there are no classical solutions of (2.1.2) beyond some finite time, even if the initial condition u_0 is a smooth function.

In order to be able to consider non-smooth solutions, the classical concept of solution can be relaxed by using the integral form of the equation which is more general than the differential form: the derivation of the latter form is based on some smoothness assumptions that do not hold in general. The use of the integral form allows one to obtain a weak formulation that involves fewer derivatives on u , and hence, requiring less smoothness.

Definition 1 *A function $u(x, t)$ is a weak solution of (2.1.1) with given initial data $u(x, 0)$ if*

$$\int_{\mathbb{R}^+} \int_{\mathbb{R}^d} \left[u(x, t) \frac{\partial \phi}{\partial t}(x, t) + \sum_{j=1}^d f^j(u) \frac{\partial \phi}{\partial x_j} \right] dx dt = - \int_{\mathbb{R}^d} \phi(x, 0) u(x, 0) dx \quad (2.1.4)$$

is satisfied for all $\phi \in C_0^1(\mathbb{R}^d \times \mathbb{R}^+)$, where $C_0^1(\mathbb{R}^d \times \mathbb{R}^+)$ is the space of continuously differentiable functions with compact support in $\mathbb{R}^d \times \mathbb{R}^+$.

Weak solutions provide an adequate generalization of the concept of classical solution for hyperbolic conservation laws. It is easy to see that strong solutions are also weak solutions, and continuously differentiable weak solutions are strong solutions.

The Rankine-Hugoniot condition [29], [30], whose derivation can be found for example in [31], [32], [33], follows from the definition of weak solution. This condition characterizes the movement of the discontinuities of the admissible weak solutions and gives information about the behavior of the conserved variables across discontinuities.

For a general conservation law the Rankine-Hugoniot condition reads:

$$[f] \cdot n = [u](n \cdot s), \quad (2.1.5)$$

where $f = (f^1, \dots, f^d)$ is a matrix containing the fluxes, u is the solution, s is the speed of propagation of the discontinuity and n is the vector normal to the discontinuity. The notation $[\cdot]$ indicates the jump on a variable across the discontinuity. For scalar problems (2.1.5) reduces to:

$$f(u_L) - f(u_R) = s(u_L - u_R)$$

where u_L and u_R are the states at the left and the right side of the discontinuity respectively. It can be shown that a function piecewise smooth function $u(x, t)$ is a weak solution of (2.1.1) if and only if (2.1.1) is satisfied in regions where u is smooth and the Rankine-Hugoniot condition is satisfied at the discontinuities of u : see [31].

However, weak solutions are often not unique (see [34]), and there are *entropy conditions* proposed to single out a unique weak solution known as entropy solution whose discontinuities behave according to the underlying physics of the system. For instance, according to Lax's E-condition [35], a discontinuity is admissible if there exists p , $1 \leq p \leq m$, such that:

$$\lambda_p(u_L(t)) \geq s'(t) \geq \lambda_p(u_R(t)). \quad (2.1.6)$$

where λ_p is the p -th eigenvalue of the flux Jacobian; $x = s(t)$ is the location of the discontinuity at time t ; and $u_L(t)$, $u_R(t)$, the left and right limits of the solution respectively.

There is also an entropy inequality based on entropy-entropy flux pairs, due to Lax [35] as well, which is closely related to vanishing viscosity solutions. There are other entropy criterion such as Oleinik's condition [36], Kruřkov's condition [37], Wendroff's condition [38] or Liu's condition [39]. To prove the existence of weak solutions satisfying a given entropy conditions is in general a great challenge. Positive answers to this existence question can be found for wide classes of multi-dimensional scalar conservation laws or one-dimensional systems. For scalar conservation laws and some hyperbolic systems of conservation laws, existence can be established by the front tracking method [40], [41], [42].

Other means of establishing existence may apply in some cases, such as Lax-Oleinik's formula [43] for scalar conservation laws with convex flux.

Cauchy problems whose initial conditions are piecewise constant with only one discontinuity are of special importance. The solution of these problems, called Riemann problems, play an important role in the design of numerical methods. Knowledge of the characteristic structure, Riemann invariants, and solution of the Rankine-Hugoniot conditions are necessary to find the solutions of Riemann problems.

2.2 Numerical methods

Next, we review some notions and results related to numerical methods for hyperbolic systems of conservation laws.

Many high-order finite difference schemes are based on the method of lines in which the system is first discretized in space using a high-order reconstruction operator and then a high-order method is applied to the resulting ODE system. Although the design of the numerical methods introduced in this work don't follow this approach, their results will be compared with standard numerical methods whose design do, such as the high-order finite difference methods of Shu-Osher based on WENO reconstructions and on TVDRK time solvers. [25], [44], [45], [46]. For this reason, we will briefly explain these methods. The original second-order Lax-Wendroff method Lax-Wendroff will be also recalled, followed by its extension to higher order proposed by Qiu and Shu [11] that can be considered as the first step in the direction of the methods designed in this thesis.

2.2.1 Computational grids

The first step to solve numerically a PDE system is to replace the continuous problem, represented by the PDE's, by a discrete approximation of it. If the system has only one space coordinate x , first the $x - t$ plane is discretized by choosing a mesh (or grid) composed by a finite set of points or volumes defined below. Then the PDE is discretized on this grid, and the resulting discrete, finite-dimensional problem, is solved. In the case of finite difference methods a point-value discretization is used: the unknowns of the discretized problem are approximation of the value of the PDE solution at the points of the mesh, while in the case of finite volume methods they are approximations of the averages of the PDE solution at the cells or volumes that compose the mesh.

Consider a scalar Cauchy problem in one space dimension,

$$\begin{cases} u_t + f(u)_x = 0, & x \in \mathbb{R}, \quad t \in \mathbb{R}^+, \\ u(x, 0) = u_0(x), \end{cases} \quad (2.2.1)$$

where $u, f : \mathbb{R} \rightarrow \mathbb{R}$.

To define the mesh, a discrete subset of points (nodes) $\{x_j\}_{j \in \mathbb{Z}}$, $x_j \in \mathbb{R} \quad \forall j$ will be considered. Uniform meshes will be considered here, i.e., $x_j - x_{j-1} = \Delta x > 0$, for each

$j \in \mathbb{Z}$. This constant is called mesh size and it will be also represented by $h = \Delta x$. From the points $\{x_j\}$ we define the cells c_j as the subintervals whose respective centers are x_j :

$$c_j = \left[\frac{x_{j-1} + x_j}{2}, \frac{x_j + x_{j+1}}{2} \right] = [x_{j-1/2}, x_{j+1/2}].$$

Depending on the context, one may understand the mesh or grid as the set of cells $\{c_j\}_{j \in \mathbb{Z}}$ or the set of nodes $\{x_j\}_{j \in \mathbb{Z}}$.

The time variable t is discretized by a set of points $\{t^n\}_{n \in \mathbb{N}}$, with $t^n < t^{n+1}$, $\forall n \in \mathbb{N}$. If $t^{n+1} - t^n$ is constant with respect to n , we denote it by Δt and call it the time increment. In the methods considered here Δt is not constant: it will be computed on the basis of a stability criterion. We will denote by $u^n = \{u_j^n\}_{j \in \mathbb{Z}}$ the approximation of the exact solution $u(x_j, t^n)$ of (2.2.1).

In real problems, the domain of definition of the equations is restricted to a bounded subset of \mathbb{R} and a finite time interval, so the grid has to be restricted to a finite number of nodes or cells.

If we consider the interval $I = [0, 1]$ and a fixed time $T > 0$, then, we can take positive numbers M and N and define a set of nodes $\{x_j\}_{0 \leq j < M}$ given by $x_j = (j + 1/2)\Delta x$, with $\Delta x = \frac{1}{M}$. The points in time $\{t^n\}_{0 \leq n < N}$ can be defined by $t^n = n\Delta t$, with $\Delta t = \frac{1}{N}$.

We can extend all of these concepts to two-dimensional problems. Let us consider a scalar conservation law in $2D$ with the form:

$$\begin{cases} u_t(x, y, t) + f(u(x, y, t))_x + g(u(x, y, t))_y = 0, & (x, y) \in \mathbb{R} \times \mathbb{R}, \quad t \in \mathbb{R}^+, \\ u(x, y, 0) = u_0(x), \end{cases}$$

and two sets of ordered points, $\{x_i\}_{i \in \mathbb{Z}}$ and $\{y_j\}_{j \in \mathbb{Z}}$, satisfying $x_i < x_{i+1}$ for all $i \in \mathbb{Z}$ and $y_j < y_{j+1}$ for all $j \in \mathbb{Z}$. Moreover, we assume as before that $\Delta x = x_{i+1} - x_i$ and $\Delta y = y_{j+1} - y_j$ are constant with respect to i and j respectively. We can define cells $c_{i,j}$ by

$$c_{i,j} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}],$$

so that each node (x_i, y_j) is the center of the cell $c_{i,j}$.

2.2.2 Conservative methods

The simplest way to approximate derivatives is by means of finite-differences schemes. If the solution to approximate is discontinuous, in principle, finite-differences schemes may not give a satisfactory approximation of the partial derivatives appearing in the equations. Finite-volume methods and Discontinuous Galerkin methods overcome this difficulty by resorting to weak formulations like (2.1.3) or (2.1.4) that do not require derivatives of the unknowns.

On the other hand, there may be more than one weak solution and the method may not converge to the right one or it may converge to a function that is not a weak solution

of the PDE. Some examples of these facts can be found, e.g. in [34]. There exists a simple requirement that we can impose on the numerical methods to guarantee that they do not converge to non-solutions. Conservative methods ensure that convergence can only be achieved to weak solutions (Lax-Wendroff's theorem).

Definition 2 *A numerical method is said to be conservative if it can be written in the form*

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x} \left(\hat{f}(u_{j-p+1}^n, \dots, u_{j+q}^n) - \hat{f}(u_{j-p}^n, \dots, u_{j+q-1}^n) \right), \quad (2.2.2)$$

where the function $\hat{f} : \mathbb{R}^{p+q} \rightarrow \mathbb{R}$ is called the numerical flux function and $p, q \in \mathbb{N}$, $p, q \geq 0$.

The purpose of conservative methods is to reproduce at a discrete level the conservation of the physical variables in the continuous equations. In fact (2.2.2) can be seen as a discrete version of the integral form (2.1.3) of the PDE.

An essential requirement on the numerical flux is the consistency condition:

Definition 3 *We say that the numerical flux function of a conservative numerical method is consistent with the conservation law if the numerical flux function \hat{f} reduces to the exact flux f for the case of constant flow, i.e.,*

$$\hat{f}(u, \dots, u) = f(u).$$

The consistency condition is necessary to ensure that a discrete form of conservation, analogous to the conservation law, is provided by conservative methods.

In general, some smoothness is required in the way in which \hat{f} approaches a certain value $f(u)$. Hence we suppose that the flux function is locally Lipschitz continuous in each variable.

Lax-Wendroff's theorem states that, if a conservative method produces a sequence of approximations that converges to a function $u(x, t)$ as the grid is refined, then u is necessarily a weak solution of the conservation law:

Theorem 2.2.1 (Lax-Wendroff, [1],[42]) *Consider a sequence of grids indexed by $k = 1, 2, \dots$ with grid sizes $(\Delta x_k, \Delta t_k)$, satisfying*

$$\begin{aligned} \lim_{k \rightarrow +\infty} \Delta x_k &= 0, \\ \lim_{k \rightarrow +\infty} \Delta t_k &= 0. \end{aligned}$$

Let $\{u_k(x, t)\}$ denote the piecewise constant function defined from the numerical solution obtained by a conservative numerical method consistent with (2.1.1) on the k -th grid. If the total variation of the function $u_k(\cdot, t)$ is uniformly bounded in k, t , i.e., $\sup_{k, t \in [0, T]} TV(u_k(\cdot, t)) < \infty$ and $u_k(x, t)$ converges in \mathcal{L}_{loc}^1 to a function $u(x, t)$ as $k \rightarrow \infty$, then u is a weak solution of the conservation law.

Nevertheless, a sequence of numerical approximations produced by a conservative method may converge to a weak solution that is not an entropy solution and thus that it is not admissible. Some extra conditions for convergence to entropy solutions have to be imposed: see [47], [48].

2.2.3 High-resolution conservative methods

The term “high-resolution” is applied to methods whose local truncation error has order higher than two, thus giving second or even higher order global errors in smooth parts of the solution, while giving well-resolved non-oscillatory approximations near discontinuities.

Although, there are several high-order resolution techniques for conservative methods, see [5], [6], [7]... , we will focus on two approaches;

- The method of lines, which refers to numerical methods for evolutionary PDEs in which the spatial derivatives are first discretized leading to a system of ordinary differential equations that is then solved by applying a numerical method for ODEs. The high-order discretization of the spatial derivatives will be discretized here by using the well-known essentially non-oscillatory (WENO) reconstructions in conservative form: [16], [49] in a conservative form. And, for discretization in time, we use the high-order methods TVD Runge- Kutta.
- Lax-Wendroff (LW) methods in the time and space discretization are performed at the same time and are based on Taylor expansions.

2.2.3.1 ENO and WENO reconstructions

The Essentially Non-Oscillatory (ENO) and the Weighted Essentially Non-Oscillatory (WENO) reconstruction operators are based on special polynomial interpolation techniques that, given the set of the values of a function at the center of the cells or its cell averages, provide approximations of the point values of the function at the intercells (one to the left and one to the right) in such a way that, if the function is smooth these approximations are high-order accurate but if the function has discontinuities, Gibbs phenomena is avoided. To compute the reconstruction at an intercell only the values at some neighbor cells called the *stencil* are used. In the case of the ENO reconstruction [26], the stencil choice is based on the ‘smoothness’ of the cell values, that is measured using undivided differences: the stencil containing the smoothest data is selected. Although ENO reconstruction of order r uses stencils of r cells, during the selection procedure r possible stencils are considered that contain in total $2r - 1$ cells.

Weighted Essentially Non-Oscillatory (WENO) reconstructions, introduced by Liu, Osher and Chan in [49], are based on the idea of increasing the order of accuracy of the method in smooth regions by considering a convex combination of the different

interpolating polynomials at the candidate stencils of the ENO method. Spatially varying weights are chosen in order to increase the accuracy of the individual reconstructions corresponding to the different stencils. Using this technique, the authors of [49] raised the order of accuracy of the ENO method using r -point stencils from r to $r + 1$ in smooth regions, while retaining the r th-order near discontinuities. The weight assigned to the interpolating polynomial associated to a given stencil depends on a smoothness indicator, for which a suitably weighted sum of squares of (undivided) differences of the data corresponding to that stencil was used. A new smoothness indicator was proposed by Jiang and Shu in [16] to achieve fifth-order reconstructions from third-order ENO reconstructions, i.e. an order of $2r - 1$ when $r = 3$.

2.2.3.2 ENO and WENO algorithms

We will focus here on reconstruction operators that, given the set $\{f(x_i)\}$ of the values at the center of the cells, provide approximations of the point values of the function at the intercells $\{f(x_{i+1/2})\}$. Let $h = \Delta x$ be the grid size. In the ENO algorithm [26] a left-biased approximation to the value $f(x_{j+1/2})$ is computed using the values $f_l = f(x_l)$ at stencils of r nodes ($r \geq 2$) that contain the node x_j . There are r stencils of r nodes that contain x_j , given by

$$S_{j+k}^r = \{x_{j+k-r+1}, \dots, x_{j+k}\}, \quad k = 0, \dots, r-1.$$

From them, r different polynomial reconstructions of degree at most $r - 1$, denoted by $p_k^r(x)$, can be constructed, each of them satisfying

$$p_k^r(x_{j+1/2}) = f(x_{j+1/2}) + \mathcal{O}(h^r),$$

if f is smooth in the corresponding stencil.

Among all the candidate substencils, the ENO algorithm selects the substencil producing the smallest divided differences, in an attempt to produce less oscillatory interpolants, see [26, 50] for further details.

Weighted ENO reconstructions appeared in [49] as an improvement upon ENO reconstructions. In [49], Liu et al. stated that there is no need of selecting just one of the possible stencils, and that a combination of them can give better results in smooth regions. If f is smooth in all stencils, a $(2r - 1)$ -th order reconstruction

$$p_{r-1}^{2r-1}(x_{j+1/2}) = f(x_{j+1/2}) + \mathcal{O}(h^{2r-1}),$$

can be computed using the stencil $S_{j+r-1}^{2r-1} = \{x_{j-r+1}, \dots, x_{j+r-1}\}$, instead of the r -th order reconstruction provided by the ENO algorithm.

If we consider the r candidate stencils of the ENO algorithm,

$$S_{j+k}^r = \{x_{j-r+1+k}, \dots, x_{j+k}\}$$

for $k = 0, \dots, r-1$, and the $(r-1)$ -th degree polynomial reconstructions $p_k^r(x)$, defined on each stencil S_{j+k}^r , satisfying $p_k^r(x_{j+1/2}) = f(x_{j+1/2}) + \mathcal{O}(h^r)$, then a (left-biased) WENO reconstruction of f is given by the convex combination:

$$q(x_{j+1/2}) = \sum_{k=0}^{r-1} w_{j,k} p_k^r(x_{j+1/2}), \quad (2.2.3)$$

where:

$$w_{j,k} \geq 0, \quad k = 0, \dots, r-1, \quad \sum_{k=0}^{r-1} w_{j,k} = 1$$

and the corresponding (left-biased) reconstruction evaluation operator is given by:

$$\mathcal{R}(f_{j-r+1}, \dots, f_{j+r-1}) = \sum_{k=0}^{r-1} w_{j,k} p_{j,k}^r(x_{j+1/2}).$$

The weights should be selected with the goal of achieving the maximal order of accuracy $2r-1$ wherever f is smooth, and r -th order, as the ENO algorithm, elsewhere. To do this, in [49], it was pointed out that, for $r \geq 2$, coefficients C_k^r , called optimal weights, can be computed such that:

$$p_{r-1}^{2r-1}(x_{j+1/2}) = \sum_{k=0}^{r-1} C_k^r p_k^r(x_{j+1/2}),$$

where,

$$C_k^r \geq 0 \quad \forall k, \quad \sum_{k=0}^{r-1} C_k^r = 1.$$

A closed explicit formula for the optimal weights have been given in [51]. The optimal weights for $r = 2, 3, 4, 5$ are displayed in Table 2.1.

r	$k = 0$	$k = 1$	$k = 2$	$k = 3$	$k = 4$
2	1/3	2/3			
3	1/10	6/10	3/10		
4	1/35	12/35	18/35	4/35	
5	1/126	20/126	60/126	40/126	5/126

Table 2.1: Optimal weights for $r = 2, 3, 4, 5$.

Notice that to accomplish the requirements on the non-linear weights w_k one can define them satisfying the condition:

$$w_{j,k} = C_k^r + \mathcal{O}(h^m), \quad k = 0, \dots, r-1, \quad (2.2.4)$$

with $m \leq r - 1$. Then, there holds, (see [49],[51]) that

$$f(x_{j+\frac{1}{2}}) - q(x_{j+\frac{1}{2}}) = \mathcal{O}(h^{r+m}), \quad (2.2.5)$$

and, if $m = r - 1$ in (2.2.4), then the approximation (2.2.5) has maximal order $2r - 1$.

Another requirement for the weights is that the ones corresponding to polynomials constructed using stencils where the function has a singularity should be very small, so that the WENO reconstruction does not take those polynomials into account and, as required, yields an approximation of an order not worse than that of the ENO interpolators. Also, the weights should be smooth functions of the cell-averages of the reconstructed function and efficiently computable.

Weights satisfying these conditions are defined in [49] as follows:

$$w_{j,k} = \frac{\alpha_k}{\sum_{i=0}^{r-1} \alpha_i}, \quad \alpha_k = \frac{C_k^r}{(\varepsilon + I_k)^p}, \quad k = 0, \dots, r - 1, \quad (2.2.6)$$

where $p \in \mathbb{N}$, C_k^r are the optimal weights, $I_k = I_k(h)$ is a smoothness indicator of the function f on the stencil S_k and ε is a small positive number, possibly dependent on h , introduced to avoid null denominators.

Jiang and Shu's smoothness indicators (see [16]) are defined as follows:

$$I_k = \sum_{l=1}^{r-1} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} (p_k^{(l)}(x))^2 dx. \quad (2.2.7)$$

They allow to obtain WENO schemes with optimal order $2r - 1$ for $r = 2, 3$. The term h^{2l-1} was introduced to remove h -dependent factors in the derivatives of the polynomial reconstructions $p_k(x)$.

In [51], Aràndiga et al. showed that the choice of ε crucial for the achievement of optimal order: they showed that if ε is proportional to the square of h , then the order of WENO reconstruction is $2r - 1$ at smooth regions regardless of neighboring extrema, and the order is r when the function has a discontinuity in the stencil but it is smooth in at least one of the sub-stencils.

2.2.3.3 WENO conservative methods

WENO conservative methods for (2.1.2) have the form

$$\frac{du_i(t)}{dt} + \frac{1}{\Delta x} (\hat{f}_{i+1/2} - \hat{f}_{i-1/2}) = 0. \quad (2.2.8)$$

where $\hat{f}_{i+1/2}$ denotes the WENO reconstruction of the numerical fluxes $\{f(u_i(t))\}$ at the intercell $x_{i+1/2}$.

For instance, for the third-order WENO scheme, the numerical flux $\hat{f}_{i+1/2}$ is defined as follows:

- The smoothness indicators (2.2.7) are

$$I_1 = (f_i - f_{i-1})^2, \quad I_2 = (f_{i+1} - f_i)^2. \quad (2.2.9)$$

- The optimal weights are

$$C_1^2 = \frac{1}{3}, \quad C_2^2 = \frac{2}{3}$$

- Finally

$$\hat{f}_{i+1/2} = w_{i,1} \hat{f}_{i+1/2}^{(1)} + w_{i,2} \hat{f}_{i+1/2}^{(2)}, \quad (2.2.10)$$

where

$$\hat{f}_{i+1/2}^{(1)} = -\frac{1}{2}f_{i-1} + \frac{3}{2}f_i, \quad \hat{f}_{i+1/2}^{(2)} = \frac{1}{2}f_i + \frac{1}{2}f_{i+1}. \quad (2.2.11)$$

and the weights are given by (2.2.6).

Notice that, in order to compute $\hat{f}_{i+1/2}$ only the values f_{i-1}, f_i, f_{i+1} have been used so that the reconstruction is left-biased. Due to this, (2.2.8) is only stable if the eigenvalues of the Jacobian of the flux function $\frac{\partial f(u)}{\partial u}$ are positive. If all of them are negative, the right-biased WENO reconstruction should be applied. For the general case in which eigenvalues have different signs and may change their sign, a splitting of the flux function

$$f(u) = f^+(u) + f^-(u). \quad (2.2.12)$$

is usually considered, such that all the eigenvalues of $\frac{\partial f^+(u)}{\partial u}$ are positive and all the ones of $\frac{\partial f^-(u)}{\partial u}$ are negative. Then, the left-biased reconstruction is applied to f^+ and the right-biased one to f^- . An example of such splitting is given by the Lax-Friderichs one:

$$f^\pm = \frac{1}{2}(f(u) \pm \alpha u),$$

where α is a positive number bigger than the eigenvalues of the Jacobian of the flux function either at global or at local level.

2.2.3.4 High-order TVD Runge-Kutta time discretization

To achieve high-order accuracy in time, the Total Variation Diminishing (TVD) Runge-kutta method are commonly used, due to the fact that they ensure that the total variation of the solutions does not increase under some time step restrictions: [44], [45], [52].

Let us write (2.2.8) in the form of a system of ODE:

$$u_t = L(u), \quad (2.2.13)$$

In the 3rd-order, 3 stages TVD Runge-Kutta method the solution is updated as follows:

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

with effective CFL $C_{eff} = 0.33$, which is known as the Shu-Osher method [52]. We will also consider the 4th-order method with 10 stages and effective CFL $C_{eff} = 0.6$,

$$\begin{aligned} u^{(1)} &= u^n + \frac{1}{6}\Delta t f(u^n), \\ u^{(i+1)} &= u^{(i)} + \frac{1}{6}\Delta t F(u^{(i)}), \quad i = 1, 2, 3, \\ u^{(5)} &= \frac{3}{5}u^n + \frac{2}{5}(u^{(4)} + \frac{1}{6}\Delta t F(u^{(4)})), \\ u^{(i+1)} &= u^{(i)} + \frac{1}{6}\Delta t F(u^{(i)}), \quad i = 5, 6, 7, 8, \\ u^{n+1} &= \frac{1}{25}u^n + \frac{9}{25}(u^{(4)} + \frac{1}{6}\Delta t F(u^{(4)})) + \frac{3}{5}(u^{(9)} + \frac{1}{6}\Delta t F(u^{(9)})). \end{aligned}$$

This method belongs to the family of the Strong Stability Preserving Runge-Kutta methods (SSPRK) see [25]. Also, it is known as the low storage SSPK_10.4 and was found by Ketcheson [46].

2.2.3.5 Lax-Wendroff type time discretizations

In [1] P.D. Lax and B. Wendroff proposed a numerical technique for solving approximately systems of hyperbolic conservation laws (2.2.1). Two essential contributions to the field were made in their article: Theorem 2.2.1 and the explicit second-order Lax-Wendroff method. For the scalar linear advection equation

$$u_t + au_x = 0,$$

in which the flux is the linear function $f(u) = au$, with a is a constant value, the derivation of the method is based on the second-order Taylor expansion:

$$u(x_i, t^{n+1}) = u(x_i, t^n) + \Delta t \partial_t^1 u(x_i, t^n) + \frac{\Delta t^2}{2} \partial_t^2 u(x_i, t^n) + O(\Delta t^3). \quad (2.2.14)$$

Then, assuming that the solution is smooth, one can replace time derivatives by space derivatives using the equation:

$$\partial_t^k u(x, t) = (-a)^k \partial_x^k u(x, t).$$

Finally, the space derivatives are approximated by centered 3-point formulas of numerical differentiation:

$$\begin{aligned}\partial_x u(x_i, t_n) &\cong \frac{1}{2\Delta x}(u(x_{i+1}, t_n) - u(x_{i-1}, t_n)), \\ \partial_x^{(2)} u(x_i, t_n) &\cong \frac{1}{\Delta x^2}(u(x_{i+1}, t_n) - 2u(x_i, t_n) + u(x_{i-1}, t_n))\end{aligned}$$

what leads to the Lax-Wendroff method:

$$u_i^{n+1} = u_i^n - \frac{a\Delta t}{2\Delta x}(u_{i+1}^n - u_{i-1}^n) + \frac{a^2\Delta t^2}{2\Delta x^2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n) \quad (2.2.15)$$

that can be written in conservative form:

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}(f_{i+1/2} - f_{i-1/2}), \quad (2.2.16)$$

where

$$f_{i+1/2} = \frac{a}{2}(u_i^n + u_{i+1}^n) - \frac{a^2\Delta t}{2\Delta x}(u_{i+1}^n - u_i^n). \quad (2.2.17)$$

It can also be written in the form:

$$u_i^{n+1} = b_{-1}u_{i-1}^n + b_0u_i^n + b_1u_{i+1}^n, \quad (2.2.18)$$

where, the coefficients b_k are functions of the Courant number

$$c = a \frac{\Delta t}{\Delta x}$$

as follows:

$$b_{-1} = \frac{1}{2}c(1+c), \quad b_0 = 1 - c^2, \quad b_1 = -\frac{1}{2}c(1-c).$$

The scheme has a 3rd local truncate error, is a second-order method and L^2 stable under the CFL condition

$$|c| \leq 1.$$

This scheme can be easily extended to linear systems of conservation laws

$$u_t + Au_x = 0,$$

where A is now a $N \times N$ matrix: the numerical method can be still written by (2.2.16) with

$$f_{i+1/2} = \frac{1}{2}A(u_i^n + u_{i+1}^n) - \frac{\Delta t}{2\Delta x}A^2(u_{i+1}^n - u_i^n). \quad (2.2.19)$$

Moreover for linear problems it can be extended to methods of higher order, as it will be seen in Chapter 3 following the same ansatz: consider the Taylor expansion in time of the solution, replace time derivatives by spacial derivatives using the equation, approach spacial derivatives using centered formulas of numerical differentiation.

2.2.3.6 Lax-Wendroff procedure for nonlinear problems

There are different way to extend the Lax-Wendroff methods to second-order methods for nolinear systems of balance laws. If we let $A(u)$ be the Jacobian of the flux function $f(u)$, a possible extension is (2.2.16) with numerical flux

$$f_{i+1/2} = \frac{1}{2} (f(u_i^n) + f(u_{i+1}^n)) - \frac{\Delta t}{2\Delta x} A_{i+1/2}^2 (u_{i+1}^n - u_i^n).$$

where $A_{i+1/2}$ is some approximation of $A(u(x_{i+1/2}, t_n))$. Another extensiona are given by the Richtmeyer two-step Lax-Wendroff method [53] or MacCormack's method [54].

The design of high-order methods for nonlinear problems that advance in time using Taylor expansions may be an interesting alternative to methods of lines combined with TVDRK methods that requires more and more stages as the order increases. The ansatz mentioned in the previous paragraph for linear problems can be followed: Taylor expansion in time - replacement of time derivatives by spacial derivatives using the equation - application of spatial derivatives using centered formulas of numerical differentiation. This is called the **Cauchy-Kowalewski** (CK) procedure.

In the ADER methods, (see [8]) the CK procedure is replaced by local space-time problems that are solved with a Galerkin method. The so-called $P_N P_M$ methods introduced in [9] that generalize ADER-WENO and DG methods also follow this approach. These methods can be applied both on structured and unstructured meshes with $CFL = 1$ condition for stability.

The Qiu and Shu finite difference Lax-Wendroff method [11] for nonlinear hyperbolic systems of conservation laws is also based on the CK procedure. Let us denote by $u_{t,i}^{(s)}$ the s -th order time derivative of u . The starting point is again the Taylor expansion:

$$u(x_i, t^{n+1}) \approx u_i + \Delta t u_t^{(1)}(x_i, t_n) + \frac{\Delta t^2}{2} u_t^{(2)}(x_i, t_n) + \frac{\Delta t^3}{6} u_t^{(3)}(x_i, t_n) + \dots + \frac{\Delta t^k}{k!} u_t^{(k)}(x_i, t_n). \quad (2.2.20)$$

Then the time derivatives $u_t^{(1)}, \dots, u_t^{(k)}$ are approximated as follows:

- The first time derivative $u_t^{(1)}$ is approached by using the conservative WENO reconstruction of order $(2r - 1)$, that will be denoted by WENO $(2r - 1)$, i.e.

$$u_t^{(1)}(x_i, t_n) = -\partial_x f(u(x_i, t_n)) \cong u_{t,i}^{(1)} = -\frac{1}{\Delta x} (\hat{f}_{i+1/2} - \hat{f}_{i-1/2}).$$

Observe that $r = 2, 3, 7$ produce a spatial reconstruction WENO3, WENO5 and WENO7 respectively.

- For the second time derivative one has

$$u_t^{(2)} = -(f^{(1)}(u)u_t^{(1)})_x,$$

where $f^{(1)}(u) = A(u)$ represents the Jacobian. Due to the factor Δ^2 in the Taylor expansion, this derivative can be approached with one order lower than $u_t^{(1)}$. This is done by applying to

$$g_i = f^{(1)}(u_i^n)u_{t,i}^{(1)} \quad (2.2.21)$$

a $(2r-2)$ th-order centered formula of numerical differentiation to approach the first-order spatial derivative at x_i . Let us denote by $u_{t,i}^{(2)}$ the obtained approximations.

- For the third time derivative one has

$$u_t^{(3)} = -(f^{(1)}(u)u^{(2)} + f^{(2)}(u)(u_t^{(1)})^2)_x. \quad (2.2.22)$$

where $f^{(2)}$ is the Hessian of f . This derivative is approximated by applying to

$$g_i = f^{(1)}(u_i^n)u_{t,i}^{(2)} + f^{(2)}(u_i^n)(u_{t,i}^{(1)})^2, \quad (2.2.23)$$

a $(2r-2)$ th-order centered formula of numerical differentiation to approach the first-order spatial derivative at x_i . Let us denote by $u_{t,i}^{(3)}$ the obtained approximations.

- For the fourth-order time derivative one has

$$u_t^{(4)} = -(f^{(1)}(u)u_t^{(2)} + 3f^{(2)}(u)u_t^{(1)}u_t^{(2)} + f^{(3)}(u)(u_t^{(1)})^3)_x. \quad (2.2.24)$$

This derivative is approximated by applying to

$$g_i = f^{(1)}(u_i)u_{t,i}^{(2)} + 3f^{(2)}(u_i)u_{t,i}^{(1)}u_{t,i}^{(2)} + f^{(3)}(u_i)(u_{t,i}^{(1)})^3 \quad (2.2.25)$$

a $(2r-4)$ th-order centered formula of numerical differentiation to approach the first-order spatial derivative at x_i . Let us denote by $u_{t,i}^{(4)}$ the obtained approximations.

- The derivatives $u_t^{(j)}$, $j = 5, \dots, k$ are computed in a similar way.

This procedure requires symbolic calculus and tensor products, as the second and higher-order time derivatives, when converted to spatial derivatives as before, involve expressions like $f^{(1)}(u)$, which is a matrix (the Jacobian) and $f^{(2)}(u)$, is a three-dimensional tensor, etc. Moreover, the set of values required to compute u_i^n , i.e. the stencil, increases with the order the method.

High-order methods based on Taylor expansions that avoid the CK procedure and the stencil growth will be discussed in the following chapters.

Chapter 3

Compact Approximate Taylor Method

In this Chapter a new family of numerical methods for nonlinear systems of conservation laws that are an extension of the high-order Lax-Wendroff methods for linear systems. Lax-Wendroff methods are based on Taylor expansions in time in which the time derivatives are transformed into spatial derivatives using the governing equations [2]-[3]-[4]. The spatial derivatives are then discretized by means of centered high-order differentiation formulas. This procedure allows to derive numerical methods of order $2p$, where p is an arbitrary integer, using a centered $(2p + 1)$ -points stencil that are L^2 stables.

The main difficulty to extend Lax-Wendroff methods to nonlinear problems comes from the transformation of time derivatives into spatial derivatives through the Cauchy-Kovalesky (CK) procedure, but this approach may be impractical from the computational point of view (symbolic calculus, tensor matrix, excessive computations...) In the context of ADER methods introduced by Toro and collaborators (see [5], [6], [7]), this difficulty have been circumvented by replacing the CK procedure by local space-time problems that are solved with a Galerkin method: see [8], [9].

We follow here the strategy introduced in [10] to avoid the CK procedure in which time derivatives are computed in a recursive way using high-order centered differentiation formulas combined with Taylor expansions in time. This strategy leads to high-order Lax-Wendroff Approximated methods (LAT) that are oscillatory close to discontinuities: in [10] they were combined with WENO reconstructions to compute the first time derivatives. The resulting methods (LAT) give non-oscillatory and accurate results.

Compact Approximated Taylor methods (CAT) circumvent the CK procedure using the same strategy as LAT methods. These methods are compact in the sense that the length of the stencils is minimal: $(2p + 1)$ -point stencils are used to get order $2p$ compared to $4p + 1$ -point stencils in LAT methods. The technique used to reduce the length of the stencil makes that the computational cost of a time step in CAT methods is bigger than

in LAT methods: the Taylor expansions are computed locally, so that the total number of expansions needed to update the numerical solution is multiplied by $(2p + 1)$. On the other hand, unlike LAT methods, CAT methods reduce to the standard high-order Lax-Wendroff methods when applied to linear problems and, due to this, they have better stability properties than LAT and allows one to increase the length of time steps, what compensates the extra cost of every time iteration: see [12].

The chapter is organized as follows. In Section 3.1 a review of high-order Lax-Wendroff methods for the linear transport equation is presented, including the study of the order, a heuristic study of the L^2 -stability, and a discussion about the computation and properties of the coefficients. Section 3.2 is devoted to their extension to nonlinear problems: first the AT technique is recalled and then CAT methods are presented. We show that they reduce to Lax-Wendroff methods when applied to a linear problem and we analyze the order of accuracy. In Section 3.3 the techniques considered to cure the spurious oscillations near the discontinuities are presented. In Section 3.4 CAT methods are compared in a number of test cases with WENO-RK methods and AT methods. The linear transport equation, Burgers equation, the 1D compressible Euler and the ideal Magnetohydrodynamics equations are considered.

3.1 The high-order Lax-Wendroff method for linear problems

Let us first consider the linear scalar equation:

$$u_t + au_x = 0. \quad (3.1.1)$$

We consider the numerical method:

$$u_i^{n+1} = u_i^n + \sum_{k=1}^m \frac{(-1)^k c^k}{k!} \sum_{j=-p}^p \delta_{p,j}^k u_{i+j}^n, \quad (3.1.2)$$

where $\{x_i\}$ are the nodes of a uniform mesh of step Δx ; u_i^n is an approximation of the point value of the solution at x_i at the time $n\Delta t$, where Δt is the time step; $p \geq 1$ is a natural number; $c = a\Delta t/\Delta x$; and $\delta_{p,j}^k$ are the coefficients of the centered interpolatory formula of numerical differentiation based on a $(2p + 1)$ -point stencil, i.e. the unique formula of the form

$$f^{(k)}(x_i) \simeq D_{p,i}^k(f, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f(x_{i+j}), \quad (3.1.3)$$

such that

$$p_f^{(k)}(x_i) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f(x_{i+j}), \quad \forall f,$$

where p_f is the Lagrange interpolation polynomial characterized by

$$p_f(x_{i+j}) = f(x_{i+j}), \quad j = -p, \dots, p.$$

Here $f^{(k)}$ represents the k -th derivative of a one-variable function f and $f^{(0)} = f$.

The expression of the numerical method is obtained by applying a Taylor expansion in time, and replacing time derivatives by space derivatives through the identities

$$\partial_t^k u = (-1)^k a^k \partial_x^k u, \quad k = 1, 2, \dots \quad (3.1.4)$$

3.1.1 Formulas of numerical differentiation

Besides (3.1.3) the following family of interpolatory formulas based on a $2p$ -point stencil will be used in this work:

$$f^{(k)}(x_i + q\Delta x) \simeq A_{p,i}^{k,q}(f, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} f(x_{i+j}), \quad (3.1.5)$$

i.e. $A_{p,i}^{k,q}(f, \Delta x)$ is the numerical differentiation formula that approximates the k -th derivative at the point $x_i + q\Delta x$ using the values of the function at the $2p$ points $x_{i-p+1}, \dots, x_{i+p}$. Observe that the coefficients, like in (3.1.3), do not depend on i .

Given a variable w , the following notation will be used:

$$D_{p,i}^k(w_*, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k w_{i+j},$$

$$A_{p,i}^{k,q}(w_*, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} w_{i+j},$$

to indicate that the formulas are applied to the approximations of w , w_i , and not to its exact point values $w(x_i)$. In cases where there are two or more indexes, the symbol $*$ will be used to indicate with respect to which the differentiation is applied. For instance:

$$\partial_x^k u(x_i + q\Delta x, t_n) \simeq A_{p,i}^{k,q}(u_*^n, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} u_{i+j}^n,$$

$$\partial_t^k u(x_i, t_n + q\Delta t) \simeq A_{p,n}^{k,q}(u_i^*, \Delta t) = \frac{1}{\Delta t^k} \sum_{r=-p+1}^p \gamma_{p,r}^{k,q} u_i^{n+r}.$$

Using this notation, (3.1.2) writes as follows:

$$u_i^{n+1} = u_i^n + \sum_{k=1}^m \frac{(-1)^k a^k \Delta t^k}{k!} D_{p,i}^k(u_i^n, \Delta x). \quad (3.1.6)$$

Let us discuss some properties of the coefficients of the numerical differentiation formulas (3.1.3) and (3.1.5) and some relations between them that will be used in that follows. Since the coefficients are independent of Δx and i , we can consider, without loss of generality, the case $i = 0$, $x_0 = 0$, $\Delta x = 1$:

$$f^{(k)}(0) \simeq D_{p,0}^k(f, 1) = \sum_{j=-p}^p \delta_{p,j}^k f(j), \quad (3.1.7)$$

$$f^{(k)}(q) \simeq A_{p,0}^{k,q}(f, 1) = \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} f(j). \quad (3.1.8)$$

Since (3.1.7) is exact for polynomials of degree $\leq 2p$, by applying the formula to x^s , $s = 0, \dots, 2p$ at $x = 0$, we get that the coefficients have to satisfy the equalities

$$\sum_{j=-p}^p j^k \delta_{p,j}^k = k!, \quad \sum_{j=-p}^p j^s \delta_{p,j}^k = 0, \quad s \neq k, \quad 0 \leq s, k \leq 2p. \quad (3.1.9)$$

Analogously:

$$\sum_{j=-p+1}^p j^k \gamma_{p,j}^{k,0} = k!, \quad \sum_{j=-p+1}^p j^s \gamma_{p,j}^{k,0} = 0, \quad s \neq k, \quad 0 \leq s, k \leq 2p-1. \quad (3.1.10)$$

$$\sum_{j=-p+1}^p \gamma_{p,j}^{k,q} = \begin{cases} 1 & \text{if } k = 0, \\ 0 & \text{otherwise.} \end{cases} \quad (3.1.11)$$

As it is well known, the coefficients $\delta_{p,j}^k$ are related to the Lagrange basis polynomials

$$F_{p,j}(x) = \prod_{r=-p, r \neq j}^p \frac{(x-r)}{(j-r)}, \quad -p \leq j \leq p, \quad (3.1.12)$$

through the equalities:

$$\delta_{p,j}^k = F_{p,j}^{(k)}(0), \quad (3.1.13)$$

which allow us to write the Taylor expansion of $F_{p,j}$ centered at $x = 0$ as follows:

$$F_{p,j}(x) = \sum_{k=0}^{2p} \frac{\delta_{p,j}^k}{k!} x^k. \quad (3.1.14)$$

Proposition 1 *The coefficients $\delta_{p,j}^k$ of the formula (3.1.7), satisfy:*

$$\delta_{p,j}^k = (-1)^k \delta_{p,-j}^k; \quad (3.1.15)$$

$$\delta_{p,0}^k = 0 \text{ if } k \text{ is odd}; \quad (3.1.16)$$

$$\sum_{j=-p}^p \delta_{p,j}^k j^{(2p+1)} = 0 \text{ if } k \text{ is even}; \quad (3.1.17)$$

$$\sum_{j=-p}^p \delta_{p,j}^k j^{(2p+2)} = 0 \text{ if } k \text{ is odd}. \quad (3.1.18)$$

Proof. (3.1.15) is deduced from the equality

$$F_{p,-j}(x) = F_{p,j}(-x). \quad (3.1.19)$$

Using (3.1.15) we get (3.1.16). (3.1.17) and (3.1.18) are deduced from (3.1.15) and (3.1.16). □

Proposition 2 *For $k \geq 1$ the following relations hold:*

$$\delta_{p,p}^k = \gamma_{p,p}^{k-1,1/2}; \quad (3.1.20)$$

$$\delta_{p,j}^k = \gamma_{p,j}^{k-1,1/2} - \gamma_{p,j+1}^{k-1,1/2}, \quad j = -p+1, \dots, p-1; \quad (3.1.21)$$

$$\delta_{p,-p}^k = -\gamma_{p,-p+1}^{k-1,1/2}. \quad (3.1.22)$$

Proof. Let us consider the formulas

$$f^{(k-1)}(1/2) \simeq A_{p,0}^{k-1,1/2}(f, 1) = \sum_{j=-p+1}^p \gamma_{p,j}^{k-1} f(j), \quad (3.1.23)$$

$$f^{(k-1)}(-1/2) \simeq A_{p,-1}^{k-1,1/2}(f, 1) = \sum_{j=-p+1}^p \gamma_{p,j}^{k-1} f(j-1), \quad (3.1.24)$$

that are exact for polynomials of degree $\leq 2p - 1$. Let us consider now the formula

$$f^{(k)}(0) \simeq A_{p,0}^{k-1,1/2}(f, 1) - A_{p,-1}^{k-1,1/2}(f, 1). \quad (3.1.25)$$

If f is a polynomial of degree $2p$, then (3.1.23) and (3.1.24) are exact for f , furthermore

$$\begin{aligned} A_{p,0}^{k-1,1/2}(f, 1) - A_{p,-1}^{k-1,1/2}(f, 1) &= f^{(k-1)}(1/2) - f^{(k-1)}(-1/2) \\ &= f^{(k)}(0), \end{aligned}$$

where we have used that the formula

$$g'(0) \simeq g(1/2) - g(-1/2),$$

is exact for polynomials of degree 1. Therefore, (3.1.25) coincide with (3.1.7). The proof is finished by writing (3.1.25) in the form

$$\begin{aligned} f^{(k)}(0) &\simeq \gamma_{p,p}^{k-1,1/2} f(p) + (\gamma_{p,p-1}^{k-1,1/2} - \gamma_{p,p}^{k-1,1/2}) f(p-1) + \dots \\ &\quad + (\gamma_{p,-p+1}^{k-1,1/2} - \gamma_{p,-p+2}^{k-1,1/2}) f(-p+1) - \gamma_{p,-p+1}^{k-1,1/2} f(-p), \end{aligned}$$

and matching the weights. □

Proposition 3 Given $1 \leq k \leq 2p - 1$, $0 \leq s \leq k$:

$$\sum_{j=-p+1}^p \gamma_{p,j}^{s,q} \gamma_{p,l}^{k-s,j} = \gamma_{p,l}^{k,q}, \quad l = -p+1, \dots, p. \quad (3.1.26)$$

Proof. The proof is similar to the one of the preceding in Proposition 2: consider the formula

$$f^{(k)}(q) \simeq \sum_{j=-p+1}^p \gamma_{p,j}^{s,q} f_j^{(k-s)},$$

with

$$f_j^{(k-s)} = \sum_{l=-p+1}^p \gamma_{p,l}^{k-s,j} f(l);$$

check that it is exact for polynomials of degree $2p - 1$; write it in the form:

$$f^{(k)}(q) \simeq \sum_{l=-p+1}^p \left(\sum_{j=-p+1}^p \gamma_{p,j}^{s,q} \gamma_{p,l}^{k-s,j} \right) f(l);$$

and match its weights with those of (3.1.8). □

3.1.2 Conservative form

From the proof of Proposition 2 we deduce an alternative form for (3.1.3):

$$f^{(k)}(x_i) \simeq \frac{1}{\Delta x} \left(A_{p,i}^{k-1,1/2}(f, \Delta x) - A_{p,i-1}^{k-1,1/2}(f, \Delta x) \right). \quad (3.1.27)$$

Using this form in (3.1.6), the numerical method (3.1.2) can be written as:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} \left(F_{i-1/2}^p - F_{i+1/2}^p \right), \quad (3.1.28)$$

with

$$F_{i+1/2}^p = \sum_{k=1}^{2p} (-1)^{k-1} \frac{a^k \Delta t^{k-1}}{k!} A_{p,i}^{k-1,1/2}(u_*^n, \Delta x). \quad (3.1.29)$$

Using (3.1.11) it is straightforward to verify that $F_{i+1/2}^p$ is a consistent numerical flux, what proves that (3.1.2) is a conservative method.

3.1.3 Computation of the coefficients: an iterative algorithm

Notice that (3.1.9) constitutes a $(2p+1) \times (2p+1)$ linear system with a Vandermonde matrix that can be used to compute $\delta_{p,i}^k$. Nevertheless, as it is well-known, this system is ill-conditioned, so that it is recommendable to compute them by using an alternative algorithm: we adapt the recursive algorithm proposed in [55]. The following notation is adopted:

$$\delta_{p,j}^k = 0 \text{ if } k > 2p \text{ or } k < 0.$$

Let us derive some recurrence formulas to compute the coefficients:

1. $\delta_{p,j}^k$ for $j = 0, \dots, p-1$.

From (3.1.12), we obtain

$$F_{p,j}(x) = \frac{(x+p) \cdot \widehat{(x-j)} \cdot (x-p)}{(j+p) \cdot \widehat{(j-j)} \cdot (j-p)}, \quad (3.1.30)$$

where factors with a hat have to be left out. Then one has

$$F_{p,j}(x) = \frac{x^2 - p^2}{j^2 - p^2} F_{p-1,j}(x). \quad (3.1.31)$$

Using then the Taylor expansions (3.1.14) in (3.1.31) we get

$$\frac{\delta_{p,j}^k}{k!} = \frac{1}{j^2 - p^2} \left[\frac{\delta_{p-1,j}^{k-2}}{(k-2)!} - p^2 \frac{\delta_{p-1,j}^k}{k!} \right],$$

that is

$$\delta_{p,j}^k = \frac{1}{p^2 - k^2} \left[p^2 \delta_{p-1,j}^k - k(k-1) \delta_{p-1,j}^{k-2} \right], \quad (3.1.32)$$

2. $\delta_{p,j}^k$ with $j = p$.

Substituting $j=p$ in (3.1.12), we get

$$\begin{aligned} F_{p,p}(x) &= \frac{(x+p) \cdot (x-p+1)}{(2p) \cdot (2)(1)} \\ &= \frac{1}{(2p)!} (x+p)(x-p+1) \frac{(x+p+1) \cdot (x-p+2)}{(2p-2) \cdot (1)} (2p-2)! \\ &= \frac{1}{(2p)(2p-1)} (x^2 + x - p(p-1)) F_{p-1,p-1}(x), \end{aligned} \quad (3.1.33)$$

using (3.1.14) we obtain

$$\frac{\delta_{p,p}^k}{k!} = \frac{1}{2p(2p-1)} \left[\frac{\delta_{p-1,p-1}^{k-2}}{(k-2)!} + \frac{\delta_{p-1,p-1}^{k-1}}{(k-1)!} - p(p-1) \frac{\delta_{p-1,p-1}^k}{k!} \right], \quad (3.1.34)$$

in explicit form

$$\delta_{p,p}^k = \frac{1}{2p(2p-1)} \left[k(k-1) \delta_{p-1,p-1}^{k-2} + k \delta_{p-1,p-1}^{k-1} - p(p-1) \delta_{p-1,p-1}^k \right]. \quad (3.1.35)$$

3. $\delta_{p,j}^k$ for $j = -p, \dots, -1$. (3.1.15) is used.

The algorithm is computed only once in increasing order of p for a specific k -th numerical derivative. The coefficients $\gamma_{p,j}^{k,q}$ are computed using the algorithms described in [55],[56] and $\gamma_p^{k,1/2}$ are obtained from Prop. 2.

3.1.4 Order of accuracy

Proposition 4 *The formula of numerical differentiation (3.1.3) has order of accuracy $\alpha_k - k$,*

with,

$$\alpha_k = \begin{cases} 2p+1 & \text{if } k \text{ is odd,} \\ 2p+2 & \text{if } k \text{ is even.} \end{cases}$$

k \ p	1	2	3	4
1	2	4	6	8
2	2	4	6	8
3		2	4	6
4		2	4	6
5			2	4
6			2	4
7				2
8				2

Table 3.1: Order of the formula (3.1.3).

Proof. Let f be a function of class C^{α_k+1} . Applying Taylor expansions and properties (3.1.9) and (3.1.17), we obtain:

if k is odd;

$$\frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f(x_{i+j}) = f^{(k)}(x_i) + \sum_{j=-p}^p \delta_{p,j}^k j^{2p+1} \frac{\Delta x^{2p+1-k}}{(2p+1)!} f^{(2p+1)}(x_i) + \mathcal{O}(\Delta x^{2p+2-k}), \quad (3.1.36)$$

and if k is pair,

$$\frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f(x_{i+j}) = f^{(k)}(x_i) + \sum_{j=-p}^p \delta_{p,j}^k j^{2p+2} \frac{\Delta x^{2p+2-k}}{(2p+2)!} f^{(2p+2)}(x_i) + \mathcal{O}(\Delta x^{2p+3-k}), \quad (3.1.37)$$

in compact form

$$\frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f(x_{i+j}) = f^{(k)}(x_i) + \varphi_k \frac{\Delta x^{\alpha_k-k}}{\alpha_k!} f^{(\alpha_k)}(x_i) + \mathcal{O}(\Delta x^{\alpha_k-k+1}), \quad (3.1.38)$$

where

$$\varphi_k = \sum_{j=-p}^p \delta_{p,j}^k j^{\alpha_k}. \quad (3.1.39)$$

□

Table 3.1 shows the order of (3.1.3) for different values of p and k .

Proposition 5 *The discretization error of the numerical method (3.1.2) is of order $\mathcal{O}(\Delta t^{m+1} + \Delta x^{2p+1})$.*

Proof. Let u be a smooth enough solution of (3.1.1). Using Proposition 4 we obtain

$$\begin{aligned}
& u(x_i, t_{n+1}) - u(x_i, t_n) - \sum_{k=1}^m \frac{(-1)^k c^k}{k!} \sum_{j=-p}^p \delta_{p,j}^k u(x_{i+j}, t_n) \\
&= u(x_i, t_{n+1}) - u(x_i, t_n) \\
&\quad - \sum_{k=1}^m \frac{(-1)^k a^k \Delta t^k}{k!} \left(\partial_x^k u(x_i, t_n) + \varphi_k \frac{\Delta x^{\alpha_k - k}}{\alpha_k!} \partial_x^{\alpha_k} u(x_i, t_n) + \mathcal{O}(\Delta x^{\alpha_k - k + 1}) \right) \\
&= u(x_i, t_{n+1}) - u(x_i, t_n) - \sum_{k=1}^m \frac{\Delta t^k}{k!} \partial_t^k u(x_i, t_n) \\
&\quad - \sum_{k=1}^m \varphi_k \frac{(-1)^k c^k}{k! \alpha_k!} \Delta x^{\alpha_k} \partial_x^{\alpha_k} u(x_i, t_n) + O(\Delta x^{\alpha_k + 1}) \\
&= \frac{1}{(m+1)!} \partial_t^{m+1} u(x_i, t_n) \Delta t^{m+1} \\
&\quad + \left(\sum_{k=0}^{p-1} \frac{\varphi_{2k+1} c^{2k+1}}{(2p+1)!(2k+1)!} \right) \partial_x^{2p+1} u(x_i, t_n) \Delta x^{2p+1} + O(\Delta t^{m+2} + \Delta x^{2p+2}),
\end{aligned}$$

where (3.1.4) has been used. □

As a consequence, the order of accuracy of (3.1.2) is $\min(m, 2p)$. Therefore, the optimal combination of these parameters is $m = 2p$. From now on, we shall assume that this relation holds.

3.1.5 Modified equation and stability

Taking into account that $m = 2p$ and (3.1.18), we find that the local discretization error is as follows:

$$\begin{aligned}
& u(x_i, t_{n+1}) - u(x_i, t_n) - \sum_{k=1}^m \frac{(-1)^k c^k}{k!} \sum_{j=-p}^p \delta_{p,j}^k u(x_{i+j}, t_n) \\
&= \frac{1}{(2p+1)!} \partial_t^{2p+1} u(x_i, t_n) \Delta t^{2p+1} + \frac{1}{(2p+2)!} \partial_t^{2p+2} u(x_i, t_n) \Delta t^{2p+2} \\
&\quad + \left(\sum_{k=0}^{p-1} \frac{\varphi_{2k+1} c^{2k+1}}{(2p+1)!(2k+1)!} \right) \partial_x^{2p+1} u(x_i, t_n) \Delta x^{2p+1} \\
&\quad - \left(\sum_{k=1}^p \frac{\varphi_{2k} c^{2k}}{(2p+2)!(2k)!} \right) \partial_x^{2p+2} u(x_i, t_n) \Delta x^{2p+2} + O(\Delta x^{2p+3}).
\end{aligned}$$

Using (3.1.39) and (3.1.14) we get:

$$\begin{aligned}
\sum_{k=0}^{p-1} \frac{\varphi_{2k+1} c^{2k+1}}{(2k+1)!} &= \sum_{j=-p}^p \left(\sum_{k=0}^{p-1} \frac{\delta_{p,j}^{2k+1} c^{2k+1}}{(2k+1)!} \right) j^{2p+1} \\
&= \frac{1}{2} \sum_{j=-p}^p \left(\sum_{l=1}^{2p} \left(\frac{\delta_{p,j}^l}{l!} - \frac{\delta_{p,-j}^l}{l!} \right) c^l \right) j^{2p+1} \\
&= \frac{1}{2} \sum_{j=-p}^p (F_{p,j}(c) - F_{p,-j}(c)) j^{2p+1} \\
&= \frac{1}{2} \left[\sum_{j=-p}^p F_{p,j}(c) j^{2p+1} - \sum_{j=-p}^p F_{p,j}(-c) j^{2p+1} \right] \\
&= \frac{1}{2} [q(c) - q(-c)]
\end{aligned}$$

where $q(c)$ is the polynomial of degree $\leq 2p$ that interpolates the points

$$\{(-p, (-p)^{2p+1}), \dots, (0, 0), \dots, (p, p^{2p+1})\}.$$

Since q is clearly an odd function, we finally obtain:

$$\sum_{k=0}^{p-1} \frac{\varphi_{2k+1} c^{2k+1}}{(2k+1)!} = q(c). \quad (3.1.40)$$

Reasoning in a similar way, we obtain:

$$\begin{aligned}
\sum_{k=1}^p \frac{\varphi_{2k} c^{2k}}{(2k)!} &= \frac{1}{2} \sum_{j=-p}^p \left(\sum_{l=1}^{2p} \left(\frac{\delta_{p,j}^l}{l!} + \frac{\delta_{p,-j}^l}{l!} \right) c^l \right) j^{2p+2} \\
&= \frac{1}{2} \sum_{j=-p}^p (F_{p,j}(c) + F_{p,-j}(c)) j^{2p+2} \\
&= \frac{1}{2} \left[\sum_{j=-p}^p F_{p,j}(c) j^{2p+2} + \sum_{j=-p}^p F_{p,j}(-c) j^{2p+2} \right] \\
&= \frac{1}{2} [r(c) + r(-c)] \\
&= r(c).
\end{aligned} \quad (3.1.41)$$

where r is the polynomial of degree $\leq 2p$ that interpolates the points

$$\{(-p, (-p)^{2p+2}), \dots, (0, 0), \dots, (p, p^{2p+2})\}.$$

Using now (3.1.4), (3.1.40), and (3.1.41), we can write the local discretization error as follows:

$$\begin{aligned} u(x_i, t_{n+1}) - u(x_i, t_n) &= \sum_{k=1}^m \frac{(-1)^k c^k}{k!} \sum_{j=-p}^p \delta_{p,j}^k u(x_{i+j}, t_n) \\ &= \frac{h_1(c)}{(2p+1)!} \partial_x^{2p+1} u(x_i, t_n) \Delta x^{2p+1} - \frac{h_2(c)}{(2p+2)!} \partial_x^{2p+2} u(x_i, t_n) \Delta x^{2p+2} + \mathcal{O}(\Delta^{2p+3}), \end{aligned}$$

with

$$h_1(c) = q(c) - c^{2p+1}, \quad (3.1.42)$$

$$h_2(c) = r(c) - c^{2p+2}. \quad (3.1.43)$$

Therefore, the numerical method solves with order $\mathcal{O}(\Delta^{2p+2})$ the following modified equation

$$u_t + au_x = \mu_1 \partial_x^{2p+1} u - \mu_2 \partial_x^{2p+2} u, \quad (3.1.44)$$

where

$$\mu_1 = \frac{h_1(c)}{(2p+1)! \Delta t} \Delta x^{2p+1}, \quad \mu_2 = \frac{h_2(c)}{(2p+2)! \Delta t} \Delta x^{2p+2}. \quad (3.1.45)$$

Following the heuristic theory proposed in [57] to study the stability in the small wave-number limit, we look for an elementary solution $u(x, t)$ of (3.1.45) of the form

$$u(x, t) = e^{\alpha t} \cdot e^{ikx},$$

where α is complex number, and

$$\begin{aligned} u_t^{(1)} &= \alpha e^{\alpha t} \cdot e^{ikx}, \\ u_x^{(2p)} &= e^{\alpha t} \cdot (-1)^p k^{2p} e^{ikx}, \\ u_x^{(2p+1)} &= e^{\alpha t} \cdot i(-1)^p k^{2p+1} e^{ikx}, \end{aligned}$$

The following equality has to be satisfied:

$$\alpha u + ikau = \mu_1 (-1)^p i k^{2p+1} u + \mu_2 (-1)^p k^{2p+2} u.$$

Therefore:

$$\alpha = -\mu_2 (-1)^{p+1} k^{2p+2} - (ka - \mu_1 (-1)^{p+1} k^{2p+1}) i.$$

The numerical method is thus expected to be stable if the real part is negative, i.e.

$$\mu_2(-1)^p \leq 0,$$

or, equivalently

$$h_2(c)(-1)^p \leq 0. \quad (3.1.46)$$

h_2 is an even polynomial of degree $2p + 2$ such that

$$\lim_{c \rightarrow \pm\infty} h_2(c) = -\infty.$$

Moreover, 0 is a double root of h_2 and $\pm 1, \dots, \pm p$ are single roots. Analyzing the change of signs of h_2 , we obtain:

$$\begin{aligned} h_2(c) &\leq 0, & \forall c \in [0, 1] & \text{ if } p \text{ even,} \\ h_2(c) &\geq 0, & \forall c \in [0, 1] & \text{ if } p \text{ odd,} \end{aligned}$$

and thus (3.1.46) is satisfied if $c \in [0, 1]$ (see Figure 3.1).

This argument shows that the method is expected to be stable at least for small wave-numbers under the standard CFL condition $c \leq 1$. In fact, in [57] (see also [11]) it has been shown that (3.1.46) is a necessary condition for stability in the von Neumann sense. The analysis of the sufficiency of this condition is out of the scope of this thesis. Nevertheless the numerical experiments seem to confirm that the method is L^2 -stable under the standard CFL condition.

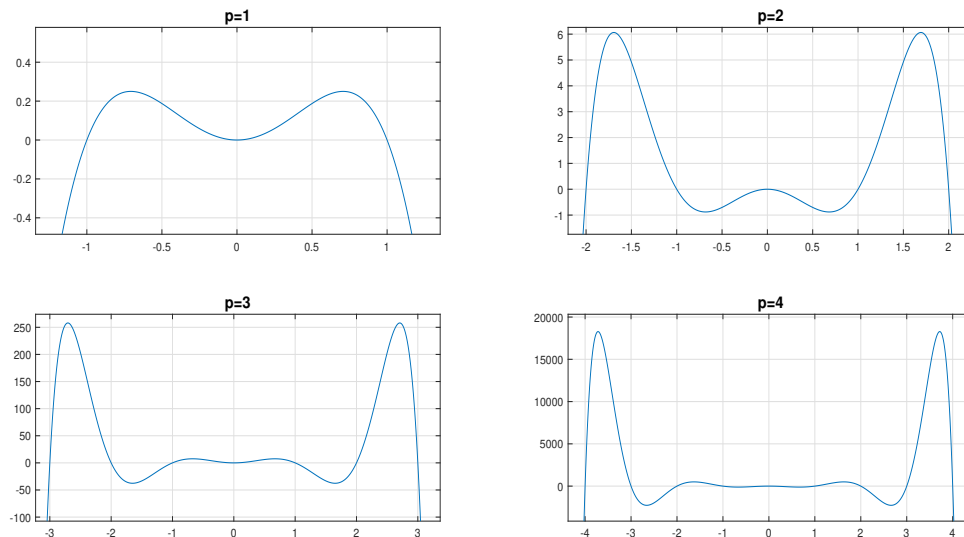


Figure 3.1: Function $h_2(c)$ for $p = 1, \dots, 4$.

3.2 Extension to nonlinear problems

3.2.1 Approximate Taylor method

Following [10], instead of using the Cauchy-Kovalevskaya process to extend (3.1.28)-(3.1.29) to nonlinear problems

$$u_t + f(u)_x = 0, \quad (3.2.1)$$

we use the equalities

$$\partial_t^k u = -\partial_x \partial_t^{k-1} f(u). \quad (3.2.2)$$

To derive the expression of the numerical method, let us suppose that approximations

$$\tilde{f}_i^{(k-1)} \cong \partial_t^{k-1} f(u)(x_i, t_n),$$

are available. Then,

$$\partial_t^k u(x_i, t_n) \cong \tilde{u}_i^{(k)} = -D_{p_{k-1}, i}^1(\tilde{f}_*^{(k-1)}, \Delta x) = -\frac{1}{\Delta x} \sum_{j=-p_{k-1}}^{p_{k-1}} \delta_{p_{k-1}, j}^1 \tilde{f}_{i+j}^{(k-1)},$$

being

$$p_k = \lceil (p - k/2) \rceil, \quad (3.2.3)$$

where, $\lceil \cdot \rceil$ denotes the ceiling function.

Using these approximations to approximate the Taylor expansion, we obtain the method

$$u_i^{n+1} = u_i^n + \sum_{k=0}^{2p} \frac{\Delta t^k}{k!} \tilde{u}_i^{(k)}. \quad (3.2.4)$$

Equivalently, using (3.1.27), we can write the numerical method in conservative form (3.1.28) with numerical flux

$$F_{i+1/2}^p = \sum_{k=1}^{2p} \frac{\Delta t^{k-1}}{k!} A_{p_{k-1}, i}^{0,1/2}(\tilde{f}_*^{(k-1)}, \Delta x), \quad (3.2.5)$$

being

$$A_{p_{k-1}, i}^{0,1/2}(\tilde{f}_*^{(k-1)}, \Delta x) = \sum_{j=-p_{k-1}+1}^{p_{k-1}} \gamma_{p_{k-1}, j}^{0,1/2} \tilde{f}_{i+j}^{(k-1)}. \quad (3.2.6)$$

Now, to compute the approximations $\tilde{f}_i^{(k-1)}$, new Taylor expansions in time are used recursively as follows:

- $k = 1$: compute $\tilde{f}_i^{(0)} = f(u_i^n)$.

- For $k = 2 \dots 2p$:

- Apply correspondent boundary conditions on $\tilde{f}_i^{(k-2)}$

- Compute

$$\tilde{u}_i^{(k-1)} = -D_{p_{k-2},i}^1(\tilde{f}_*^{(k-2)}, \Delta x).$$

- Compute

$$\tilde{f}_i^{k-1,n+r} = f \left(u_i^n + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} \tilde{u}_i^{(l)} \right), \quad r = -p_{k-1}, \dots, p_{k-1}.$$

- Compute

$$\tilde{f}_i^{(k-1)} = D_{p_{k-1},n}^{k-1}(\tilde{f}_i^{k-1,*}, \Delta t),$$

where

$$D_{p_{k-1},n}^{k-1}(\tilde{f}_i^{k-1,*}, \Delta t) = \frac{1}{\Delta t^{k-1}} \sum_{r=-p_{k-1}}^{p_{k-1}} \delta_{p_{k-1},r}^{k-1} \tilde{f}_i^{k-1,n+r}.$$

Observe that Taylor expansions are used to approximate $f(u(x_i, t_n + r\Delta t))$ and once these approximations have been computed, the centered formula of numerical differentiation (3.1.3) is used to approximate the temporal derivatives.

This method is order $2p$, but it is not a generalization of (3.1.2) in the sense that this latter method is not recovered if $f(u) = au$. To see this, consider $p = 1$ and $f(u) = au$: it can be easily checked that (3.2.4) writes as follows

$$u_i^{n+1} = u_i^n - \frac{c}{2}(u_{i+1}^n - u_{i-1}^n) - \frac{c^2}{8}(u_{i+2}^n - 2u_i^n + u_{i-2}^n), \quad (3.2.7)$$

which is different from the standard Lax-Wendroff method: (3.2.4) is a $(4p + 1)$ -point method whose stability properties are worse than those of the standard Lax-Wendroff method. (see [2]).

3.2.2 Compact Approximate Taylor method

In order to prevent the increase of the stencil observed for Approximate Taylor methods, we consider a modification based on the conservative form of the method. The numerical flux $F_{i+1/2}^p$ will be computed using only the approximations

$$u_{i-p+1}^n, \dots, u_{i+p}^n, \quad (3.2.8)$$

so that the values used to update u_i^{n+1} are only those of the centered $(2p + 1)$ -point stencil, like in the linear case. In fact, we will show that this modification is a proper generalization of the Lax-Wendroff method for linear problems.

In order to be able to compute the numerical fluxes using only (3.2.8), for every i we will compute *local* approximations of

$$\partial_t^{k-1} f(u(x_{i-p+1}, t^n), \dots, \partial_t^{k-1} f(u(x_{i+p}, t^n),$$

that will be represented by

$$\tilde{f}_{i,j}^{(k-1)} \cong \partial_t^{k-1} f(u)(x_{i+j}, t_n), \quad j = -p+1, \dots, p.$$

These approximations are local in the sense that $i_1 + j_1 = i_2 + j_2$, does not imply that $\tilde{f}_{i_1, j_1}^{(k-1)} = \tilde{f}_{i_2, j_2}^{(k-1)}$. Once these approximations have been computed, the numerical flux is given by

$$F_{i+1/2}^p = \sum_{k=1}^{2p} \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x), \quad (3.2.9)$$

with

$$A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x) = \sum_{j=-p+1}^p \gamma_{p,j}^{0,1/2} \tilde{f}_{i,j}^{(k-1)}. \quad (3.2.10)$$

Now, given i , to compute the approximations $\tilde{f}_{i,j}^{(k-1)}$, new Taylor expansions in time are used recursively as follows:

- $k = 1$: compute $\tilde{f}_{i,j}^{(0)} = f(u_{i+j}^n)$, $j = -p+1, \dots, p$.
- For $k = 2 \dots 2p$:

- Compute

$$\tilde{u}_{i,j}^{(k-1)} = -A_{p,0}^{1,j}(\tilde{f}_{i,*}^{(k-2)}, \Delta x),$$

where

$$A_{p,0}^{1,j}(\tilde{f}_{i,*}^{(k-2)}, \Delta x) = \frac{1}{\Delta x} \sum_{r=-p+1}^p \gamma_{p,r}^{1,j} \tilde{f}_{i,r}^{(k-2)}.$$

- Compute

$$\tilde{f}_{i,j}^{k-1, n+r} = f \left(u_{i+j}^n + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} \tilde{u}_{i,j}^{(l)} \right), \quad j, r = -p+1, \dots, p.$$

- Compute

$$\tilde{f}_{i,j}^{(k-1)} = A_{p,n}^{k-1,0}(\tilde{f}_{i,j}^{k-1,*}, \Delta t), \quad j = -p+1, \dots, p.$$

with

$$A_{p,n}^{k-1,0}(\tilde{f}_{i,j}^{k-1,*}, \Delta t) = \frac{1}{\Delta t^{k-1}} \sum_{r=-p+1}^p \gamma_{p,r}^{k-1,0} \tilde{f}_{i,j}^{k-1, n+r}.$$

Notice that, unlike the Approximate Taylor methods (in which all the derivatives were approximated using the centered $(2p + 1)$ -point formula), in this algorithm the stencil $x_{i-p+1}, \dots, x_{i+p}$ is used for the space derivatives and the stencil $t_{n-p+1}, \dots, t_{n+p}$ for the time derivative.

Theorem 3.2.1 *The compact approximate Taylor method reduces to (3.1.2) when $f(u) = au$.*

Proof. For $k > 1$ we have:

$$\begin{aligned}
\tilde{f}_{i,j}^{(k-1)} &= \frac{1}{\Delta t^{k-1}} \sum_{r=-p+1}^p \gamma_{p,r}^{k-1,0} \tilde{f}_{i,j}^{k-1,n+r} \\
&= \frac{a}{\Delta t^{k-1}} \sum_{r=-p+1}^p \gamma_{p,r}^{k-1,0} \left(u_{i+j}^n + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} \tilde{u}_{i,j}^{(l)} \right) \\
&= \frac{a}{\Delta t^{k-1}} \left(\left(\sum_{r=-p+1}^p \gamma_{p,r}^{k-1,0} \right) u_{i+j}^n + \sum_{l=1}^{k-1} \frac{\Delta t^l}{l!} \left(\sum_{r=-p+1}^p \gamma_{p,r}^{k-1,0} r^l \right) \tilde{u}_{i,j}^{(l)} \right) \\
&= a \tilde{u}_{i,j}^{(k-1)},
\end{aligned}$$

where (3.1.10) has been used. On the other hand:

$$\begin{aligned}
\tilde{u}_{i,j}^{(k)} &= -\frac{1}{\Delta x} \sum_{r=-p+1}^p \gamma_{p,r}^{1,j} \tilde{f}_{i,r}^{(k-1)} \\
&= -\frac{a}{\Delta x} \sum_{r=-p+1}^p \gamma_{p,r}^{1,j} \tilde{u}_{i,r}^{(k-1)} \\
&= \frac{a^2}{\Delta x^2} \sum_{r=-p+1}^p \gamma_{p,r}^{1,j} \sum_{s=-p+1}^p \gamma_{p,s}^{1,r} \tilde{u}_{i,s}^{(k-2)} \\
&= \frac{a^2}{\Delta x^2} \sum_{s=-p+1}^p \left(\sum_{r=-p+1}^p \gamma_{p,r}^{1,j} \gamma_{p,s}^{1,r} \right) \tilde{u}_{i,s}^{(k-2)} \\
&= \frac{a^2}{\Delta x^2} \sum_{s=-p+1}^p \gamma_{p,s}^{2,j} \tilde{u}_{i,s}^{(k-2)},
\end{aligned}$$

where (3.1.26) has been used. By recurrence:

$$\tilde{u}_{i,j}^{(k)} = \frac{(-1)^k a^k}{\Delta x^k} \sum_{r=-p+1}^p \gamma_{p,r}^{k,j} u_{i+r}^n. \quad (3.2.11)$$

Next,

$$\begin{aligned}
A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x) &= \frac{1}{\Delta x} \sum_{j=-p+1}^p \gamma_{p,j}^{0,1/2} \tilde{f}_{i,j}^{(k-1)} \\
&= \frac{a}{\Delta x} \sum_{j=-p+1}^p \gamma_{p,j}^{0,1/2} \tilde{u}_{i,j}^{(k-1)} \\
&= (-1)^{k-1} \frac{a^k}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{0,1/2} \sum_{r=-p+1}^p \gamma_{p,r}^{k-1,j} u_{i+r}^n \\
&= (-1)^{k-1} \frac{a^k}{\Delta x^k} \sum_{r=-p+1}^p \left(\sum_{j=-p+1}^p \gamma_{p,j}^{0,1/2} \gamma_{p,r}^{k-1,j} \right) u_{i+r}^n \\
&= (-1)^{k-1} \frac{a^k}{\Delta x^k} \sum_{r=-p+1}^p \gamma_{p,j}^{k-1,1/2} u_{i+r}^n \\
&= (-1)^{k-1} a^k A_{p,i}^{k-1,1/2}(u_*^n, \Delta x),
\end{aligned}$$

where (3.1.26) has been used. Finally,

$$\begin{aligned}
F_{i+1/2}^p &= \sum_{k=1}^{2p} \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x) \\
&= \sum_{k=1}^{2p} (-1)^{k-1} \frac{a^k \Delta t^{k-1}}{k!} A_{p,i}^{k-1,1/2}(u_*^n, \Delta x),
\end{aligned}$$

what is the numerical flux (3.1.29) corresponding to (3.1.2), as we wanted to prove. \square

As a consequence, we obtain that the compact approximate Taylor method is linearly stable (in the L^2 sense) under the usual CFL condition

$$\max_i (|f'(u_i)|) \frac{\Delta t}{\Delta x} \leq 1. \quad (3.2.12)$$

Theorem 3.2.2 *The compact approximate Taylor method is order $2p$.*

Proof. Let us perform a step of the method starting from the point values at time t^n , $u(x_i, t_n)$, of a smooth enough exact solution. We assume that $\Delta t/\Delta x$ remains constant.

First we have:

$$\tilde{u}_{i,j}^{(1)} = -A_{p,0}^{1,j}(\tilde{f}_{i,*}^{(0)}, \Delta x) = -\partial_x f(u)(x_{i+j}, t_n) + O(\Delta x^{2p-1}) = \partial_t u(x_{i+j}, t_n) + O(\Delta x^{2p-1}).$$

Next:

$$\tilde{f}_{i,j}^{1,n+r} = f(u(x_{i+j}, t_n) + \tilde{u}_{i,j}^{(1)} r \Delta t) = f(P_{i,j}^1(r \Delta t)) + O(\Delta x^{2p}),$$

where

$$P_{i,j}^1(s) = u(x_{i+j}, t_n) + s \partial_t u(x_{i+j}, t_n),$$

is the first-order Taylor polynomial in time of u in (x_{i+j}, t_n) . Then

$$\begin{aligned} \tilde{f}_{i,j}^{(1)} &= A_{p,n}^{1,0}(\tilde{f}_{i,j}^{k,*}, \Delta t) \\ &= \frac{1}{\Delta t} \sum_{r=-p+1}^p \gamma_{p,j}^{1,0} \tilde{f}_{i,j}^{1,n+r} \\ &= \frac{1}{\Delta t} \sum_{r=-p+1}^p \gamma_{p,j}^{1,0} f(P_{i,j}^1(r \Delta t)) + O(\Delta x^{2p}) \\ &= \frac{1}{\Delta t} \sum_{r=-p+1}^p \gamma_{p,j}^{1,0} \sum_{k=0}^{2p-1} \frac{1}{k!} d^k(f \circ P_{i,j}^1)(t_n) r^k \Delta t^k + O(\Delta x^{2p-1}) \\ &= \frac{1}{\Delta t} \sum_{k=0}^{2p-1} \frac{1}{k!} d^k(f \circ P_{i,j}^1)(t_n) \Delta t^k \sum_{r=-p+1}^p \gamma_{p,j}^{1,0} r^k + O(\Delta x^{2p-1}) \\ &= d^1(f \circ P_{i,j}^1)(t_n) + O(\Delta x^{2p-1}) \\ &= \partial_t f(u)(x_{i+j}, t_n) + O(\Delta x^{2p-1}), \end{aligned}$$

where (3.1.10) has been used. This result can be extended by induction to every k between 1 and $2p - 1$ as follows:

$$\tilde{f}_{i,j}^{(k)} = \partial_t^k f(u)(x_{i+j}, t_n) + O(\Delta t^{2p-k}), \quad k = 1, \dots, 2p - 1. \quad (3.2.13)$$

Using this equality we get:

$$\begin{aligned} &u(x_i, t_{n+1}) - u(x_i, t_n) + \frac{\Delta t}{\Delta x} \left(F_{i+1/2}^p - F_{i-1/2}^p \right) \\ &= u(x_i, t_{n+1}) - u(x_i, t_n) + \frac{1}{\Delta x} \sum_{k=1}^{2p} \frac{\Delta t^k}{k!} \left(A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x) - A_{p,0}^{0,1/2}(\tilde{f}_{i-1,*}^{(k-1)}, \Delta x) \right) \\ &= u(x_i, t_{n+1}) - u(x_i, t_n) + \frac{1}{\Delta x} \sum_{k=1}^{2p} \frac{\Delta t^k}{k!} \left(A_{p,i}^{0,1/2}(\partial_t^{k-1} f(u), \Delta x) - A_{p,i-1}^{0,1/2}(\partial_t^{k-1} f(u), \Delta x) \right) \\ &\quad + O(\Delta x^{2p+1}) \\ &= u(x_i, t_{n+1}) - u(x_i, t_n) + \frac{1}{\Delta x} \sum_{k=1}^{2p} \frac{\Delta t^k}{k!} D_{p,i}^1(\partial_t^{k-1} f(u), \Delta x) + O(\Delta x^{2p+1}) \end{aligned}$$

$$\begin{aligned}
&= u(x_i, t_{n+1}) - u(x_i, t_n) + \frac{1}{\Delta x} \sum_{k=1}^{2p} \frac{\Delta t^k}{k!} \partial_t^{k-1} f(u)(x_i, t_n) + O(\Delta x^{2p+1}) \\
&= u(x_i, t_{n+1}) - u(x_i, t_n) - \frac{1}{\Delta x} \sum_{k=1}^{2p} \frac{\Delta t^k}{k!} \partial_t^k u(x_i, t_n) + O(\Delta x^{2p+1}) \\
&= O(\Delta x^{2p+1}).
\end{aligned}$$

□

Remark: In the Approximate Taylor method proposed in [10] the derivatives $\tilde{u}_i^{(k+1)}$ are computed by applying the $2p_k+1$ -point centered differentiation formula for first derivatives to $\tilde{f}_i^{(k)}$, where p_k is given by (3.2.3): notice that p_k decreases as k increases. The same reduction of the stencil used to compute $\tilde{u}_{i,j}^{(k)}$ could be applied here, what would allow us to reduce the number of computations while preserving the overall order of accuracy. Nevertheless, the resulting method will not be an extension of the linear Lax-Wendroff method. On the other hand, the CPU reduction will be not significant.

3.2.3 Examples of CAT schemes

In this section the expressions of the CAT2 and CAT4 methods in 1D (second and fourth-order respectively) are explicitly given. Since the method is conservative, only the computation of the numerical flux (3.2.9) has to be given.

3.2.3.1 Second-order compact approximate Taylor method

Let us consider (3.2.9) computed with $p = 1$, then the second-order CAT numerical flux is:

$$F_{i+1/2}^1 = \sum_{k=1}^2 \frac{\Delta t^{k-1}}{k!} A_{1,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x) \quad (3.2.14)$$

$$= \frac{1}{2}(\tilde{f}_{i,1}^{(0)} + \tilde{f}_i^{(0)}) + \frac{1}{2}(\tilde{f}_{i,1}^{(1)} + \tilde{f}_{i,0}^{(1)}) \quad (3.2.15)$$

$$= \frac{1}{2}(f_{i,1}^n + f_{i,0}^n) + \frac{1}{4}(\tilde{f}_{i,1}^{1,n+1} + \tilde{f}_{i,0}^{1,n+1} - f_{i,1}^n - f_{i,0}^n) \quad (3.2.16)$$

$$= \frac{1}{4}(\tilde{f}_{i,0}^{1,n+1} + \tilde{f}_{i,0}^{1,n+1} + f_{i,1}^n + f_{i,0}^n), \quad (3.2.17)$$

where

$$\tilde{f}_{i,s}^{1,n+1} = f(u_{i,s}^n + \Delta t \tilde{u}_{t,i,s}^{(1)}), \quad (3.2.18)$$

$$= f(u_{i,s}^n - \frac{\Delta t}{\Delta x}(u_{i,1}^n - u_{i,0}^n)), \quad s = \{0, 1\}. \quad (3.2.19)$$

Observe that if $f(u) = au$ then (3.2.14) it reduces to the standard second-order Lax-Wendroff method.

3.2.3.2 Fourth-order compact approximate Taylor method

Calculate (3.2.9) for $p = 2$ is not so straightforward as in second-order scheme. To simplify, we will compute each component of (3.2.9) by separated i.e.

$$\kappa_{i+1/2}^k = A_{2,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x), \quad \text{for } k = 1, 2, 3, 4. \quad (3.2.20)$$

then procedure is as follows:

- $\kappa_{i+1/2}^1$: First the assignment

$$\tilde{f}_{i,j}^{(0)} = f(u_{i+j}^n), \quad j = -1, \dots, 2,$$

is done and then:

$$\kappa_{i+1/2}^1 = A_{2,0}^{0,1/2}(\tilde{f}_{i,*}^{(0)}, \Delta x) = \frac{-\tilde{f}_{i,-1}^{(0)} + 7\tilde{f}_{i,0}^{(0)} + 7\tilde{f}_{i,1}^{(0)} - \tilde{f}_{i,2}^{(0)}}{12}.$$

- $\kappa_{i+1/2}^2$: The first-order time derivatives of u at the nodes $i - 1, \dots, i + 2$ are approximated by applying the corresponding differentiation numerical formula to $\tilde{f}_{i,j}^{(0)}$:

$$\begin{aligned} \tilde{u}_{i,-1}^{(1)} &= -A_{2,0}^{1,-1}(\tilde{f}_{i,*}^{(0)}, \Delta x) = -\frac{11/6\tilde{f}_{i,-1}^{(0)} - 3\tilde{f}_{i,0}^{(0)} + 3/2\tilde{f}_{i,1}^{(0)} - 1/3\tilde{f}_{i,2}^{(0)}}{\Delta x}, \\ \tilde{u}_{i,0}^{(1)} &= -A_{2,0}^{1,0}(\tilde{f}_{i,*}^{(0)}, \Delta x) = -\frac{1/3\tilde{f}_{i,-1}^{(0)} + 1/2\tilde{f}_{i,0}^{(0)} - \tilde{f}_{i,1}^{(0)} + 1/6\tilde{f}_{i,2}^{(0)}}{\Delta x}, \\ \tilde{u}_{i,1}^{(1)} &= -A_{2,0}^{1,1}(\tilde{f}_{i,*}^{(0)}, \Delta x) = -\frac{-1/6\tilde{f}_{i,-1}^{(0)} + \tilde{f}_{i,0}^{(0)} - 1/2\tilde{f}_{i,1}^{(0)} - 1/3\tilde{f}_{i,2}^{(0)}}{\Delta x}, \\ \tilde{u}_{i,2}^{(1)} &= -A_{2,0}^{1,2}(\tilde{f}_{i,*}^{(0)}, \Delta x) = -\frac{1/3\tilde{f}_{i,-1}^{(0)} - 3/2\tilde{f}_{i,0}^{(0)} + 3\tilde{f}_{i,1}^{(0)} - 11/6\tilde{f}_{i,2}^{(0)}}{\Delta x}. \end{aligned}$$

Next first-order Taylor expansions are used to approximate the values of the flux sixteen space-time local nodes: for $r = -1, \dots, 2$

$$\begin{aligned}
\tilde{f}_{i,-1}^{1,r} &= f(u_{i-1}^n + r\Delta t \tilde{u}_{i,-1}^{(1)}), \\
\tilde{f}_{i,0}^{1,r} &= f(u_{i,0}^n + r\Delta t \tilde{u}_{i,0}^{(1)}), \\
\tilde{f}_{i,1}^{1,r} &= f(u_{i+1}^n + r\Delta t \tilde{u}_{i,+1}^{(1)}), \\
\tilde{f}_{i,2}^{1,r} &= f(u_{i+2}^n + r\Delta t \tilde{u}_{i,+2}^{(1)}).
\end{aligned}$$

Then, the first-order time derivatives of the flux at the nodes $i-1, \dots, i+2$ are approximated by applying the corresponding differentiation numerical formula to $\tilde{f}_{i,j}^{1,r}$:

$$\begin{aligned}
\tilde{f}_{i,-1}^{(1)} &= A_{2,n}^{1,0}(\tilde{f}_{i,-1}^{1,*}, \Delta t) = \frac{-1/3\tilde{f}_{i,-1}^{1,n-1} - 1/2\tilde{f}_{i,-1}^{1,n} + \tilde{f}_{i,-1}^{1,n+1} - 1/6\tilde{f}_{i,-1}^{1,n+2}}{\Delta t}, \\
\tilde{f}_{i,0}^{(1)} &= A_{2,n}^{1,0}(\tilde{f}_{i,0}^{1,*}, \Delta t) = \frac{-1/3\tilde{f}_{i,0}^{1,n-1} - 1/2\tilde{f}_{i,0}^{1,n} + \tilde{f}_{i,0}^{1,n+1} - 1/6\tilde{f}_{i,0}^{1,n+2}}{\Delta t}, \\
\tilde{f}_{i,1}^{(1)} &= A_{2,n}^{1,0}(\tilde{f}_{i,1}^{1,*}, \Delta t) = \frac{-1/3\tilde{f}_{i,1}^{1,n-1} - 1/2\tilde{f}_{i,1}^{1,n} + \tilde{f}_{i,1}^{1,n+1} - 1/6\tilde{f}_{i,1}^{1,n+2}}{\Delta t}, \\
\tilde{f}_{i,2}^{(1)} &= A_{2,n}^{1,0}(\tilde{f}_{i,2}^{1,*}, \Delta t) = \frac{-1/3\tilde{f}_{i,2}^{1,n-1} - 1/2\tilde{f}_{i,2}^{1,n} + \tilde{f}_{i,2}^{1,n+1} - 1/6\tilde{f}_{i,2}^{1,n+2}}{\Delta t}.
\end{aligned}$$

Finally;

$$\kappa_{i+1/2}^2 = A_{2,0}^{0,1/2}(\tilde{f}_{i,*}^{(1)}, \Delta x) = \frac{-\tilde{f}_{i,-1}^{(1)} + 7\tilde{f}_{i,0}^{(1)} + 7\tilde{f}_{i,1}^{(1)} - \tilde{f}_{i,2}^{(1)}}{12}.$$

- $\kappa_{i+1/2}^3$: the second-order time derivatives at the nodes are approximated by

$$\begin{aligned}
\tilde{u}_{i,-1}^{(2)} &= -A_{2,0}^{1,-1}(\tilde{f}_{i,*}^{(1)}, \Delta x) = -\frac{11/6\tilde{f}_{i,-1}^{(1)} - 3\tilde{f}_{i,0}^{(1)} + 3/2\tilde{f}_{i,1}^{(1)} - 1/3\tilde{f}_{i,2}^{(1)}}{\Delta x}, \\
\tilde{u}_{i,0}^{(2)} &= -A_{2,0}^{1,0}(\tilde{f}_{i,*}^{(1)}, \Delta x) = -\frac{1/3\tilde{f}_{i,-1}^{(1)} + 1/2\tilde{f}_{i,0}^{(1)} - \tilde{f}_{i,1}^{(1)} + 1/6\tilde{f}_{i,2}^{(1)}}{\Delta x}, \\
\tilde{u}_{i,1}^{(2)} &= -A_{2,0}^{1,1}(\tilde{f}_{i,*}^{(1)}, \Delta x) = -\frac{-1/6\tilde{f}_{i,-1}^{(1)} + \tilde{f}_{i,0}^{(1)} - 1/2\tilde{f}_{i,1}^{(1)} - 1/3\tilde{f}_{i,2}^{(1)}}{\Delta x}, \\
\tilde{u}_{i,2}^{(2)} &= -A_{2,0}^{1,2}(\tilde{f}_{i,*}^{(1)}, \Delta x) = -\frac{1/3\tilde{f}_{i,-1}^{(1)} - 3/2\tilde{f}_{i,0}^{(1)} + 3\tilde{f}_{i,1}^{(1)} - 11/6\tilde{f}_{i,2}^{(1)}}{\Delta x}.
\end{aligned}$$

Second-order Taylor expansions are used to compute the fluxes at the sixteen nodes in the space-time mesh: for $r = -1, \dots, 2$

$$\begin{aligned}\tilde{f}_{i,-1}^{2,r} &= f \left(u_{i-1}^n + r\Delta t \tilde{u}_{i,-1}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,-1}^{(2)} \right), \\ \tilde{f}_{i,0}^{2,r} &= f \left(u_i^n + r\Delta t \tilde{u}_{i,0}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,0}^{(2)} \right), \\ \tilde{f}_{i,1}^{2,r} &= f \left(u_{i+1}^n + r\Delta t \tilde{u}_{i,1}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,1}^{(2)} \right), \\ \tilde{f}_{i,2}^{2,r} &= f \left(u_{i+2}^n + r\Delta t \tilde{u}_{i,2}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,2}^{(2)} \right).\end{aligned}$$

Next, compute

$$\begin{aligned}\tilde{f}_{i,-1}^{(2)} &= A_{2,n}^{2,0}(\tilde{f}_{i,-1}^{2,*}, \Delta t) = \frac{\tilde{f}_{i,-1}^{2,n-1} - 2\tilde{f}_{i,-1}^{2,n} + \tilde{f}_{i,-1}^{2,n+1}}{\Delta t^2}, \\ \tilde{f}_{i,0}^{(2)} &= A_{2,n}^{2,0}(\tilde{f}_{i,0}^{2,*}, \Delta t) = \frac{\tilde{f}_{i,0}^{2,n-1} - 2\tilde{f}_{i,0}^{2,n} + \tilde{f}_{i,0}^{2,n+1}}{\Delta t^2}, \\ \tilde{f}_{i,1}^{(2)} &= A_{2,n}^{2,0}(\tilde{f}_{i,1}^{2,*}, \Delta t) = \frac{\tilde{f}_{i,1}^{2,n-1} - 2\tilde{f}_{i,1}^{2,n} + \tilde{f}_{i,1}^{2,n+1}}{\Delta t^2}, \\ \tilde{f}_{i,2}^{(2)} &= A_{2,n}^{2,0}(\tilde{f}_{i,2}^{2,*}, \Delta t) = \frac{\tilde{f}_{i,2}^{2,n-1} - 2\tilde{f}_{i,2}^{2,n} + \tilde{f}_{i,2}^{2,n+1}}{\Delta t^2}.\end{aligned}$$

And finally;

$$\kappa_{i+1/2}^3 = A_{2,0}^{0,1/2}(\tilde{f}_{i,*}^{(2)}, \Delta x) = \frac{-\tilde{f}_{i,-1}^{(2)} + 7\tilde{f}_{i,0}^{(2)} + 7\tilde{f}_{i,1}^{(2)} - \tilde{f}_{i,2}^{(2)}}{12}.$$

- $\kappa_{i+1/2}^4$: the third-order time derivatives at the nodes are approximated by

$$\begin{aligned}\tilde{u}_{i,-1}^{(3)} &= -A_{2,0}^{1,-1}(\tilde{f}_{i,*}^{(2)}, \Delta x) = -\frac{11/6\tilde{f}_{i,-1}^{(2)} - 3\tilde{f}_{i,0}^{(2)} + 3/2\tilde{f}_{i,1}^{(2)} - 1/3\tilde{f}_{i,2}^{(2)}}{\Delta x}, \\ \tilde{u}_{i,0}^{(3)} &= -A_{2,0}^{1,0}(\tilde{f}_{i,*}^{(2)}, \Delta x) = -\frac{1/3\tilde{f}_{i,-1}^{(2)} + 1/2\tilde{f}_{i,0}^{(2)} - \tilde{f}_{i,1}^{(2)} + 1/6\tilde{f}_{i,2}^{(2)}}{\Delta x}, \\ \tilde{u}_{i,1}^{(3)} &= -A_{2,0}^{1,1}(\tilde{f}_{i,*}^{(2)}, \Delta x) = -\frac{-1/6\tilde{f}_{i,-1}^{(2)} + \tilde{f}_{i,0}^{(2)} - 1/2\tilde{f}_{i,1}^{(2)} - 1/3\tilde{f}_{i,2}^{(2)}}{\Delta x},\end{aligned}$$

$$\tilde{u}_{i,2}^{(3)} = -A_{2,0}^{1,2}(\tilde{f}_{i,*}^{(2)}, \Delta x) = -\frac{1/3\tilde{f}_{i,-1}^{(2)} - 3/2\tilde{f}_{i,0}^{(2)} + 3\tilde{f}_{i,1}^{(2)} - 11/6\tilde{f}_{i,2}^{(2)}}{\Delta x}.$$

Compute the approximations of the fluxes: for $r = -1, \dots, 2$

$$\begin{aligned}\tilde{f}_{i,-1}^{3,r} &= f\left(u_{i-1}^n + r\Delta t \tilde{u}_{i,-1}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,-1}^{(2)} + \frac{r^3\Delta t^3}{6} \tilde{u}_{i,-1}^{(3)}\right), \\ \tilde{f}_{i,0}^{3,r} &= f\left(u_i^n + r\Delta t \tilde{u}_{i,0}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,0}^{(2)} + \frac{r^3\Delta t^3}{6} \tilde{u}_{i,0}^{(3)}\right), \\ \tilde{f}_{i,1}^{3,r} &= f\left(u_{i+1}^n + r\Delta t \tilde{u}_{i,1}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,1}^{(2)} + \frac{r^3\Delta t^3}{6} \tilde{u}_{i,1}^{(3)}\right), \\ \tilde{f}_{i,2}^{3,r} &= f\left(u_{i+2}^n + r\Delta t \tilde{u}_{i,2}^{(1)} + \frac{r^2\Delta t^2}{2} \tilde{u}_{i,2}^{(2)} + \frac{r^3\Delta t^3}{6} \tilde{u}_{i,2}^{(3)}\right).\end{aligned}$$

Next, compute:

$$\begin{aligned}\tilde{f}_{i,-1}^{(3)} &= A_{2,n}^{3,0}(\tilde{f}_{i,-1}^{3,*}, \Delta t) = \frac{-\tilde{f}_{i,-1}^{3,n-1} + 3\tilde{f}_{i,-1}^{3,n} - 3\tilde{f}_{i,-1}^{3,n+1} + \tilde{f}_{i,-1}^{3,n+2}}{\Delta t^3}, \\ \tilde{f}_{i,0}^{(3)} &= A_{2,n}^{3,0}(\tilde{f}_{i,0}^{3,*}, \Delta t) = \frac{-\tilde{f}_{i,0}^{3,n-1} + 3\tilde{f}_{i,0}^{3,n} - 3\tilde{f}_{i,0}^{3,n+1} + \tilde{f}_{i,0}^{3,n+2}}{\Delta t^3}, \\ \tilde{f}_{i,1}^{(3)} &= A_{2,n}^{3,0}(\tilde{f}_{i,1}^{3,*}, \Delta t) = \frac{-\tilde{f}_{i,1}^{3,n-1} + 3\tilde{f}_{i,1}^{3,n} - 3\tilde{f}_{i,1}^{3,n+1} + \tilde{f}_{i,1}^{3,n+2}}{\Delta t^3}, \\ \tilde{f}_{i,2}^{(3)} &= A_{2,n}^{3,0}(\tilde{f}_{i,2}^{3,*}, \Delta t) = \frac{-\tilde{f}_{i,2}^{3,n-1} + 3\tilde{f}_{i,2}^{3,n} - 3\tilde{f}_{i,2}^{3,n+1} + \tilde{f}_{i,2}^{3,n+2}}{\Delta t^3}.\end{aligned}$$

Finally;

$$\kappa_{i+1/2}^4 = A_{2,0}^{0,1/2}(\tilde{f}_{i,*}^{(2)}, \Delta x) = \frac{-\tilde{f}_{i,-1}^{(2)} + 7\tilde{f}_{i,0}^{(2)} + 7\tilde{f}_{i,1}^{(2)} - \tilde{f}_{i,2}^{(2)}}{12}.$$

If $f(u) = au$, then:

$$\begin{aligned}F_{i+1/2}^2 &= \frac{a}{12}(-u_{i-1}^n + 7u_i^n + 7u_{i+1}^n - u_{i+2}^n) + \frac{a^2\Delta t}{24\Delta x}(-u_{i-1}^n + 15u_i^n - 15u_{i+1}^n + u_{i+2}^n) \\ &\quad + \frac{a^3\Delta t^2}{12\Delta x^2}(u_{i-1}^n - u_i^n - u_{i+1}^n + u_{i+2}^n) + \frac{a^4\Delta t^3}{24\Delta x^3}(u_{i-1}^n - 3u_i^n + 3u_{i+1}^n - u_{i+2}^n),\end{aligned}$$

which coincides with the numerical flux of the fourth-order linear Lax-Wendroff in conservative form.

3.3 Shock-capturing techniques

Although the Compact Approximate Taylor methods are linearly stable in the L^2 sense under the usual CFL condition, they may produce strong oscillations close to a discontinuity of the solution. The goal of this section is to modify the numerical method to avoid these oscillations. Since CAT2 can be considered a generalization of the second-order Lax-Wendroff method, we will apply it the well known technique of flux limiters methods. For CAT2 p ($p \geq 2$), WENO reconstructions will be applied as is done in [10].

3.3.1 Flux limiter - CAT methods

Let us consider the numerical method (3.1.28) with

$$F_{i+1/2} = (1 - \varphi_{i+1/2})F_{i+1/2}^L + \varphi_{i+1/2}F_{i+1/2}^2, \quad (3.3.1)$$

where $F_{i+1/2}^L$ is a first-order robust numerical flux, $F_{i+1/2}^2$ is given by (3.2.14), and $\varphi_{i+1/2}$ is a TVD centered flux limiter function, see [3], [58], [59]. In addition, we consider here

$$\varphi_{i+1/2} = \varphi(r_{i+1/2}), \quad (3.3.2)$$

where φ is the van Albada second version flux limiter:

$$\varphi(r) = \max\left(0, \frac{2r}{1+r^2}\right), \quad (3.3.3)$$

and

$$r_{i+1/2} = \frac{\Delta upw}{\Delta loc} = \begin{cases} \frac{u_i^n - u_{i-1}^n}{u_{i+1}^n - u_i^n} & \text{if } a_{i+1/2} > 0, \\ \frac{u_{i+2}^n - u_{i+1}^n}{u_{i+1}^n - u_i^n} & \text{if } a_{i+1/2} < 0, \end{cases}$$

where $a_{i+1/2}$ is an estimate of the wave speed, for instance the one corresponding to Roe's method:

$$a_{i+1/2} = \begin{cases} \frac{f(u_{i+1}^n) - f(u_i^n)}{u_{i+1}^n - u_i^n} & \text{if } u_i^n \neq u_{i+1}^n; \\ f'(u_i^n) & \text{otherwise.} \end{cases}$$

3.3.2 WENO-CAT methods

Following [11] we use WENO reconstructions of the flux to stabilize the method. The only differences with the algorithm described in Section (3.2.2) are the computation of $\tilde{u}_{i,j}^{(1)}$, that is now performed as follows:

$$\tilde{u}_{i,j}^{(1)} = -\frac{\hat{f}_{i+j+1/2} - \hat{f}_{i+j-1/2}}{\Delta x},$$

where $\hat{f}_{i+1/2}$ denotes the WENO flux splitting, reconstructions at $x_{i+1/2}$ of the flux function described in [60]. The expression of the numerical flux is then given by:

$$F_{i+1/2}^p = \hat{f}_{i+1/2} + \sum_{k=2}^{2p} \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x). \quad (3.3.4)$$

3.3.3 Systems of conservation laws

For systems of conservation laws

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \quad (3.3.5)$$

where $\mathbf{u} = [u_1, \dots, u_M]^T$, $\mathbf{f}(\mathbf{u}) = [f_1(u_1, \dots, u_M), \dots, f_M(u_1, \dots, u_M)]^T$, the expression of CAT methods is given again by ((3.2.9)) and ((3.2.10)) just using bold characters for vectors.

Concerning the shock-capturing techniques:

- The implementation of the flux-limiter technique for systems of M conserved variables, is done by computing (3.3.2) for every component u_k , with $k = 1, \dots, M$, as follows:

First compute:

$$r_{k,i+1/2}^L = \frac{u_{k,i}^n - u_{k,i-1}^n}{u_{k,i+1}^n - u_{k,i}^n}, \quad r_{k,i+1/2}^R = \frac{u_{k,i+2}^n - u_{k,i+1}^n}{u_{k,i+1}^n - u_{k,i}^n},$$

next, compute

$$\varphi_k(r) = \min\{\varphi(r_k^L), \varphi(r_k^R)\}, \quad (3.3.6)$$

where $\varphi_k(r)$ is the flux limiter for the u_k component. See [3] for more details.

Finally, apply $\varphi_k(r)$ for each conserved variables by separate do not insure the stability of solutions, a right procedure is the decomposition of the correction terms of the flux limiter as wave limiters (see [34], [59]) which is not our interest case for now. Instead of this, we use only one value per node i.e.

$$\varphi_{i+1/2} = \min_k\{\varphi_k(r)\}. \quad (3.3.7)$$

- WENO reconstructions are computed in conserved variables using the procedure described in [60].

3.4 Numerical Experiments

The following numerical methods

- CAT $2p$: Compact Approximate Taylor method of order $2p$ (space and time);
- FL-CAT2: Compact Approximate Taylor method of order 2 with flux limiter technique. The first-order methods considered are Lax-Friedrich for scalar problems and HLL for systems;
- WENO s -CAT $2p$: Compact Approximate Taylor method of order $2p$ with WENO reconstructions of order $s = 2p + 1$ to compute $\tilde{u}_{t,i}^{(1)}$;
- WENO s -RK3: WENO method of order $s = 2p + 1$ for the space discretization and TVD-RK3 for the time discretization, see [45];
- WENO s -LAT q : Approximate Taylor method of order $q = 2p + 1$ with WENO reconstructions of order $s = 2p + 1$ to compute $\tilde{u}_{t,i}^{(1)}$, see [10];

will be applied to different 1D scalar conservation laws and systems: transport and Burgers equations, Euler and the ideal Magnetohydrodynamics equations (MHD). All cases time t is in seconds.

3.4.1 1D scalar equations

3.4.1.1 Test 3.1 Transport equation - Discontinuous solutions 1

We consider first (3.1.1) with $a = 1$, in the spatial interval $[0, 1]$, with initial condition

$$u(x, 0) = \begin{cases} 1 & 0 \leq x < 7/10, \\ 2 & 2/10 \leq x < 7/10, \\ 1 & 7/10 \leq x < 1, \end{cases} \quad (3.4.1)$$

periodic boundary conditions, a uniform mesh with $N = 80$ points and $t = 1$ is considered. The CAT method (that, in this case, coincides with the Lax-Wendroff method) is applied for $p = 1, \dots, 5$.

Numerical simulations are shown in Figure 3.2: the L^2 stability of the scheme and the appearance of oscillations near the discontinuities can be observed.

3.4.1.2 Test 3.2 Transport equation - Discontinuous solutions 2

We apply to the previous problem the CAT4, FL-CAT2, WENO5-CAT4, WENO5-RK3, and WENO5-LAT5 methods. A general view is shown in Figure 3.3 together with an enlarged view of the area of interest. As it can be observed, the results given by WENO5-CAT4, WENO5-RK3 and WENO5-LAT5 are almost identical. Nevertheless, as it will be

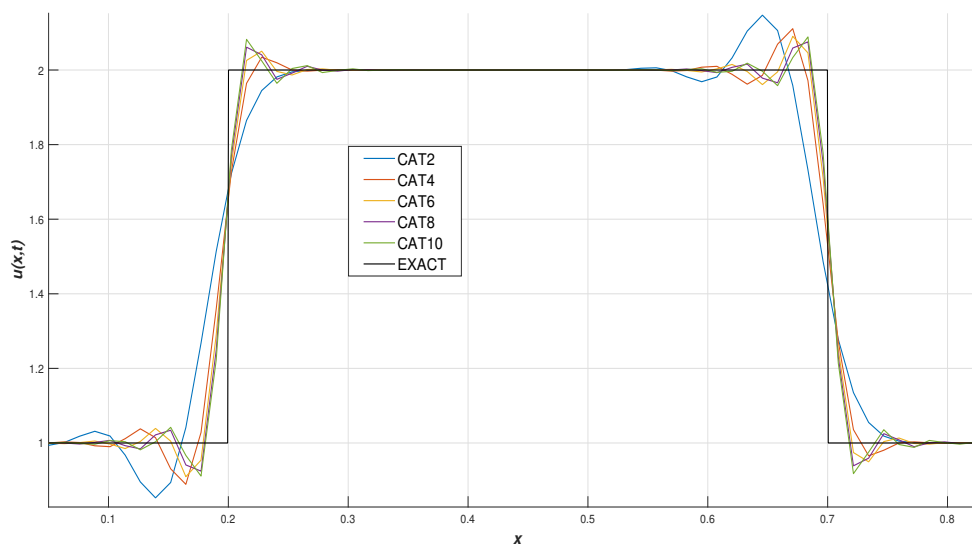


Figure 3.2: Test 3.1. Transport equation with initial condition ((3.4.1)), CFL= 0.9 and $t = 1$. Solutions using CAT $2p$ methods with $p = 1, 2, 3, 4, 5$.

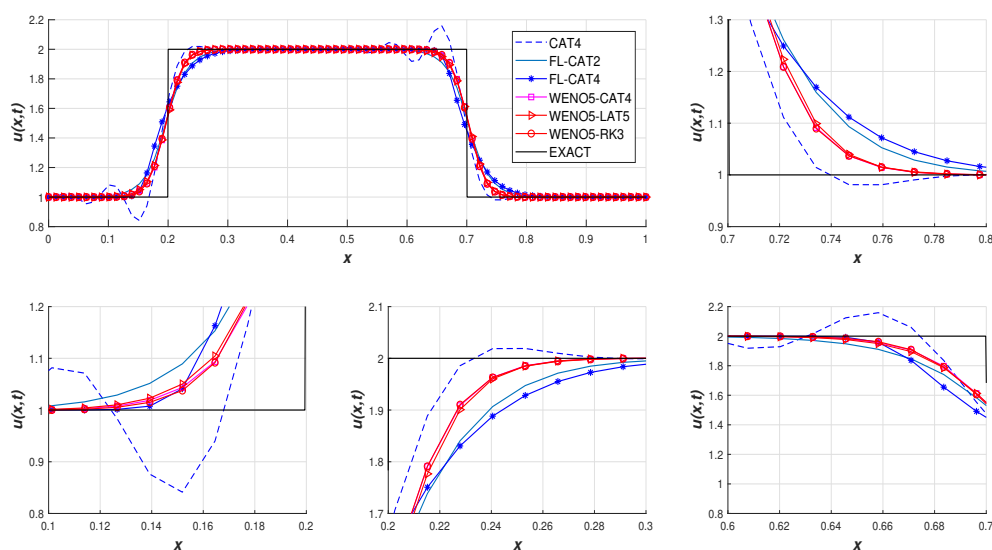


Figure 3.3: Test 3.2. Transport equation with initial condition (3.4.1), CFL= 0.5 and $t = 1$. Left-top: general view. a, b, c and d : enlarged view of interest areas.

seen in the next test problem, WENO5-CAT4 still gives good results for CFL close to

one, what is not the case for WENO5-RK3 or WENO5-LAT5.

Increasing the order of CAT methods implies a significant increase of flops (number of operations required), that should be considered, see Table 3.2.

Order	CAT2	CAT4	CAT6	CAT8	CAT10
Rate flops	1	1.61	2.51	3.69	5.16

Table 3.2: Average rate flops to increase from CAT2 to CAT2p, for $p = 2, 3, 4, 5$ using the scalar transport equation with initial conditions (3.4.1).

3.4.1.3 Test 3.3 Transport equation - Accuracy order

We consider (3.1.1) in the spatial interval $[0, 2]$ with initial condition,

$$u(x, 0) = 0.25 \sin(\pi x), \quad (3.4.2)$$

and periodic boundary conditions. Table 3.3 shows the error and the empirical order for CAT2, CAT4, CAT6, and Table 3.4 for WENO5-RK3 and WENO5-LAT5 which coincides in all cases with the theoretical one. For smooth solutions WENO-CAT2p reduce to the corresponding CAT2p, so that the accuracy test is not necessary.

Remark: in order to achieve fifth order accuracy in time for WENO5-RK3 we set $\Delta t = h^{5/3}$.

Δx	CAT2		CAT4		CAT6	
	Error $\ \cdot \ _1$	Order	Error $\ \cdot \ _1$	Order	Error $\ \cdot \ _1$	Order
0.1053	3.68e-02		1.40e-02		7.88e-03	
0.0526	6.84e-03	2.43	3.50e-05	8.64	4.25e-08	7.50
0.0263	1.70e-03	2.00	2.19e-06	4.00	6.49e-10	6.03
0.0132	4.27e-04	2.00	1.36e-07	4.00	9.89e-12	6.04
0.0066	1.06e-04	2.00	8.55e-09	4.00	1.53e-13	6.01
0.0033	2.66e-05	2.00	5.34e-10	4.00	2.64e-15	5.96

Table 3.3: Test 3.3. Transport equation with initial condition (3.4.2), CFL= 0.5 and $t = 1$: L^1 errors and accuracy order for CAT2p, $p = 1, 2, 3$.

3.4.1.4 Test 3.4 Burgers equation - Discontinuous solutions 1

Let us consider (3.2.1) with

$$f(u) = \frac{u^2}{2}.$$

Δx	WENO5-RK3		WENO-LAT5	
	Error $\ \cdot \ _1$	Order $\ \cdot \ _1$	Error $\ \cdot \ _1$	Order $\ \cdot \ _1$
0.1053	2.03e-03		5.44e-05	
0.0526	6.06e-05	5.06	1.65e-06	5.04
0.0263	1.87e-06	5.02	5.04e-08	5.04
0.0132	5.83e-08	5.00	1.51e-09	5.05
0.0066	1.82e-09	5.00	4.41e-11	5.10
0.0033	5.65e-11	5.01	1.15e-12	5.25

Table 3.4: Test 3.3. Transport equation with initial condition (3.4.2), CFL= 0.5 and $t = 1$: L^1 errors and accuracy order for WENO5-RK3 and WENO5-LAT5.

When CAT methods are applied to approximate a discontinuous solution of this nonlinear problem, the oscillations appearing close to the shocks tend to grow and to spoil the numerical solution. Nevertheless, it is still possible to apply these methods by reducing the CFL parameter (the reduction increases with p): for instance, Figure 3.4 shows the results obtained with CAT_{2p} , $p = 1, 2, 3, 4$ and CFL= 0.8, 0.4, 0.2, 0.1, respectively, with initial conditions (3.4.1), periodic boundary conditions, a grid of 80-point mesh and $t = 1.2s$.

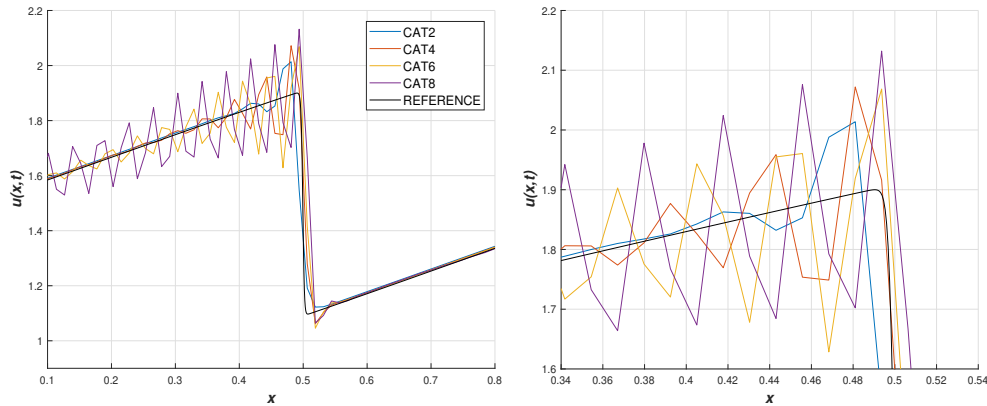


Figure 3.4: Test 3.4. Burgers equation with initial condition (3.4.1), CFL= 0.8, 0.4, 0.2, 0.1, 0.05 and $t = 1.2$. Solutions for CAT_{2p} , $p = 1, 2, 3, 4$. Left: general view. Right: enlarged view.

3.4.1.5 Test 3.5 Burgers equation - Discontinuous solutions 2

The previous test problem is solved using CAT_4 , FL- CAT_2 , WENO5-RK3 and WENO5-LAT5 methods. Using CFL= 0.5 and $t = 2$, we obtain numerical solutions without

spurious oscillations for all the methods (except for CAT4). Figure 3.5 shows a general view of solutions and the van Albada flux limiter function on every inter cell used for FL-CAT2. In order to show solutions for CFL close to 1, the same test is solved using $N = 250$, $\text{CFL} = \{0.5, 0.9\}$ and $t = \{1.2, 12\}$. From Figure 3.6 we can conclude:

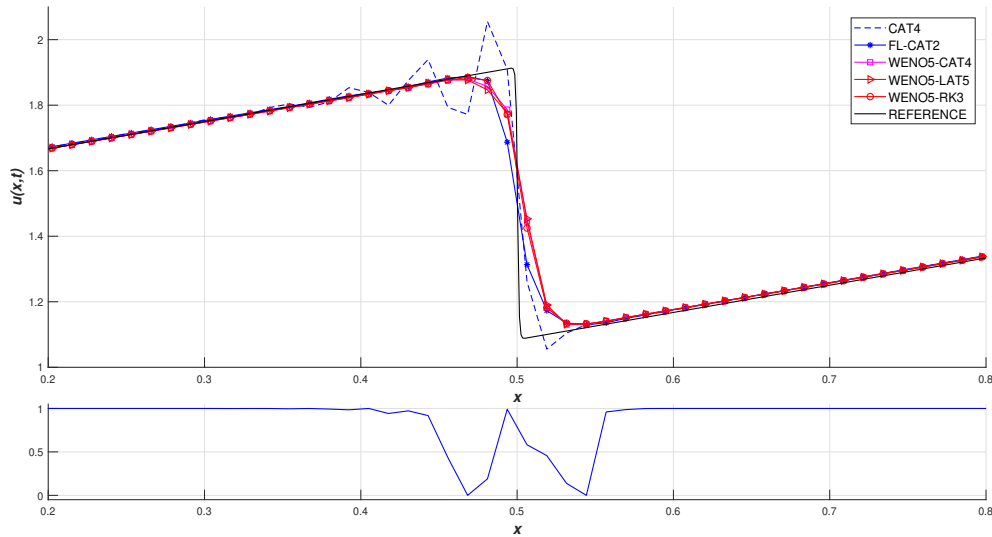


Figure 3.5: Test 3.5. Burgers equation with initial condition (3.4.1), $\text{CFL} = 0.5$ and $t = 1.2$. Top: general view. Bottom: flux limiter function $\varphi_{i+1/2}$ for FL-CAT2.

- $\text{CFL} \leq 0.5$
 - CAT4 shows oscillations near the discontinuities, but it is stable.
 - FL-CAT2 is very diffusive near to the discontinuities, due to the selected first-order accurate flux limiter function.
 - WENO5-CAT4, WENO5-LAT5 and WENO5-RK3 show good results, stable and essentially the same values.
- $\text{CFL} > 0.5$
 - CAT4: the amplitude of oscillations increases near the discontinuities. However, they remain stable.
 - FL-CAT2: conversely to the previous CFL condition, it shows acceptable solutions near the discontinuities.

- WENO5-CAT4 ,WENO5-LAT5 and WENO5-RK3 : slight oscillations appear near the discontinuities at the beginning of the simulations. Nevertheless, as the time increases, these oscillations tend to diminish and the result remains acceptable and stable for WENO5-CAT4, while the solutions given by WENO5-LAT5 is very diffusive and the one given by WENO5-RK3 is overdamped.

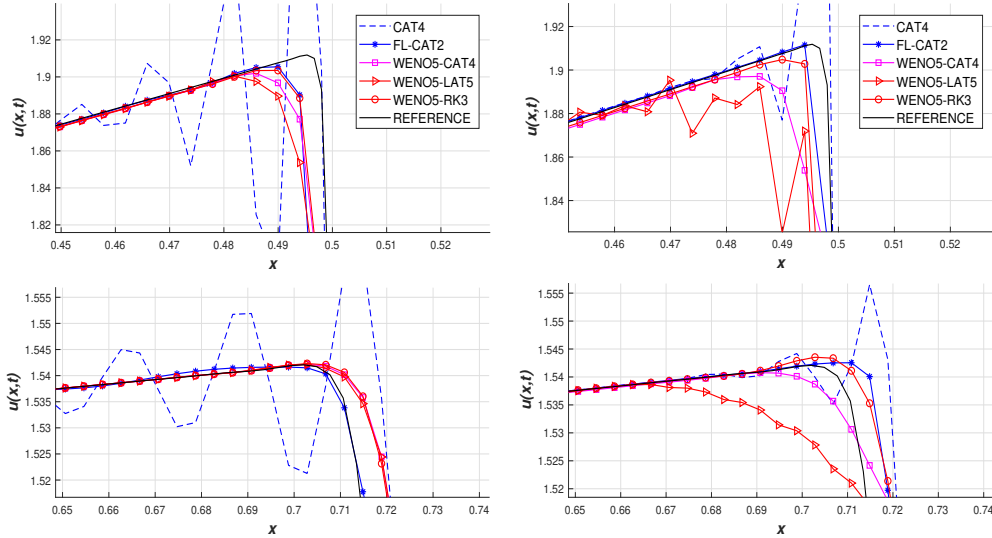


Figure 3.6: Test 3.5. Burgers equation with initial condition (3.4.1), CFL= 0.5 and CFL= 0.9, $t = 1.2$ and $t = 12$: enlarged view of the numerical results. Left-top: CFL= 0.5 and $t = 1.2$. Left-bottom: CFL= 0.5 and $t = 12$. Right-top: CFL= 0.9 and $t = 1.2$. Right-bottom: CFL= 0.9 and $t = 12$.

Although FL-CAT2 shows better results for bigger CFL, it fails in smooth regions close to critical points and for systems (as it will be seen in Euler equations).

3.4.1.6 Test 3.6 Burgers equation - Order of accuracy

We consider again initial condition (3.4.2) and periodic boundary conditions. A reference solution at time $t = 0.5$ (when the solution is still smooth) is obtained with WENO5-RK3 using a fine grid of 1400 nodes. The errors and the empirical order are shown in Table 3.5: the numerical results verify the theoretical analysis.

3.4.2 1D Euler equations

We solve the 1D Euler equations for gas dynamics

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \quad (3.4.3)$$

Δx	CAT2		CAT4		CAT6	
	Error $\ \cdot \ _1$	Order	Error $\ \cdot \ _1$	Order	Error $\ \cdot \ _1$	Order
0.1053	7.94e-03		9.01e-04		2.09e-04	
0.0526	2.08e-03	1.93	6.13e-05	3.88	4.27e-06	5.62
0.0263	5.22e-04	1.99	3.89e-06	3.98	7.49e-08	5.83
0.0132	1.29e-04	2.01	2.44e-07	4.00	1.20e-09	5.96
0.0066	3.08e-05	2.00	1.51e-08	4.00	1.87e-11	6.00
0.0033	6.16e-06	2.00	8.76e-10	4.00	2.84e-13	6.00

Table 3.5: Test 3.6. Burgers equation with initial condition (3.4.2), CFL= 0.5 and $t = 0.5$: L^1 errors and accuracy order for CAT2p, $p = 1, 2, 3$.

with

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ u(E + p) \end{bmatrix}, \quad (3.4.4)$$

where ρ is the density, u the velocity, E the total energy per unit volume, and p the pressure. We assume an ideal gas with the equation of state,

$$p(\rho, e) = (\gamma - 1)\rho e, \quad (3.4.5)$$

being γ the ratio of specific heat capacities of the gas taken as 1.4 and e is the internal energy per unit mass given by:

$$E = \rho(e + 0.5u^2). \quad (3.4.6)$$

3.4.2.1 Test 3.7 Order of accuracy

We consider the spatial interval $[0, 2]$ with the initial condition:

$$\begin{aligned} \rho(x, 0) &= 0.75 + 0.5 \sin(\pi x), \\ \rho u(x, 0) &= 0.25 + 0.5 \sin(\pi x), \\ E(x, 0) &= 0.75 + 0.5 \sin(\pi x), \end{aligned} \quad (3.4.7)$$

and periodic boundary conditions. For this test we take CFL= 0.5 and $t = 0.5$. We use a fine grid with 1400-point mesh to compute CAT8 as a reference solution. The results in Table 3.6 support the theoretically obtained accuracy.

Δx	CAT2		CAT4		CAT6	
	Error $\ \cdot \ _1$	Order	Error $\ \cdot \ _1$	Order	Error $\ \cdot \ _1$	Order
0.1053	3.34e-03		8.57e-04		5.49e-04	
0.0526	8.82e-03	1.92	9.93e-05	3.11	3.53e-05	4.96
0.0263	2.28e-04	1.95	7.31e-06	3.76	1.01e-06	5.12
0.0132	5.69e-05	2.01	4.81e-07	3.93	1.94e-08	5.71
0.0066	1.35e-05	2.07	3.02e-08	3.99	3.21e-10	5.92
0.0033	2.71e-06	2.30	1.78e-09	4.08	4.99e-12	6.01

Table 3.6: Test 3.7. 1D Euler equations with initial condition (3.4.7), CFL= 0.5 and $t = 0.5$: L^1 errors and order of accuracy for CAT2p, $p = 1, 2, 3$.

3.4.2.2 Test 3.8 Sod shock tube problem

$$(\rho, u, p) = \begin{cases} (1, 0, 1) & \text{if } x < 1/2, \\ (0.125, 0, 0.1) & \text{if } x > 1/2. \end{cases}$$

Here, $x \in [0, 1]$, CFL= 0.5, $t = 0.25$, and outflow- boundary conditions are considered at both sides. For details of this problem see [61]. We compare FL-CAT2, WENO5-CAT4, WENO5-LAT5 and WENO5-RK3 using 450 points. A reference solution is computed with the algorithm HE-E1RPEXACT by Toro, see [3].

While all numerical solutions show stable and similar values over smooth regions (see Figure 3.7), the quality is different in the interest regions (a, b, c, d) : an enlarged view of them can be seen in Figure 3.8. By using CFL= 0.5 we observe that the solution given by FL-CAT2 is the most diffusive one, meanwhile, WENO5-CAT4, WENO5-LAT5 and WENO5-RK3 plots essentially the same results. Choosing the CFL= 0.9, we find notorious differences in the solutions, mostly in the approximate Taylor solutions. WENO-CAT4 and WENO-RK3 remains similar solutions to those obtained with CFL= 0.5, which is not the case for WENO-LAT5, see Figure 3.9.

3.4.2.3 Test 3.9 Shu-Osher problem

$$(\rho, u, p) = \begin{cases} (3.8571, 2.6293, 10.3333) & \text{if } x < -4, \\ (1 + 0.2 \sin(5x), 0, 1) & \text{if } x > -4. \end{cases}$$

We consider the spatial interval $x \in [-5, 5]$, CFL= 0.5 and time $t = 1$. For details see [20] test 8. We compare FL-CAT2, WENO5-CAT4, WENO5-LAT5 and WENO5-RK3 using 450- point mesh and a reference solution computed with WENO5-RK3 method using a 2500-point mesh. For this test, all solutions are closely similar and near to the reference solution with the exception of FL-CAT2.

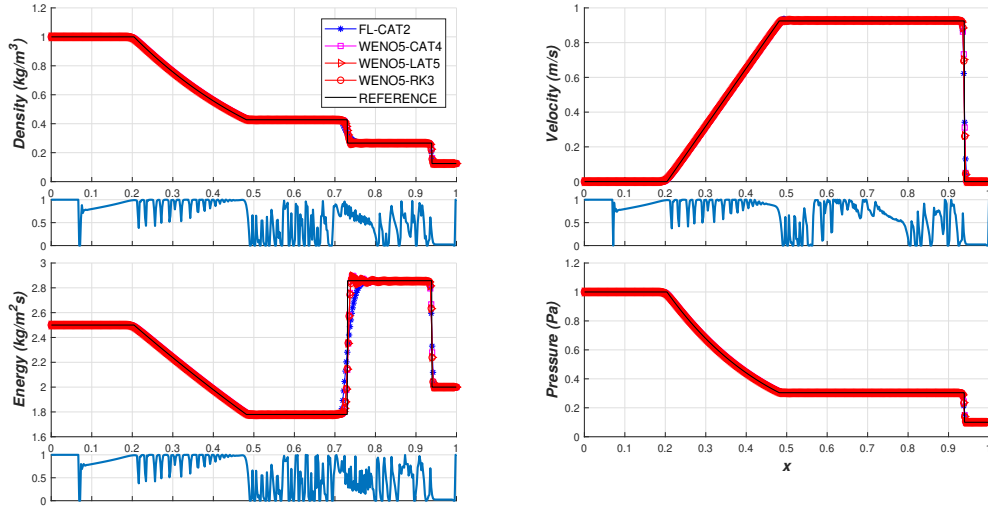


Figure 3.7: Test 3.8. The Sod shock tube problem, CFL= 0.5 and $t = 0.25$. Left-top: general view of numerical solutions for density ρ and φ_{i+2}^ρ . Left-bottom: general view of numerical solutions for the internal energy and φ_{i+2}^E . Right-top: general view of numerical solutions for velocity u and $\varphi_{i+2}^{\rho u}$ FL-CAT2. Right-down: general view of numerical solutions for the pressure p .

3.4.3 1D MHD equations

Finally we consider the 1D ideal Magnetohydrodynamics (MHD) system of equations whose expression is the following:

$$\mathbf{u}_t + \mathbf{f}(\mathbf{u})_x = 0, \quad (3.4.8)$$

with

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho v_z \\ B_x \\ B_y \\ B_z \\ E \end{bmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{bmatrix} \rho v_x \\ \rho v_x^2 + p^* - B_x^2 \\ \rho v_x v_y - B_x B_y \\ \rho v_x v_z - B_x B_z \\ 0 \\ v_x B_y - v_y B_x \\ v_x B_z - v_z B_x \\ v_x (E + p^*) - B_x (\mathbf{v} \cdot \mathbf{B}) \end{bmatrix}, \quad (3.4.9)$$

where ρ is the mass density, $\mathbf{v} = [v_x, v_y, v_z]^T$ and $\mathbf{B} = [B_x, B_y, B_z]^T$ are the velocity and magnetic fields respectively, E is the total energy per unit volume, and p^* the total

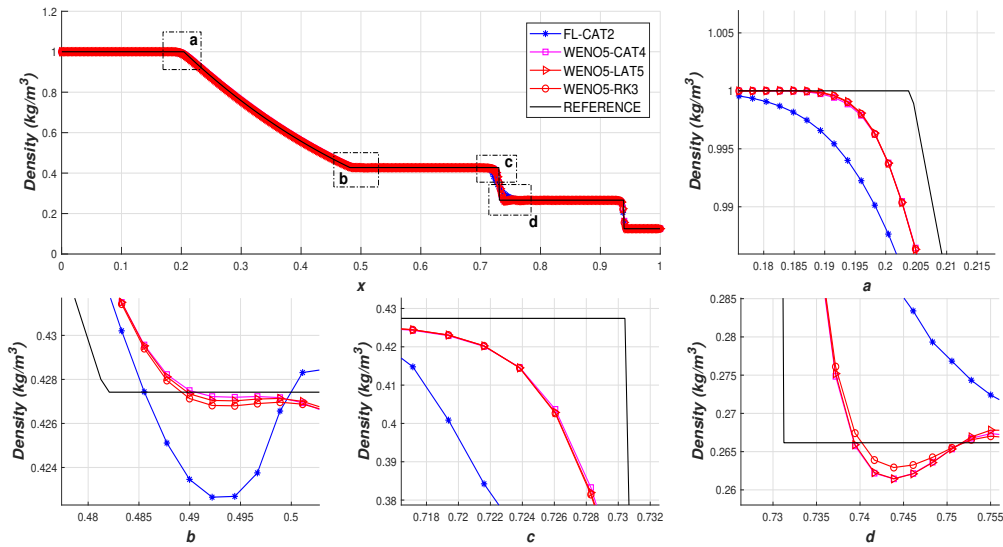


Figure 3.8: Test 3.8. The Sod shock tube problem, CFL= 0.5 and $t = 0.25$. General view and enlarge view of the numerical results for ρ close to regions *a, b, c, d*.

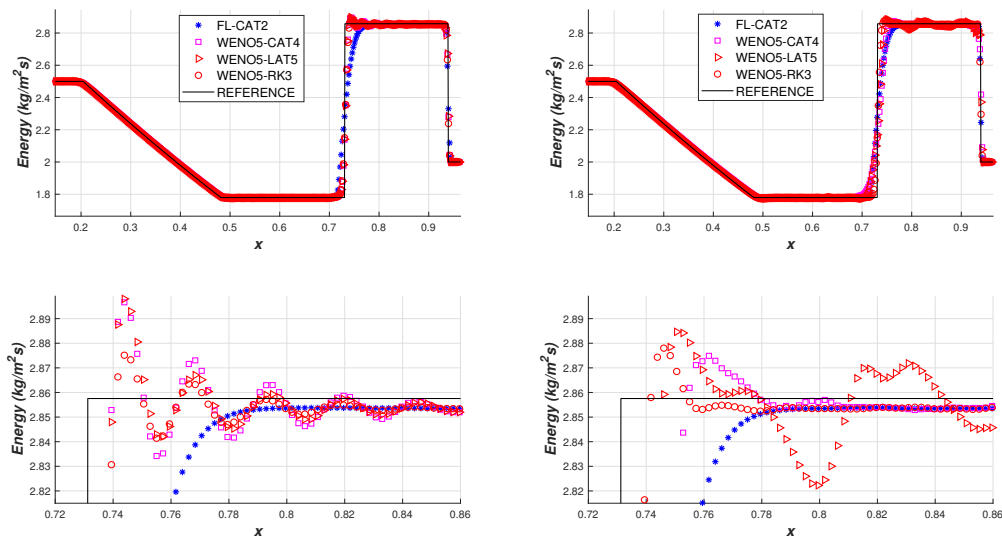


Figure 3.9: Test 3.8. The Sod shock tube problem, CFL= 0.5 and $t = 0.25$. General view and enlarge view of the numerical solutions for internal energy e close to regions *a, b, c, d*.

pressure. We assume an ideal gas with the equation of state

$$p(\rho, e) = (\gamma - 1)\rho e, \tag{3.4.10}$$

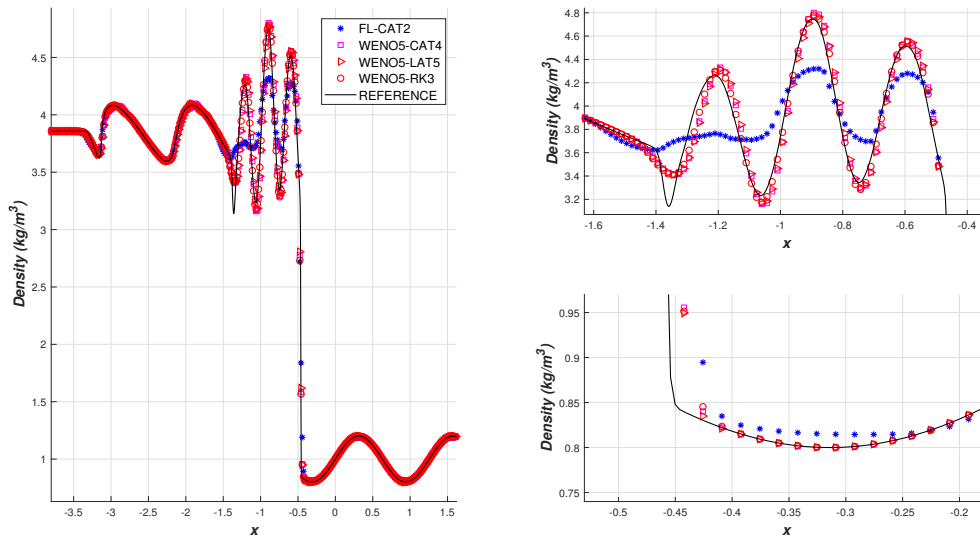


Figure 3.10: Test 3.9. The Shu-Osher problem, CFL= 0.5 and $t = 1$. Left: general view of numerical solutions for density. Right-top: enlarged view. Right-bottom: enlarged view.

where p is the hydrostatic pressure; γ , the adiabatic constant; and e , the internal energy per unit mass related to the total energy by the equation

$$E = \frac{1}{2}\rho|\mathbf{v}|^2 + \frac{1}{2}|\mathbf{B}|^2 + \rho e. \quad (3.4.11)$$

Finally, the total pressure p^* is given by $p + p_m$, where

$$p_m = \frac{1}{2}|\mathbf{B}|^2$$

is the magnetic pressure. The spectral structure of (3.4.9) has been analyzed in [62].

Following [62], we consider two tests for the MHD equations involving discontinuous weak solutions.

3.4.3.1 Test 3.10 Brio-Wu shock tube problem

$$(\rho, \mathbf{v}, \mathbf{B}, p) = \begin{cases} (1, 0, 0, 0, 0, 0.75, 1, 0, 1) & \text{if } x < 0, \\ (0.125, 0, 0, 0, 0, 0.75, -1, 0, 0.1) & \text{if } x > 0. \end{cases}$$

We consider the spatial interval $[-1, 1]$, a $\gamma = 2$, 800-point mesh, Dirichlet boundary conditions, CFL= 0.8, and time $t = 0.2$. A reference solution is computed using the HLL

method with a 20000-point mesh. The solution of this test presents a compound wave consisting of an intermediate shock followed by a slow rarefaction wave.

Plots of the numerical solutions for ρ , v_x , B_y , and p , using WENO5-CAT4, WENO5-RK3 and WENO5-LAT5 methods are shown in Figure 3.11. From the solutions we can observe that all methods give similar solutions. The numerical solutions given by all of the methods present oscillations that remain bounded: see Figure 3.11. While the numerical results for the density are similar, WENO5-RK3 is more oscillatory for v_x in some areas and WENO5-LAT5 produces non-smooth behaviors near shocks caused by the choice CFL= 0.8: see Figures 3.12 and Figure 3.13.

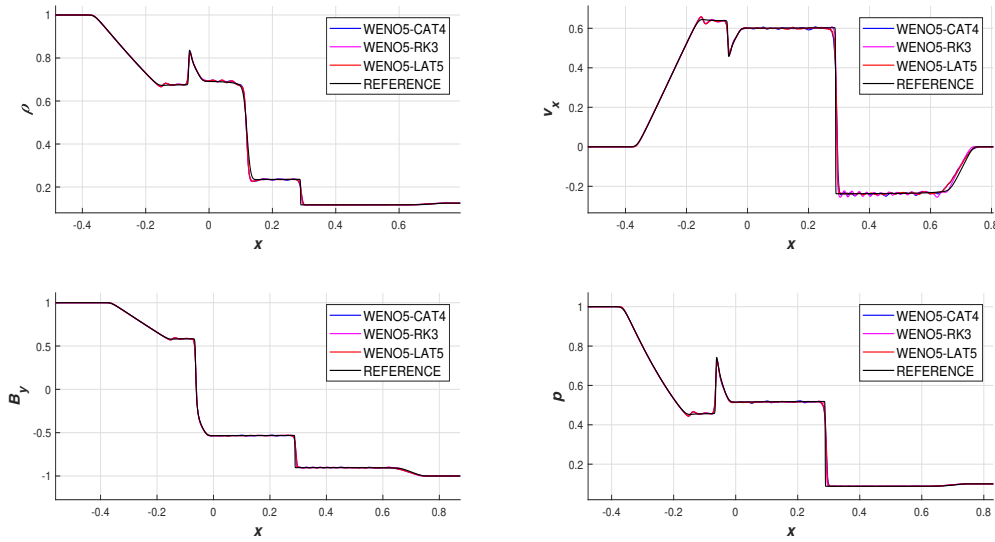


Figure 3.11: Test 3.10. The Brio-Wu shock tube problem, CFL= 0.8 and $t = 0.2$. Numerical solutions for ρ , v_x , B_y , p .

3.4.3.2 Test 3.11 High mach shock tube problem

$$(\rho, \mathbf{v}, \mathbf{B}, p) = \begin{cases} (1, 0, 0, 0, 0, 0, 1, 0, 1000) & \text{if } x < 0, \\ (0.125, 0, 0, 0, 0, 0, -1, 0, 0.1) & \text{if } x > 0. \end{cases}$$

In this case we consider the spatial interval $[-1, 1]$, a 400-point mesh, $\gamma = 2$, Dirichlet boundary conditions, CFL= $\{0.5, 0.8\}$, and time $t = 0.12$. The reference solution is computed as in the previous test. From plots of solutions of ρ , v_x , B_y , p using WENO5-CAT4, WENO5-RK3, and WENO5-LAT5 methods we observe that, with CFL= 0.5, acceptable and stable solutions are obtained for all of the methods: see Figure 3.14. With CFL= 0.8 WENO5-LAT5 is not stable and WENO5-RK3 is more oscillatory than WENO-CAT although discontinuities are captured slightly better: Figure 3.15 shows a

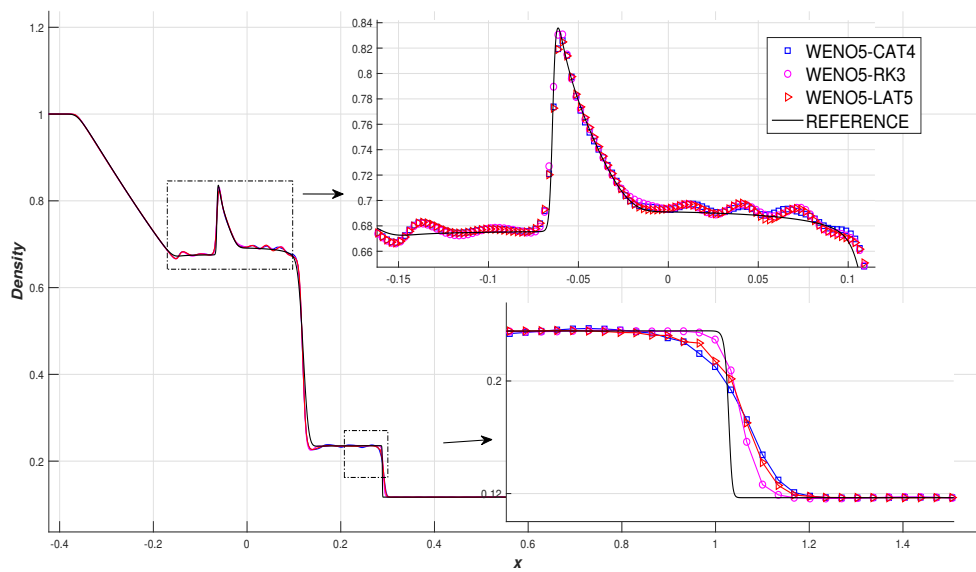


Figure 3.12: Test 3.10. The Brio-Wu shock tube problem, CFL= 0.8 and $t = 0.2$. Enlarged view for ρ .

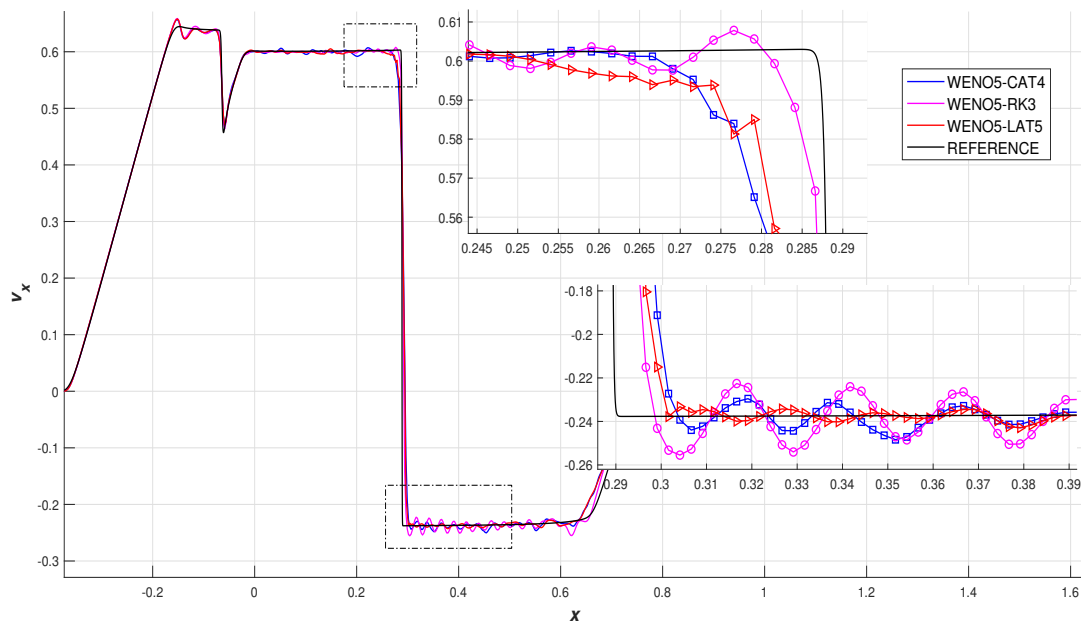


Figure 3.13: Test 3.10. The Brio-Wu shock tube problem, CFL= 0.8 and $t = 0.2$. Enlarged view for v_x .

general view of solutions for ρ , v_x , B_y , p and Figure 3.15 enlarged views of ρ and B_y are shown.

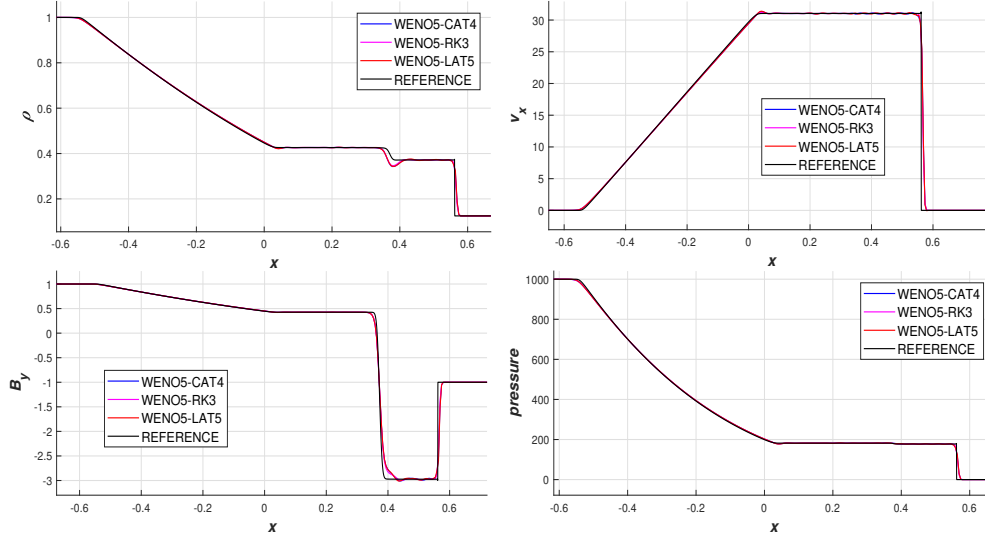


Figure 3.14: Test 3.11. The high mach shock problem, CFL= 0.5 and $t = 0.012$. Numerical solutions for ρ , v_x , B_y , p .

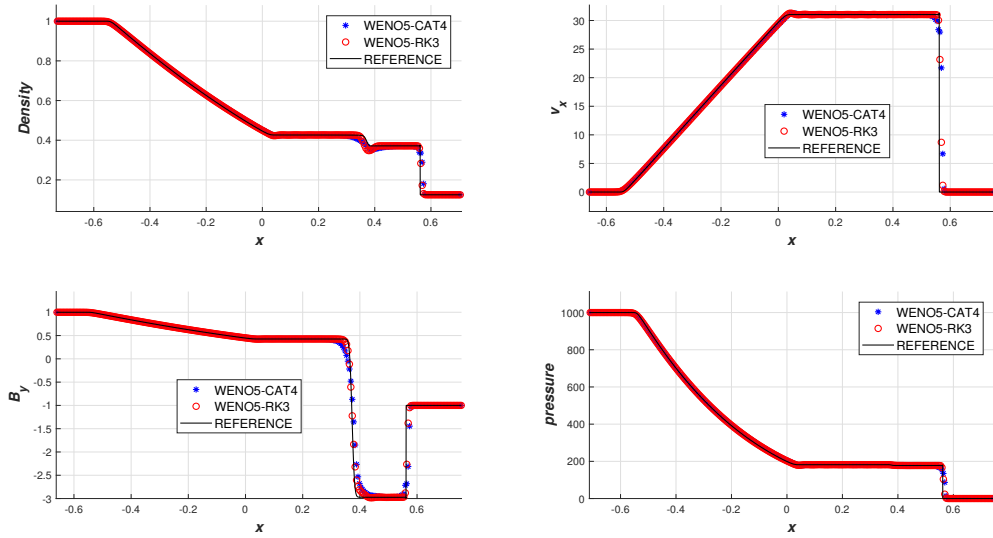


Figure 3.15: Test 3.11. The high mach shock problem, CFL= 0.8 and $t = 0.012$. General view of the numerical solutions provided by WENO5-CAT4 and WENO5-RK3 for ρ , v_x , B_y and p .

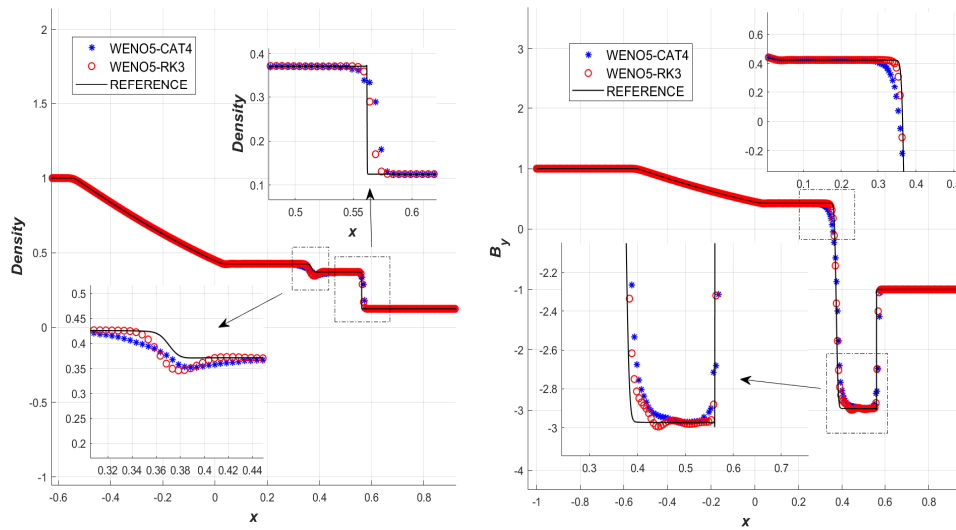


Figure 3.16: Test 3.11. 1D MHD equations with the High mach shock tube problem, CFL= 0.8 and $t = 0.012$. Left: enlarged views for ρ . Right: enlarged views for B_y .

Chapter 4

Adaptive Compact Approximate Taylor Method

The Compact Approximated Taylor methods (CAT) introduced in the previous Chapter circumvent the CK procedure using the same strategy as LAT methods [10]. These methods are compact in the sense that the length of the stencils is minimal: $(2p + 1)$ -point stencils are used to get order $2p$ compared to $(4p + 1)$ -point stencils in LAT methods. The technique used to reduce the length of the stencil makes that the computational cost of a time step in CAT methods is higher than in LAT methods: the Taylor expansions are computed locally, so that the total number of expansions needed to update the numerical solution is multiplied by $(2p + 1)$. On the other hand, unlike LAT methods, CAT methods reduce to the standard high-order Lax-Wendroff methods when applied to linear problems and, due to this, they have better stability properties than LAT and allows one to increase the length of time steps, what compensates the extra cost of every time iteration: see [12].

Both LAT and CAT methods produce oscillations close to the discontinuities of the solution. The use of Weighted Essentially Non-Oscillatory (WENO) reconstructions (see [15], [16]) to compute the first-order time derivatives allows one to prevent these oscillations: this technique has been used in [10] and also in Section 3.3.2. Chapter 5 will focus on the combination of CAT and LAT with different WENO implementations.

A different strategy was also considered in Section 3.3.1 to avoid spurious oscillations close to a discontinuity: to combine CAT2 with a robust first-order numerical method by using a flux-limiter function. The flux-limiter is based on a smoothness indicator, so that the first-order method is used when large gradients are detected and the second-order one is used otherwise. The goal of this chapter is to introduce a new family of shock-capturing high-order numerical methods, called Adaptive Compact Approximation Taylor (ACAT) schemes, which are based on an extension of this strategy: these methods use centered $(2p + 1)$ -point stencils, where p may take values in $\{1, 2, \dots, P\}$ according to a new family of smoothness indicators in the stencils. The methods are based on a combination of a robust first-order scheme and the Compact Approximate Taylor (CAT) methods of order

$2p$ -order, $p = 1, 2, \dots, P$ so that they are first-order accurate near discontinuities and have order $2p$ in smooth regions, where $(2p + 1)$ is the size of the biggest stencil in which data are smooth according to the smoothness indicators.

The advantage of this technique compared to the use of WENO reconstructions is that, in this case, all the combined methods (but the first-order one) are of even order, while WENO methods have odd order of accuracy so that its combination with CAT is not optimal. Moreover, the restriction of the time step imposed by WENO methods may spoil the advantages due to the better stability property of CAT methods.

This chapter is organized as follows. In Section 4.1, we introduce the new family of high-order smoothness indicators. In Section 4.2, first the expression of the ACAT methods for the 1D scalar problems is summarized and then they are extended to 1D systems and 2D problems. Finally in Section 4.3 the results of the numerical experiments for some selected tests, involving 1D and 2D linear and nonlinear systems of conservation laws, are given in order to compare the performance of the ACAT methods with WENO methods.

4.1 Adaptive Compact Approximate Taylor Method

Although Compact Approximate Taylor methods are linearly stable in the L^2 sense under the usual CFL condition, they may produce strong oscillations close to a discontinuity of the solution. Two different techniques were considered in 3, section 3.3 to avoid these oscillations: to combine CAT2 with a first-order robust method using a flux limiter (FL-CAT2 method) or, following [10], to use WENO reconstructions to compute the first-order time derivatives (WENO-CAT methods).

The strategy to be followed here consists on selecting automatically the stencil used to compute $F_{i+1/2}$ so that its length is maximal among those for which the solution is smooth. More specifically, let us suppose that solutions at time $n\Delta t$ $\{u_i^n\}$ have been computed. The maximum length of the stencil to compute $F_{i+1/2}$ is set to, say, $2P$, where P is a natural number. Then, the candidates stencils to compute $F_{i+1/2}$ are

$$S_p = \{x_{i-p+1}, \dots, x_{i+p}\}, \quad p = 1, \dots, P.$$

In order to select the stencil, some smoothness indicators $\psi_{i+1/2}^p$, $p = 1, \dots, P$ are computed such that:

$$\psi_{i+1/2}^p \approx \begin{cases} 1 & \text{if } \{u_i^n\} \text{ is 'smooth' in } S_p, \\ 0 & \text{otherwise.} \end{cases} \quad (4.1.1)$$

Define now:

$$\mathcal{A} = \{p \in \{1, \dots, P\} \text{ s.t. } \psi_{i+1/2}^p \cong 1\}.$$

The idea would be then to define:

$$F_{i+1/2}^A = \begin{cases} F_{i+1/2}^{lo} & \text{if } \mathcal{A} = \emptyset; \\ F_{i+1/2}^{p_s} & \text{where } p_s = \max(\mathcal{A}) \text{ otherwise;} \end{cases}$$

where $F_{i+1/2}^{p_s}$ is the numerical flux of CAT2 p_s and $F_{i+1/2}^{lo}$ is a robust first-order numerical flux. Nevertheless, it is not possible to determine if the solution is smooth or not in the stencil S_1 where only two values u_i^n , u_{i+1}^n are available. Therefore, what will be done in practice is to define:

$$\mathcal{A} = \{p \in \{2, \dots, P\} \text{ s.t. } \psi_{i+1/2}^p \cong 1\}. \quad (4.1.2)$$

and then:

$$F_{i+1/2}^A = \begin{cases} F_{i+1/2}^* & \text{if } \mathcal{A} = \emptyset; \\ F_{i+1/2}^{p_s} & \text{where } p_s = \max(\mathcal{A}) \text{ otherwise;} \end{cases} \quad (4.1.3)$$

where $F_{i+1/2}^*$ is the numerical flux of the FL-CAT2 (that uses the stencil S_2 as well). In what follows, we recall first the expression of the FL-CAT2 numerical flux; next, we introduce the smoothness indicators; then, we summarize the expression of the high-order ACAT methods; and finally we briefly discuss its application to systems of conservation laws.

4.1.1 FL-CAT2 numerical flux

Let us consider the one-dimensional system of conservation laws

$$u_t + f(u)_x = 0, \quad u(x, 0) = u_0(x), \quad -\infty < x < \infty. \quad (4.1.4)$$

with $m = 1$.

The expression of the FL-CAT2 numerical flux is as follows:

$$F_{i+1/2}^* = \psi_{i+1/2}^1 F_{i+1/2}^1 + (1 - \psi_{i+1/2}^1) F_{i+1/2}^{lo}, \quad (4.1.5)$$

where $F_{i+1/2}^1$ is given by

$$F_{i+1/2}^1 = \frac{1}{4}(\tilde{f}_{i,1}^{1,n+1} + \tilde{f}_{i,0}^{1,n+1} + f_{i+1}^n + f_i^n), \quad (4.1.6)$$

where

$$\tilde{f}_{i,j}^{1,n+1} = f \left(u_{i+j}^n - \frac{\Delta t}{\Delta x} (f(u_{i+1}^n) - f(u_i^n)) \right), \quad j = \{0, 1\}. \quad (4.1.7)$$

$F_{i+1/2}^{lo}$ is a first-order robust numerical flux; and $\psi_{i+1/2}^1$ is a standard flux limiter:

$$\psi_{i+1/2}^1 = \psi^1(r_{i+1/2}), \quad (4.1.8)$$

where

$$r_{i+1/2} = \frac{\Delta upw}{\Delta loc} = \begin{cases} r_{i+1/2}^- := \frac{u_i^n - u_{i-1}^n}{u_{i+1}^n - u_i^n} & \text{if } a_{i+1/2} > 0, \\ r_{i+1/2}^+ := \frac{u_{i+2}^n - u_{i+1}^n}{u_{i+1}^n - u_i^n} & \text{if } a_{i+1/2} < 0; \end{cases} \quad (4.1.9)$$

and $a_{i+1/2}$ is an estimate of the wave speed like for instance Roe's intermediate speed:

$$a_{i+1/2} = \begin{cases} \frac{f(u_{i+1}^n) - f(u_i^n)}{u_{i+1}^n - u_i^n} & \text{if } u_i^n \neq u_{i+1}^n; \\ f'(u_i^n) & \text{otherwise.} \end{cases}$$

An alternative that avoids the computation of an intermediate speed was introduced in [3]: it consists in defining

$$\psi_{i+1/2}^1 = \min(\psi^1(r_{i+1/2}^+), \psi^1(r_{i+1/2}^-)). \quad (4.1.10)$$

4.1.2 Smoothness indicators

Let us introduce a new family of local smoothness indicators $\psi_{i+1/2}^p$, $p \geq 2$, for scalar conservation laws and analyze their properties.

Given the nodal approximations f_i of a function f at the stencil S_p , $p \geq 2$, centered at $x_{i+1/2}$, first define the lateral weights:

$$I_{p,L} := \sum_{j=-p+1}^{-1} (f_{i+1+j} - f_{i+j})^2 + \varepsilon, \quad I_{p,R} := \sum_{j=1}^{p-1} (f_{i+1+j} - f_{i+j})^2 + \varepsilon, \quad (4.1.11)$$

where ε is a small quantity that is added to prevent the lateral weights to vanish when the function is constant. Next, compute:

$$I_p := \frac{I_{p,L} I_{p,R}}{I_{p,L} + I_{p,R}}. \quad (4.1.12)$$

Finally, define the smoothness indicator of the stencil of S_p by

$$\psi_{i+1/2}^p := \left(\frac{I_p}{I_p + \tau_p} \right), \quad (4.1.13)$$

where

$$\tau_p := (\Delta_{i-p+1}^{2p-1} f)^2. \quad (4.1.14)$$

Here, $\Delta_{i-p+1}^{2p-1} f$ represents the undivided difference of $\{f_{i-p+1}, \dots, f_{i+p}\}$:

$$\Delta_{i-p+1}^{2p-1} f = (2p-1)! \sum_{j=-p+1}^p \gamma_{p,j}^{2p-1,1/2} f_{i+j}^n. \quad (4.1.15)$$

Before going into technical details, let us give a motivation of this choice. If data in the stencil S_p are smooth, then

$$I_{p,L} = O(\Delta x^2), \quad I_{p,R} = O(\Delta x^2), \quad \tau_p = O(\Delta x^{4p}).$$

Since

$$\frac{1}{I_p} = \frac{1}{I_{p,L}} + \frac{1}{I_{p,R}}$$

then $I_p = O(\Delta x^2)$ and thus

$$\psi_{i+1/2}^p = \frac{I_p}{I_p + \tau_p} = \frac{O(\Delta x^2)}{O(\Delta x^2) + O(\Delta x^{4p})} \approx 1.$$

On the other hand, if there is an isolated discontinuity in the stencil then

$$\tau_p = O(1)$$

and

$$I_{p,L} = O(1), \quad I_{p,R} = O(\Delta x^2)$$

or

$$I_{p,L} = O(\Delta x), \quad I_{p,R} = O(1).$$

In both cases $I_p = O(\Delta x^2)$ and thus:

$$\psi_{i+1/2}^p = \frac{I_p}{I_p + \tau_p} = \frac{O(\Delta x^2)}{O(\Delta x^2) + O(1)} \approx 0.$$

Nevertheless, in the case of smooth data, special care has to be taken if there is a critical point in the stencil, since in this case the order of I_p depends on the order of the critical point, what can prevent the smoothness indicator to be close of 1, as it will be seen in Propositions 4.1.1-4.1.3 below. The following definition is assumed in these results: a point x is said to be a critical point of f of order n if $f^{(j)}(x) = 0$, $j = 1, \dots, n$ and $f^{(n+1)} \neq 0$.

Before analysing the smoothness indicators, let us introduce some definitions and notation, taken from [17]: we refer to Section 2.1 of this reference for further details.

Given $\alpha \in \mathbb{R}^+$ and $f : (0, h^*) \mapsto \mathbb{R}$ with $h^* \in (0, \infty]$, the notation $f(h) = \mathcal{O}(h^\alpha)$ means, as usual, that

$$\limsup_{h \rightarrow 0^+} \left| \frac{f(h)}{h^\alpha} \right| < +\infty,$$

and the notation $f(h) = \bar{\mathcal{O}}(h^\alpha)$ means that

$$\limsup_{h \rightarrow 0^+} \left| \frac{f(h)}{h^\alpha} \right| < +\infty \quad \text{and} \quad \liminf_{h \rightarrow 0^+} \left| \frac{f(h)}{h^\alpha} \right| > 0.$$

If $f, g : (0, h^*) \mapsto \mathbb{R}$ and α, β are two positive real numbers, the following relations hold:

$$\begin{aligned} f(h) = \mathcal{O}(h^\alpha), \quad g(h) = \mathcal{O}(h^\beta) &\implies f(h)g(h) = \mathcal{O}(h^{\alpha+\beta}); \\ f(h) = \bar{\mathcal{O}}(h^\alpha), \quad g(h) = \bar{\mathcal{O}}(h^\beta) &\implies f(h)g(h) = \bar{\mathcal{O}}(h^{\alpha+\beta}); \\ f > 0, f(h) = \bar{\mathcal{O}}(h^\alpha) &\implies f(h)^{-1} = \bar{\mathcal{O}}(h^{1/\alpha}). \end{aligned}$$

Lemma 4.1.1 *Let $c, d, z \in \mathbb{R}$. Assume that*

$$\begin{cases} f^{(j)}(z) = 0 \text{ for } j = 1, \dots, k, & f^{(k+1)}(z) \neq 0, \text{ and } f \in \mathcal{C}^{k+2} & \text{if } c + d \neq 0; \\ f^{(2j-1)}(z) = 0 \text{ for } j = 1, \dots, n, & f^{(2n+1)}(z) \neq 0, \text{ and } f \in \mathcal{C}^{2n+2} & \text{if } c + d = 0. \end{cases}$$

Then

$$f(z + dh) - f(z - dh) = \bar{\mathcal{O}}(h^s),$$

where

$$s = \begin{cases} k + 1 & \text{if } c + d \neq 0; \\ 2n + 1 & \text{if } c + d = 0. \end{cases}$$

From this lemma, whose proof is given in [17], one can deduce that, given the values $f_j = f(x_j)$, $j = i - p + 1, \dots, i + p$ of a smooth enough function f in the stencil S_p , the following estimates hold:

$$f_{j+1} - f_j = \mathcal{O}(h), \quad j = i - p + 1, \dots, i + p - 1$$

if the stencil does not contain any critical point of f ;

$$f_{j+1} - f_j = \bar{\mathcal{O}}(h^{k+1}), \quad j = i - p + 1, \dots, i + p - 1, \quad (4.1.16)$$

if the stencil contains a critical point x^* of even order k or a critical point of odd order that is not located at the center of any sub-interval of the stencil.

Finally, if there exists i_0 such that $x^* = 0.5(x_{i_0} + x_{i_0+1})$ is a critical point of odd order, then (4.1.16) holds for every $j \neq i_0$ and

$$f_{i_0+1} - f_{i_0} = \bar{\mathcal{O}}(h^{2n+1}) \quad (4.1.17)$$

where $2n + 1$ is the first odd number such that

$$f^{(2n+1)}(x^*) \neq 0.$$

Let us analyze the behavior of the smoothness indicators (4.1.13) assuming that $\varepsilon = 0$ (the role of ε is only relevant for the implementation of the method):

Proposition 4.1.1 *Let $f_j = f(x_j)$, $j = i - p + 1, \dots, i + p$ be the values of a function f in the stencil S_p , with $p > 2$. The following estimates hold:*

$$\psi_{i+1/2}^p = \begin{cases} 1 - \mathcal{O}(\Delta x^{4(p-1)-2k}) & \text{if } f \in \mathcal{C}^{\max(2p-1, k+2)}; \\ \bar{\mathcal{O}}(\Delta x^{2(k+1)}) & \text{if } f \text{ is piecewise } \mathcal{C}^{k+2} \text{ and } S_p \text{ contains an isolated jump discontinuity of } f; \end{cases}$$

where $k = 0$ if there is no critical point of f in S_p or k equal to the order of the critical point if there is one.

Proof. If $f \in C^{2p-1}$ there exists ξ such that

$$\Delta_{i-p+1}^{2p-1} f = (2p-1)! f^{(2p-1)}(\xi) \Delta x^{2p-1},$$

and thus

$$\Delta_{i-p+1}^{2p-1} f = \mathcal{O}(\Delta x^{2p-1}),$$

what implies

$$\tau_p = \mathcal{O}(\Delta x^{4p-2}).$$

On the other hand, if S_p contains an isolated jump discontinuity, then

$$\Delta_{i-p+1}^{2p-1} f = \mathcal{O}(1),$$

and thus

$$\tau_p = \bar{\mathcal{O}}(1).$$

From the discussion above, the estimate

$$f_{j+1} - f_j = \bar{\mathcal{O}}(\Delta x^{k+1}),$$

holds for every $j \in i - p + 1, \dots, i + p - 1$ with the exception of at most one index i_0 , in which the order is higher.

Nevertheless, since both $I_{p,L}$ and $I_{p,R}$ are the sum of at least two terms of the form $(f_{j+1} - f_j)^2$, we can conclude that

$$I_{p,L} = \bar{\mathcal{O}}(\Delta x^{2+2k}), \quad I_{p,R} = \bar{\mathcal{O}}(\Delta x^{2+2k}).$$

Hence:

$$I_p = \frac{I_{p,L} I_{p,R}}{I_{p,L} + I_{p,R}} = \frac{\bar{\mathcal{O}}(\Delta x^{2+2k}) \bar{\mathcal{O}}(\Delta x^{2+2k})}{\bar{\mathcal{O}}(\Delta x^{2+2k}) + \bar{\mathcal{O}}(\Delta x^{2+2k})} = \frac{\bar{\mathcal{O}}(\Delta x^{4+4k})}{\bar{\mathcal{O}}(\Delta x^{2+2k})} = \bar{\mathcal{O}}(\Delta x^{2+2k}).$$

Now, if S_p contains a discontinuity, then, by construction, there exists a side $\alpha \in \{L, R\}$ such that $I_{p,\alpha} = \bar{\mathcal{O}}(1)$ (the side that contains the discontinuity) while the other side, $\beta \in \{L, R\} \setminus \{\alpha\}$, satisfies $I_{p,\beta} = \bar{\mathcal{O}}(\Delta x^{2+2k})$. Therefore

$$I_p = \frac{I_{p,L} I_{p,R}}{I_{p,L} + I_{p,R}} = \frac{I_{p,\alpha} I_{p,\beta}}{I_{p,\alpha} + I_{p,\beta}} = \frac{\bar{\mathcal{O}}(1) \bar{\mathcal{O}}(\Delta x^{2+2k})}{\bar{\mathcal{O}}(1) + \bar{\mathcal{O}}(\Delta x^{2+2k})} = \frac{\bar{\mathcal{O}}(\Delta x^{2+2k})}{\bar{\mathcal{O}}(1)} = \bar{\mathcal{O}}(\Delta x^{2+2k}).$$

Combining the above results, we have that, if f is smooth:

$$\psi_{i+1/2}^p = \frac{I_p}{I_p + \tau_p} = \frac{1}{1 + \frac{\tau_p}{I_p}} = \frac{1}{1 + \frac{\mathcal{O}(\Delta x^{4p-2})}{\bar{\mathcal{O}}(\Delta x^{2+2k})}} = \frac{1}{1 + \mathcal{O}(\Delta x^{4(p-1)-2k})} = 1 - \mathcal{O}(\Delta x^{4(p-1)-2k}).$$

On the other hand, if S_p contains a discontinuity, then

$$\psi_{i+1/2}^p = \frac{I_p}{I_p + \tau_p} = \frac{1}{1 + \frac{\tau_p}{I_p}} = \frac{1}{1 + \frac{\bar{\mathcal{O}}(1)}{\bar{\mathcal{O}}(\Delta x^{2+2k})}} = \frac{1}{1 + \bar{\mathcal{O}}(\Delta x^{-2(k+1)})} = \bar{\mathcal{O}}(\Delta x^{2(k+1)}),$$

which finishes the proof. \square

Observe that the indicator $\psi_{i+1/2}^p$ is able to detect smoothness in the presence of a critical point whose order is lower than $2(p-1)$.

In the case $p=2$ similar arguments lead to prove the following estimates:

Proposition 4.1.2 *Let $f_j = f(x_j)$, $j = i-1, \dots, i+2$ be the values of a function f in the stencil S_2 . The following estimates hold:*

$$\psi_{i+1/2}^2 = \begin{cases} 1 - \mathcal{O}(\Delta x^{4-2k}) & \text{if } f \in \mathcal{C}^3; \\ \bar{\mathcal{O}}(\Delta x^{2(k+1)}) & \text{if } f \text{ is piecewise } \mathcal{C}^{k+2} \text{ and } S_p \text{ contains an isolated jump discontinuity of } f; \end{cases}$$

where $k=0$ if there is no critical point of f in S_2 and $k=1$ if there is a critical point x^* of order 1 such that $f^{(3)}(x^*) \neq 0$ or such that $x^* \neq 0.5(x_j + x_{j+1})$ for $j = i-1, i+1$.

Nevertheless, the estimate cannot be proved when S_2 includes a critical point of order 1 located at $0.5(x_{i-1} + x_i)$ or $0.5(x_{i+1} + x_{i+2})$ and such that $f^{(3)}(x^*) \neq 0$: the argument in the proof of Proposition 4.1.1 cannot be used since there is only one term in the definition of the local weights. This is not a limitation in many applications, since this situation is very specific and, even if it happens, unless there is a discontinuity close to the critical point, smoothness will be detected by at least one of the indicators $\psi_{i+1/2}^p$ with $p > 2$ so that the stencil S_p will be used to update the solution. In any case, the smoothness indicator for $p=2$ can be modified to properly handle these situations as follows: compute the couple of lateral weights:

$$I_{2,L}^1 := (f_i - f_{i-1})^2 + \varepsilon, \quad I_{2,R}^1 := (f_{i+1} - f_i)^2 + (f_{i+2} - f_{i+1})^2 + \varepsilon, \quad (4.1.18)$$

$$I_{2,L}^2 := (f_i - f_{i-1})^2 + (f_{i+1} - f_i)^2 + \varepsilon, \quad I_{2,R}^2 := (f_{i+2} - f_{i+1})^2 + \varepsilon. \quad (4.1.19)$$

Next, compute:

$$I_2^j := \frac{I_{2,L}^j I_{2,R}^j}{I_{2,L}^j + I_{2,R}^j}, \quad j = 1, 2. \quad (4.1.20)$$

and then, the smoothness indicator of the stencil S_2 is given by

$$\tilde{\psi}_{i+1/2}^2 := \max\left(\frac{I_2^1}{I_2^1 + \tau_2}, \frac{I_2^2}{I_2^2 + \tau_2}\right). \quad (4.1.21)$$

The following estimate can be then proved:

Proposition 4.1.3 *Let $f_j = f(x_j)$, $j = i - 1, \dots, i + 2$ be the values of a function f in the stencil S_2 . The following estimates hold:*

$$\tilde{\psi}_{i+1/2}^2 = \begin{cases} 1 - \mathcal{O}(\Delta x^{4-2k}) & \text{if } f \in \mathcal{C}^3; \\ \bar{\mathcal{O}}(\Delta x^{2(k+1)}) & \text{if } f \text{ is piecewise } \mathcal{C}^{k+2} \text{ and } S_p \text{ contains an isolated jump discontinuity of } f; \end{cases}$$

where $k = 0$ if there is no critical points of f in S_2 or $k = 1$ if there is a critical point x^* of order 1.

Proof. The arguments of the proof of Proposition (4.1.1) are used again. The difference comes from the case in which there is a critical point of order 1 located at $0.5(x_{i-1} + x_i)$ or $0.5(x_{i+1} + x_{i+2})$ and such that $f^{(3)}(x^*) = 0$. In this case, there exists $j \in \{1, 2\}$ (the one in which the sub-interval with the critical point and the central sub-interval are considered together in the same lateral weight) such that

$$\frac{I_2^j}{I_2^j + \tau_2} = 1 - \mathcal{O}(\Delta x^2).$$

Using this estimate we can conclude the proof as in Proposition (4.1.1)

Let us remark finally that the smoothness indicators (4.1.13) and (4.1.21) have finally the following homothetic invariance property: given a function f and positive numbers α, β , define

$$g(x) = \alpha f(\beta x).$$

Then the smoothness indicator of f at a stencil S_p centered at $x_{i+1/2}$ in a mesh with step Δx is equal to the smoothness indicator of g at the stencil S_p centered at $\beta x_{i+1/2}$ in a mesh with step $\beta \Delta x$. This property is very important in practice to have smoothness indicators whose behaviour do not depend on Δx and scaling factors of f .

4.1.3 ACAT2P methods

The expression of the Adaptive Compact Approximate Taylor Method (ACAT2P) of maximal order $2P$ for a scalar conservation law is given then by:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} (\mathcal{F}_{i-1/2}^A - \mathcal{F}_{i+1/2}^A). \quad (4.1.22)$$

The numerical fluxes $\mathcal{F}_{i+1/2}^A$ are defined by (4.1.2)-(4.1.3) where $F_{i+1/2}^*$ is the numerical flux of the FL-CAT2 (4.1.5) and the smoothness indicators are given by (4.1.8), (4.1.13). For $p = 2$ (4.1.13) can be replaced by (4.1.21).

Observe that, by definition, $\mathcal{F}_{i+1/2}^A$ reduces to:

- a first-order flux if $\psi_{i+1/2}^1 = 0$ and $\psi_{i+1/2}^p = 0$ for all $p = 2, \dots, P$;
- a second-order flux if $\psi_{i+1/2}^1 = 1$ and $\psi_{i+1/2}^p \approx 0$ for all $p = 2, \dots, P$;
- $2p_s$ -order flux if $\psi_{i+1/2}^{p_s} \approx 1$.

Furthermore, if $p_s = P$, then ACAT2P coincides with CAT2P which has $2P$ -order accuracy and is L^2 -stable under $\text{CFL} \leq 1$.

Let us suppose that f is smooth and has an isolated critical point x^* of order k in $S_1 = \{x_i, x_{i+1}\}$. Then:

- If $k < 2(P - 1)$ the smoothness indicator $\psi_{i+1/2}^P$ is close to one and the maximum allowed stencil S_P is used, so that the local accuracy of the method is $2P$.
- If $k > 2(P - 1)$ then all the smoothness indicators would fail, so that the first-order robust numerical method will be used. Nevertheless in this case, $f^{(j)}(x^*) = 0$ for $j = 1, \dots, 2P - 1$ so that, when the local error of the first-order method is estimated through Taylor expansions, only terms of order $O(\Delta x^{2P})$ or bigger will remain. Therefore, in this case the local accuracy of the method is again $2P$.
- If $k = 2(P - 1)$ again the smoothness indicators would fail and the first-order robust numerical method will be used. Since in this case, $f^{(j)}(x^*) = 0$ for $j = 1, \dots, 2P - 2$ the local error of the first-order method is of order $2P - 1$.

Summing up, the local accuracy of the method close to a critical point is always $2P$ with the only exception of critical points of order $2P - 2$: in that case, the order of accuracy will be reduced by one. This order reduction could be avoided by introducing optimal smoothness indicators in the spirit of [17],[18].

4.1.4 Systems of conservation laws

For systems of conservation laws (4.1.4) with $m < 1$ the expression of the ACAT2P method is the same as in the scalar case: the only difference is the computation of the smoothness indicators. In the case of systems, smoothness indicators are first computed for every variable:

$$\psi_{i+1/2}^{j,p}, \quad p = 1, \dots, P,$$

where

- $\psi_{i+1/2}^{j,1}$ is obtained by applying the smoothness indicator (4.1.8), (4.1.10) to the j th component of the numerical solutions $\{u_i^{j,n}\}$.
- $\psi_{i+1/2}^{j,p}$, $p > 2$ is obtained by applying the smoothness indicator (4.1.13) to the j th component of the numerical solutions $\{u_i^{j,n}\}$.
- $\psi_{i+1/2}^{j,2}$ is obtained by applying the smoothness indicator (4.1.13) or (4.1.21) to the j th component of the numerical solutions $\{u_i^{j,n}\}$.

Once these scalar smoothness indicators have been computed, we define

$$\psi_{i+1/2}^p = \min_{j=1,\dots,m} \psi_{i+1/2}^{j,p},$$

so that the selected stencil is the one of maximal length among those in which all the variables are smooth.

Remark 4.1.1 *Standard WENO schemes applied componentwise usually produce oscillatory solutions near shock discontinuities. To alleviate this problem, it is possible to perform a WENO reconstruction on the characteristic variables, as described in [11]. This technique reduces the oscillations but dramatically increases the computational cost. Here we do not feel the need of such a procedure, since our reconstructions are usually much less oscillatory than componentwise WENO.*

4.2 Two-dimensional problems

In this section we focus on the extension of ACAT methods to non-linear two-dimensional systems of hyperbolic conservation laws

$$u_t + f(u)_x + g(u)_y = 0. \quad (4.2.1)$$

The following multi-index notation will be used:

$$\mathbf{i} = (i_1, i_2) \in \mathbb{Z} \times \mathbb{Z},$$

and

$$\mathbf{0} = (0, 0), \quad \mathbf{1} = (1, 1), \quad \mathbf{1/2} = (1/2, 1/2), \quad \mathbf{e}_1 = (1, 0), \quad \mathbf{e}_2 = (0, 1).$$

We consider Cartesian meshes with nodes

$$\mathbf{x}_i = (i_1 \Delta x, i_2 \Delta y).$$

Using this notation, we can write the general form of the CAT2 p method as follows:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} \left[\mathcal{F}_{i-\frac{1}{2}\mathbf{e}_1}^p - \mathcal{F}_{i+\frac{1}{2}\mathbf{e}_1}^p \right] + \frac{\Delta t}{\Delta y} \left[\mathcal{G}_{i-\frac{1}{2}\mathbf{e}_2}^p - \mathcal{G}_{i+\frac{1}{2}\mathbf{e}_2}^p \right], \quad (4.2.2)$$

where the numerical fluxes $\mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^p$, $\mathcal{G}_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^p$ will be computed using the values of the numerical solution $u_{\mathbf{i}}^n$ in the p^2 -point stencil centered at $\mathbf{x}_{\mathbf{i}+\frac{1}{2}} = ((i_1 + 1/2)\Delta x, (i_2 + 1/2)\Delta y)$

$$S_p = \{\mathbf{x}_{\mathbf{i}+\mathbf{j}}, \quad \mathbf{j} \in \mathcal{I}_p\},$$

where

$$\mathcal{I}_p = \{\mathbf{j} = (j_1, j_2) \in \mathbb{Z} \times \mathbb{Z}, \quad -p + 1 \leq j_k \leq p, \quad k = 1, 2\}.$$

See Figure 4.1 for an example.

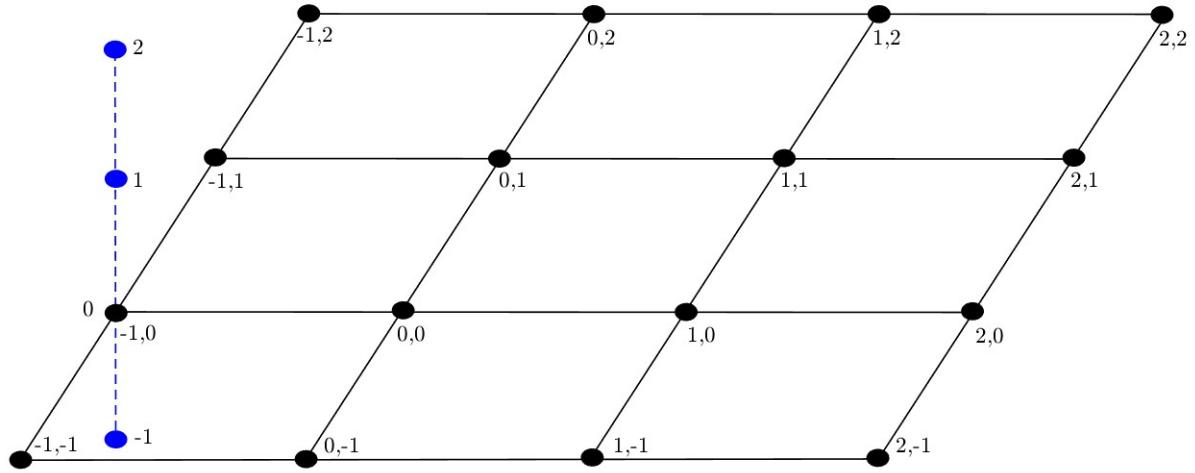


Figure 4.1: Stencil S_2 centered in $\mathbf{x}_{1/2} = (0.5\Delta x, 0.5\Delta y)$

For instance, the expression of the CAT2 numerical flux is as follows:

$$F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^* = \frac{1}{4} \left(\tilde{f}_{\mathbf{i},\mathbf{0}}^{1,n+1} + \tilde{f}_{\mathbf{i},\mathbf{e}_1}^{1,n+1} + f_{\mathbf{i}}^n + f_{\mathbf{i}+\mathbf{e}_1}^n \right), \quad (4.2.3)$$

$$G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^* = \frac{1}{4} \left(\tilde{g}_{\mathbf{i},\mathbf{0}}^{1,n+1} + \tilde{g}_{\mathbf{i},\mathbf{e}_2}^{1,n+1} + g_{\mathbf{i}}^n + g_{\mathbf{i}+\mathbf{e}_2}^n \right), \quad (4.2.4)$$

where

$$\begin{aligned} \tilde{f}_{\mathbf{i},\mathbf{j}}^{1,n+1} &= f \left(u_{\mathbf{i}+\mathbf{j}}^n + \Delta t \tilde{u}_{\mathbf{i},\mathbf{j}}^{(1)} \right), \\ \tilde{g}_{\mathbf{i},\mathbf{j}}^{1,n+1} &= g \left(u_{\mathbf{i}+\mathbf{j}}^n + \Delta t \tilde{u}_{\mathbf{i},\mathbf{j}}^{(1)} \right), \end{aligned}$$

for $\mathbf{j} = \mathbf{0}, \mathbf{e}_1, \mathbf{e}_2$. Furthermore,

$$\begin{aligned} \tilde{u}_{\mathbf{i},\mathbf{0}}^{(1)} &= -\frac{1}{\Delta x} (f_{\mathbf{i}+\mathbf{e}_1}^n - f_{\mathbf{i}}^n) - \frac{1}{\Delta y} (g_{\mathbf{i}+\mathbf{e}_2}^n - g_{\mathbf{i}}^n), \\ \tilde{u}_{\mathbf{i},\mathbf{e}_1}^{(1)} &= -\frac{1}{\Delta x} (f_{\mathbf{i}+\mathbf{e}_1}^n - f_{\mathbf{i}}^n) - \frac{1}{\Delta y} (g_{\mathbf{i}+\mathbf{1}}^n - g_{\mathbf{i}+\mathbf{e}_1}^n), \end{aligned}$$

$$\tilde{u}_{\mathbf{i},\mathbf{e}_2}^{(1)} = -\frac{1}{\Delta x} (f_{\mathbf{i}+\mathbf{1}}^n - f_{\mathbf{i}+\mathbf{e}_2}^n) - \frac{1}{\Delta y} (g_{\mathbf{i}+\mathbf{e}_2}^n - g_{\mathbf{i}}^n),$$

where

$$f_{\mathbf{j}}^n = f(u_{\mathbf{j}}^n), \quad g_{\mathbf{j}}^n = g(u_{\mathbf{j}}^n), \quad \forall \mathbf{j}.$$

Observe that $\tilde{u}_{\mathbf{i},\mathbf{0}}^{(1)} \neq \tilde{u}_{\mathbf{i},\mathbf{e}_1}^{(1)}$ and $\tilde{u}_{\mathbf{i},\mathbf{0}}^{(1)} \neq \tilde{u}_{\mathbf{i},\mathbf{e}_2}^{(1)}$ in opposition to the 1D case where $\tilde{u}_{i,0}^{(1)} = \tilde{u}_{i,1}^{(1)}$.

The following algorithm will be used to compute the numerical fluxes of the CAT2p method:

1. Define

$$\tilde{f}_{\mathbf{i},\mathbf{j}}^{(0)} = f_{\mathbf{i}+\mathbf{j}}^n, \quad \tilde{g}_{\mathbf{i},\mathbf{j}}^{(0)} = g_{\mathbf{i}+\mathbf{j}}^n, \quad \mathbf{j} \in \mathcal{I}_p.$$

2. For $k = 2 \dots 2p$:

(a) Compute

$$\tilde{u}_{\mathbf{i},\mathbf{j}}^{(k-1)} = -A_{p,0}^{1,j_1}(\tilde{f}_{\mathbf{i},(*,j_2)}^{(k-2)}, \Delta x) - A_{p,0}^{1,j_2}(\tilde{g}_{\mathbf{i},(j_1,*)}^{(k-2)}, \Delta y), \quad \mathbf{j} \in \mathcal{I}_p.$$

(b) Compute

$$\tilde{f}_{\mathbf{i},\mathbf{j}}^{k-1,n+r} = f \left(u_{\mathbf{i}+\mathbf{j}}^n + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} \tilde{u}_{\mathbf{i},\mathbf{j}}^{(l)} \right), \quad \mathbf{j} \in \mathcal{I}_p, r = -p+1, \dots, p.$$

(c) Compute

$$\tilde{f}_{\mathbf{i},\mathbf{j}}^{(k-1)} = A_{p,n}^{k-1,0}(\tilde{f}_{\mathbf{i},\mathbf{j}}^{k-1,*}, \Delta t), \quad \mathbf{j} \in \mathcal{I}_p.$$

3. Compute

$$F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^p = \sum_{k=1}^{2p} \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{\mathbf{i},(*,0)}^{(k-1)}, \Delta x), \quad (4.2.5)$$

$$G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^p = \sum_{k=1}^{2p} \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{g}_{\mathbf{i},(0,*)}^{(k-1)}, \Delta y). \quad (4.2.6)$$

The notation used for the approximation of the spacial partial derivatives is the following:

$$A_{p,j_1}^{k,q}(f_{\mathbf{i},(*,j_2)}, \Delta x) = \frac{1}{\Delta x^k} \sum_{l=-p+1}^p \gamma_{p,l}^{k,q} f_{\mathbf{i},(l,j_2)}$$

$$A_{p,j_2}^{k,q}(g_{\mathbf{i},(j_1,*)}, \Delta y) = \frac{1}{\Delta y^k} \sum_{l=-p+1}^p \gamma_{p,l}^{k,q} g_{\mathbf{i},(j_1,l)}$$

Remark 4.2.1 In the last step of the algorithm above the set \mathcal{I}_p can be replaced by its $(2p-1)$ -point subset

$$\mathcal{I}_p^0 = \{\mathbf{j} = (j_1, j_2) \text{ s.t. } j_1 = 0 \text{ or } j_2 = 0\}$$

since only the corresponding values of $\tilde{f}_{\mathbf{i},\mathbf{j}}^{(k-1)}$ are used to compute the numerical fluxes (4.2.5) and (4.2.6).

Once the numerical flux of the CAT2p method has been introduced, the numerical flux of ACAT2 is extended to two-dimensional problems as follows:

$$\mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^1 = \psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^1 F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^* + (1 - \psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^1) F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^{lo}, \quad (4.2.7)$$

$$\mathcal{G}_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^1 = \psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^1 G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^* + (1 - \psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^1) G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^{lo}, \quad (4.2.8)$$

where, $F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^{lo}$ and $G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^{lo}$ are some robust first-order methods; $\psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^1$ and $\psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^1$ are the flux limiters computed dimension by dimension.

Finally, the expression of the ACAT2P method for two-dimensional problems is

$$u_{\mathbf{i}}^{n+1} = u_{\mathbf{i}}^n + \frac{\Delta t}{\Delta x} \left(\mathcal{F}_{\mathbf{i}-\frac{1}{2}\mathbf{e}_1}^{A_1} - \mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^{A_1} \right) + \frac{\Delta t}{\Delta y} \left(\mathcal{G}_{\mathbf{i}-\frac{1}{2}\mathbf{e}_2}^{A_2} - \mathcal{G}_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^{A_2} \right), \quad (4.2.9)$$

where the numerical fluxes are defined as follows: first define the set

$$\mathcal{A}_1 = \{p \in \{2, \dots, P\} \text{ s.t. } \psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^p \cong 1\}, \quad (4.2.10)$$

$$\mathcal{A}_2 = \{p \in \{2, \dots, P\} \text{ s.t. } \psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^p \cong 1\}, \quad (4.2.11)$$

$$(4.2.12)$$

where $\psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^p$, $\psi_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^p$ are the smoothness indicators introduced in Section 4.1.2 computed dimension by dimension. Then define:

$$F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^{A_1} = \begin{cases} F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^* & \text{if } \mathcal{A}_1 = \emptyset; \\ F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^{p_1} & \text{where } p_1 = \max(\mathcal{A}_1) \text{ otherwise;} \end{cases} \quad (4.2.13)$$

$$G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^{A_2} = \begin{cases} G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^* & \text{if } \mathcal{A}_2 = \emptyset; \\ G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^{p_2} & \text{where } p_2 = \max(\mathcal{A}_2) \text{ otherwise.} \end{cases} \quad (4.2.14)$$

Observe that, since the smoothness indicators are computed dimension by dimension, a rectangular stencil

$$S_{p_1, p_1} = \{\mathbf{x}_{\mathbf{i},\mathbf{j}}, \quad i_1 - p_1 + 1 \leq j_1 \leq i_1 + p_1, \quad i_2 - p_2 + 1 \leq j_2 \leq i_2 + p_2\},$$

is used in practice to compute the numerical fluxes $F_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}^{p_1}$, $G_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}^{p_2}$. The extension of CAT methods to such rectangular stencils is straightforward. using the values

4.3 Numerical experiments

In this section we apply ACAT2P methods to several 1D and 2D problems: the 1D linear transport equation, Burgers equation, and the 1D and 2D Euler equation for gas dynamic. The Super Bee flux limiter [63] is used in FL-CAT2 and the smoothness indicators (4.1.13) are used for $p \geq 2$: no loss of precision for first-order critical points has been observed in any of the test problems considered here due to the use of $\psi_{i+1/2}^2$. Fornberg's algorithm is used to compute the coefficients of the numerical differentiation formulas. ACAT methods will be compared with the Lax-Friedrichs (LF), HLL first-order schemes and with WENO($2p + 1$) finite difference methods based on the Lax-Friedrichs splitting (see [60]) combined with SSPRK3 ([25]) for the time discretization. The order and the number of points of their stencils in 1d are recalled in Table 4.1. Since ACAT2P reduces to CAT2P and the order of accuracy of the latter have been checked in Chapter 3, no test order will be considered here.

Method	Stencil	Order
LF	3	1
HLL	3	1
ACAT2 or FL-CAT2	3	2
ACAT2P	$2P + 1$	$2P$
WENO($2p + 1$)-RK3	$2p + 1$	$2p + 1$

Table 4.1: Numerical methods: order of accuracy and number of points of the stencils for 1d problems.

4.3.1 1D Scalar equations

4.3.1.1 Test 4.1 Transport equation - Smooth solutions

Let us consider the linear scalar conservation law

$$u_t + u_x = 0. \quad (4.3.1)$$

with initial condition:

$$u_0(x) = \frac{1}{2} \sin(\pi x) \quad (4.3.2)$$

We solve numerically this problem in the spatial interval $[0, 2]$, using a 160-mesh points, CFL= 0.9, and periodic boundary conditions.

Figure 4.2 and 4.3 show the numerical solutions at time $t = 4$ and $t = 40$ respectively. Zooms of an interest area are included, in which the loss of accuracy with time for the lower order methods can be clearly seen. As it can be observed, the numerical solutions of ACAT4 and ACAT6 match the exact solution at both times while ACAT2 is more

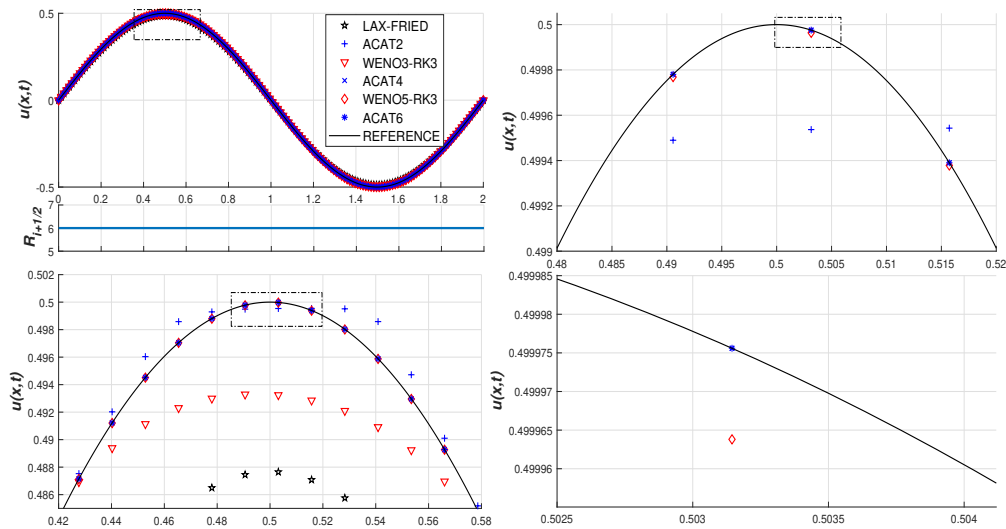


Figure 4.2: Test 4.1. Transport equation with initial condition (4.3.2). Numerical solution at $t = 4$: general view (*left-top*); local order of accuracy for ACAT6 (*sub-frame*); consecutive zooms close to the local maximum (*left-bottom*, *right-top* and *right-bottom*).

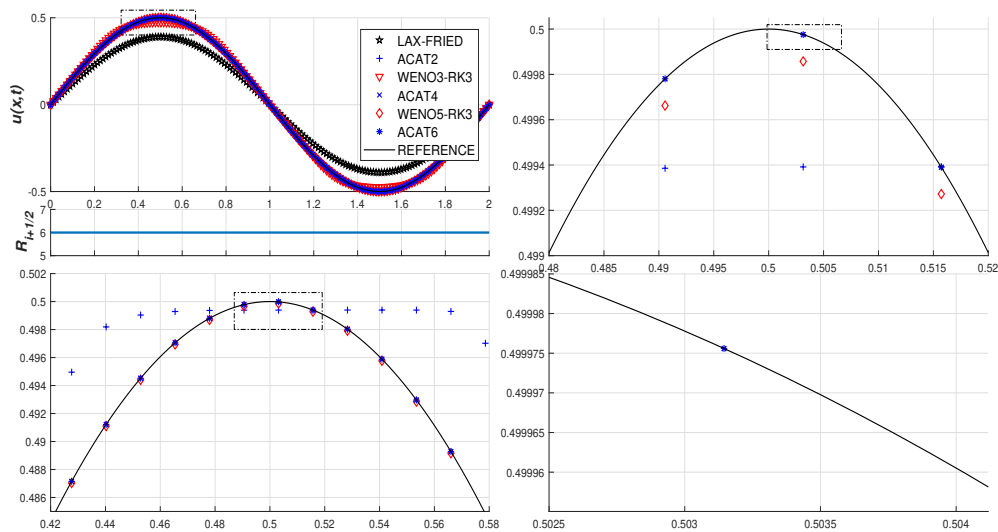


Figure 4.3: Test 4.1 Transport equation with initial condition (4.3.2). Numerical solution at $t = 40$: general view (*left-top*); local order of accuracy for ACAT6 (*sub-frame*); consecutive zooms close to the local maximum (*left-bottom*, *right-top* and *right-bottom*).

diffusive near the critical points. This loss of accuracy close to the critical points can also be observed for WENO-RK methods, although this drawback can be overcome by using optimal weights in the WENO reconstructions: see [17], [18].

The loss of accuracy of ACAT2 close to the critical points compared to ACAT4 or 6 is due to the fact that, while the smoothness indicators $\psi_{i+1/2}^2$ and $\psi_{i+1/2}^3$ are always close to one, the Superbee flux limiter $\psi_{sb,i+1/2}$ detects a discontinuity at the critical points and the first-order method is then locally used: to make this clear, Figure 4.4 (top) shows the solution obtained with ACAT6 at time $t = 4$ for (4.3.1) with initial condition

$$u_0(x) = \frac{1}{2} \sin(2\pi x) \quad (4.3.3)$$

in the interval $[0, 2]$ using again a 160-point mesh, CFL = 0.9, and periodic boundary conditions. Figure 4.4 (down) shows the graph of the three smoothness indicators.

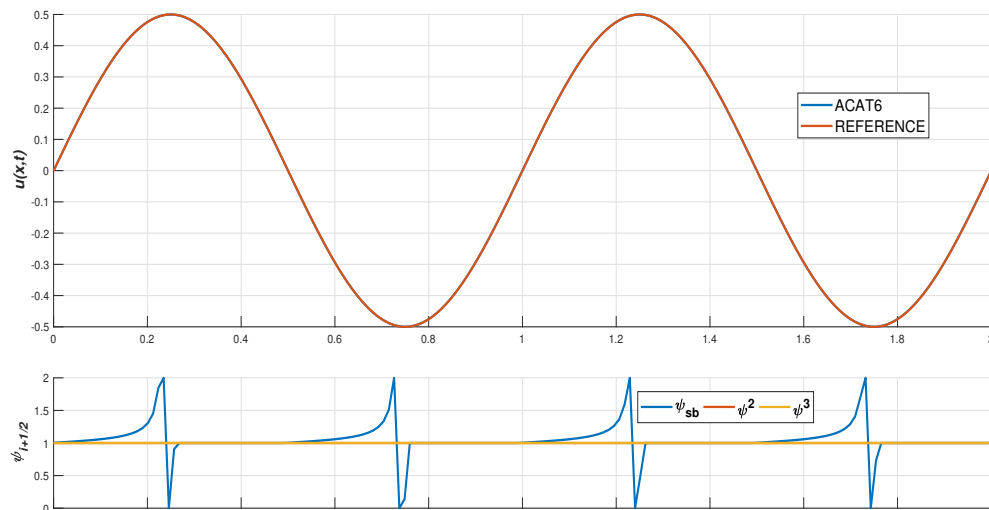


Figure 4.4: Test 4.1. Transport equation with initial condition (4.3.3). Solution obtained with ACAT6 at time 4 (top) and graphs of the smoothness indicators ψ_{sb} , ψ^2 and ψ^3 (bottom).

4.3.1.2 Test 4.2 Transport equation - Discontinuous solutions

We consider next equation (4.3.1) with a piecewise continuous initial condition

$$u_0(x) = \begin{cases} 1 & \text{if } \frac{1}{2} \leq x \leq 1; \\ 0 & \text{if } 0 \leq x < \frac{1}{2} \text{ or } \frac{3}{2} < x \leq 2; \\ -1 & \text{if } 1 < x \leq \frac{3}{2}. \end{cases} \quad (4.3.4)$$

We solve numerically this problem in the spatial interval $[0, 2]$, using again a 160-mesh points, CFL= 0.9, and periodic boundary conditions. Figure 4.5 shows solutions from

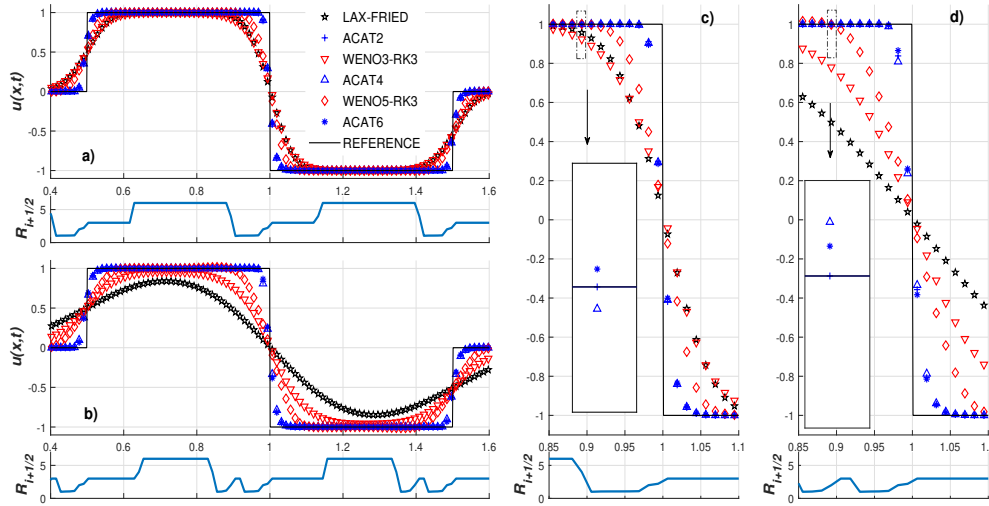


Figure 4.5: Test 4.2. Transport equation with initial condition (4.3.4). Numerical solutions at $t = 2$ (a) and $t = 20$ (b). Zoom of the numerical solutions at time $t = 2$ (c) and $t = 20$ (d). Sub-frames: local order of accuracy for ACAT6.

ACAT $2P$, $P = 2, 4, 6$ and WENO q -RK3, $q = 3, 5$ after 2 and 20 seconds. As it can be observed, ACAT methods capture better the discontinuity than WENO-RK schemes. In this case, ACAT4 and ACAT6 reduce to ACAT2 at the discontinuities due to the order adaptation technique. WENO methods give accurate solutions for short times but spurious oscillations appear with time due to the choice CFL= 0.9.

4.3.1.3 Test 4.3 Burgers equation - Discontinuous solutions

Let us consider the Burgers equation

$$u_t + \left(\frac{u^2}{2} \right)_x = 0, \quad (4.3.5)$$

with initial condition (4.3.2). The problem is numerically solved in the interval $[0, 2]$ using an uniform mesh with 160, CFL= 0.9, and periodic boundary conditions. A reference solution has been computed with the Lax-Friedrichs methods using 1400-point mesh.

Figures 4.6 and 4.7 show respectively the general view and a zoom of the numerical solutions obtained with the different methods at times $t = \{0.25, 0.5, 1, 10\}$. The local order of accuracy of ACAT6 is also shown: as it can be seen, this method reduces to the first-order one only at the shock once it has been generated.

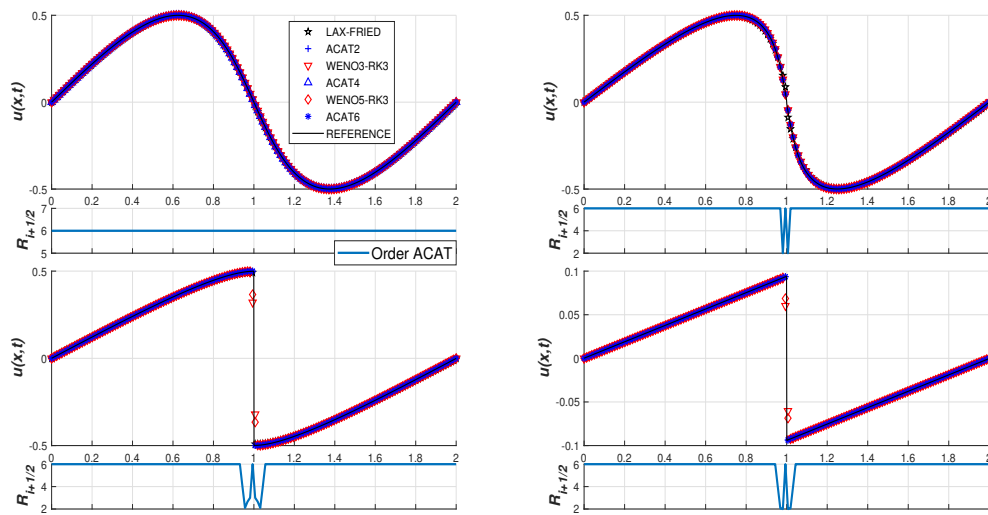


Figure 4.6: Test 4.3. Burgers equation with initial condition (4.3.2). Numerical solutions obtained at times $t = 0.25$ (left-top), $t = 0.5$ (right-top), $t = 1$ (left-bottom), and $t = 10$ (right-bottom). Sub-frames: local order of accuracy for ACAT6.

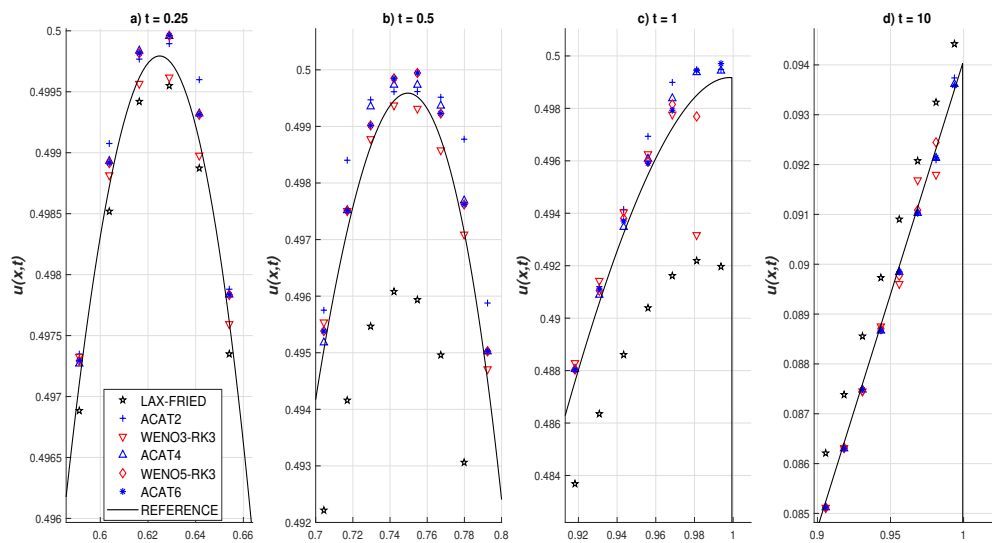


Figure 4.7: Test 4.3. Burgers equation with initial condition (4.3.2). Zoom of the numerical solutions obtained at times $t = 0.25$ (a), $t = 0.5$ (b), $t = 0.1$ (c), and $t = 10$ (d). Sub-frames: local accuracy order for ACAT6.



4.3.2 1D Euler equations

Let us now consider the 1D Euler equations for gas dynamics

$$u_t + f(u)_x = 0, \quad (4.3.6)$$

with

$$u = \begin{bmatrix} \rho \\ \rho v \\ E \end{bmatrix}, \quad f(u) = \begin{bmatrix} \rho v \\ p + \rho v^2 \\ v(E + p) \end{bmatrix}, \quad (4.3.7)$$

where ρ is the density measured in Kg/m^3 ; v , the velocity in m/s ; E the total energy per unit volume in $Kg/(ms^2)$; and p is the pressure in Pascal Pa . We assume an ideal gas with the equation of state

$$p(\rho, e) = (\gamma - 1)\rho e, \quad (4.3.8)$$

being γ the ratio of specific heat capacities of the gas taken as 1.4 and e is the internal energy per unit mass is related to E by:

$$E = \rho(e + 0.5v^2). \quad (4.3.9)$$

We consider three Riemann problems for (4.3.6): the Sod problem [61], the Einfeldt problem [64], and the right blast wave Woodward and Colella problem [65]. In all the cases: the initial discontinuity is placed at $x = 0.5$, the equations are numerically solved at the spatial interval $[0, 1]$ and the exact solution is provided by the HE-E1RPEXACT solver introduced in [3]. The CFL parameter is set to 0.8 and outflow-inflow boundary conditions are considered.

4.3.2.1 Test 4.4 Sod shock tube problem

$$(\rho, v, p) = \begin{cases} (1, 0, 1) & \text{if } x < 1/2, \\ (0.125, 0, 0.1) & \text{if } x > 1/2. \end{cases} \quad (4.3.10)$$

The solution involves a rarefaction wave, a contact discontinuity and a shock. We compare the numerical solutions with the exact one: see [3].

Figure 4.8 shows the solutions provided by ACAT2-4-6 and WENO3-5 for density, velocity, internal energy and pressure p using a 200 points. The local accuracy of ACAT6 is also shown. Zooms of the behaviour of the numerical densities can be observed in Figure 4.9. As it can be seen in zooms *a* and *b*, WENO5-RK3 gives sharper but more oscillatory solutions than ACAT methods. Moreover, increasing the accuracy order for ACAT methods we obtain sharper results. Similar conclusions for the internal energy can be drawn: see Figure 4.10.

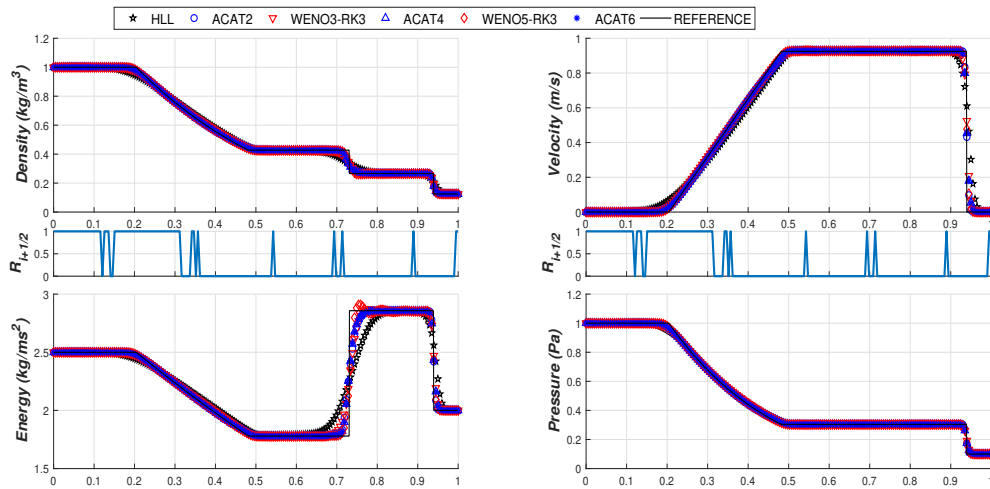


Figure 4.8: Test 4.4. 1D Euler equations: the Sod problem. Numerical solutions at $t = 0.25$ using CFL= 0.8 and 200 points: density (*left-top*), velocity (*right-top*), internal energy (*left-bottom*), pressure (*right-bottom*). Sub-frames: local order of accuracy for ACAT6.

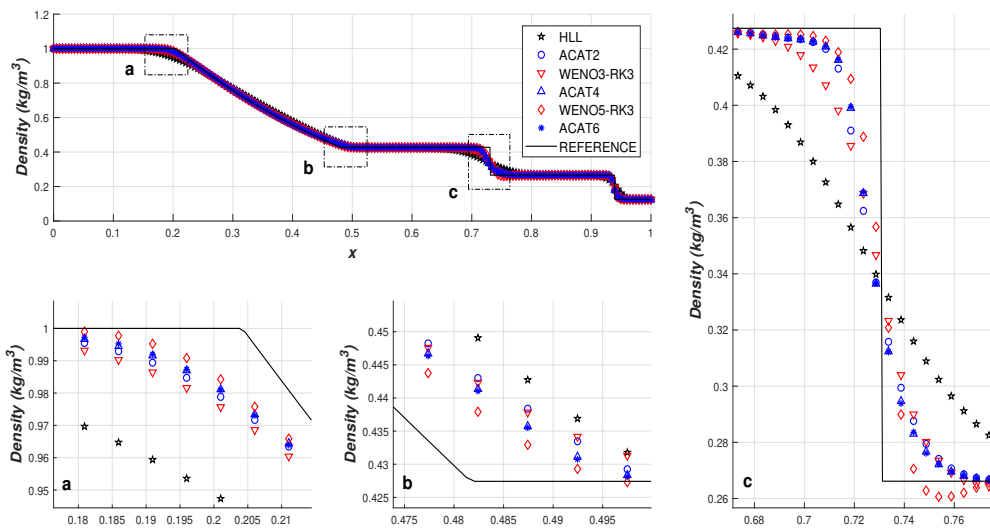


Figure 4.9: Test 4.4. 1D Euler equations: the Sod problem. Numerical density at $t = 0.25$ using CFL= 0.8 and 200 points: general view and zooms close to the points *a*, *b*, *c* and *d*.

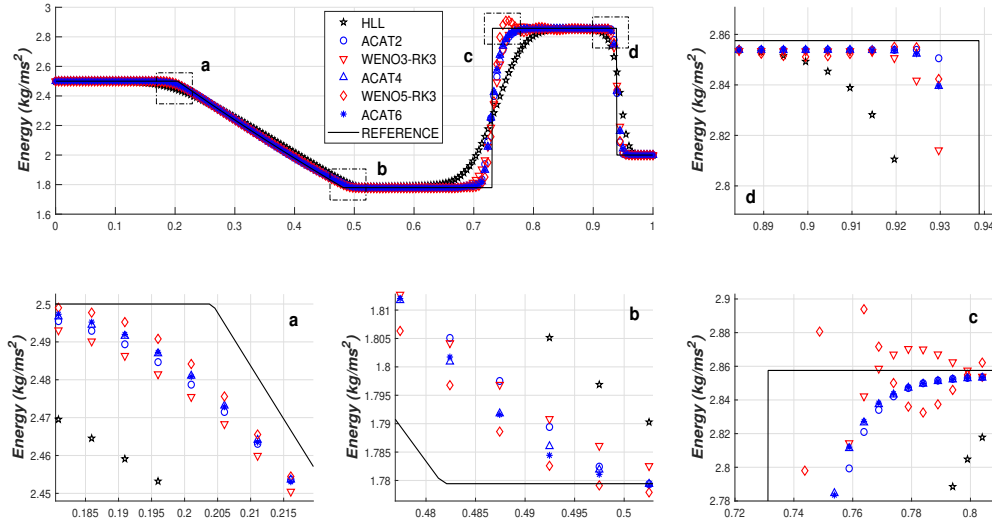


Figure 4.10: Test 4.4 1D Euler equations: the Sod problem. Numerical internal energy at $t = 0.25$ using CFL= 0.8 and 200 points: general view and zooms close to the points a, b, c and d . 1D Euler equations.

4.3.2.2 Test 4.5 123 Einfeldt problem

$$(\rho, v, p) = \begin{cases} (1.0, -2.0, 0.4) & \text{if } x < 1/2, \\ (1.0, 2.0, 0.4) & \text{if } x > 1/2. \end{cases} \quad (4.3.11)$$

The solutions of this problem involves two strong rarefaction waves and an intermediate state that is close to vacuum, what makes this problem a hard test for numerical methods. ACAT methods give stable solutions under CFL ≤ 1 condition: Figure 4.11 shows the time evolution of the numerical results obtained with ACAT6. The smoothness indicators $\psi_{i+1/2}^3$ is also depicted: it can be seen how the discontinuities of the first-order derivatives are correctly captured. It can be also observed that, while at the rarefaction waves order 6 is selected, lower accuracy is used at the constant regions close to the boundaries: this order reduction is due to the numerical oscillations produced by the 6th-order method. A comparison of the different methods at time $t = 0.15$ is shown in Figure 4.12, where ACAT methods provide similar stable solutions. Although WENO solutions are stable, the third-order one is diffusive and the fifth-order one is oscillatory.

4.3.2.3 Test 4.6 Right blast wave problem of Woodward & Colella

$$(\rho, u, p) = \begin{cases} (1.0, 0.0, 1000) & \text{if } x < 1/2, \\ (1.0, 0.0, 0.01) & \text{if } x > 1/2. \end{cases} \quad (4.3.12)$$

For this test we use 450 points. The solution involves two strong shocks. Figure 4.13 shows the numerical densities obtained at time $t = 0.012s$: it can be observed that WENO

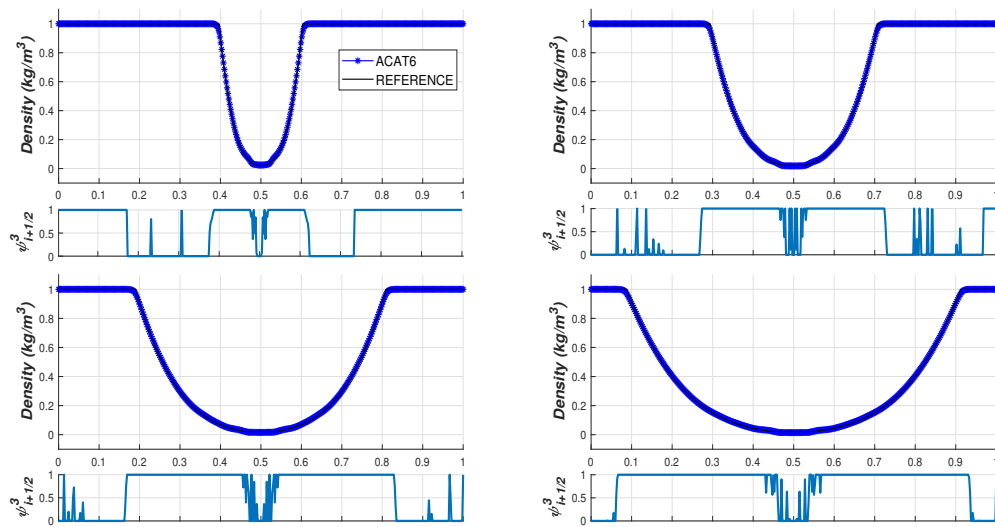


Figure 4.11: Test 4.5. 1D Euler equations: the 123 Einfeldt problem using CFL= 0.8 and 200 points. Density obtained with ACAT6 and graph of the smoothness indicator ψ^3 for $t = t_s/4$ (left-top), $t_s/2$ (right-top), $3t_s/4$ (left-bottom), t_s (right-bottom), with $t_s = 0.15$.

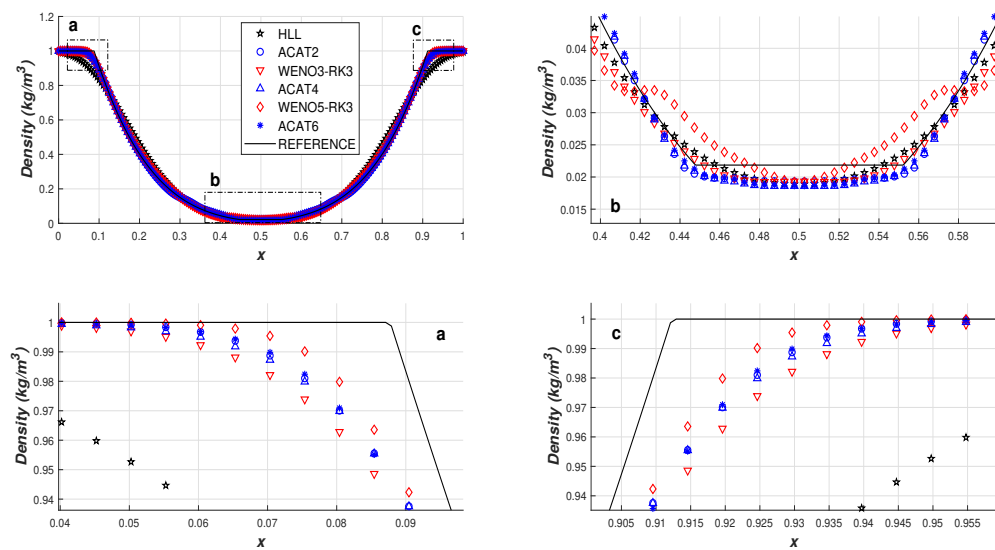


Figure 4.12: Test 4.5. 1D Euler equations: the 123 Einfeldt problem using CFL= 0.8 and 200 points. Numerical densities at time $t = 0.15$: general view (left-top) and zooms close to the points a (left-bottom), b (right-top), and c (right-bottom).

methods produce oscillating solutions, while ACAT methods give stable solutions whose accuracy increase with the order. In particular, this behavior can be seen in the two



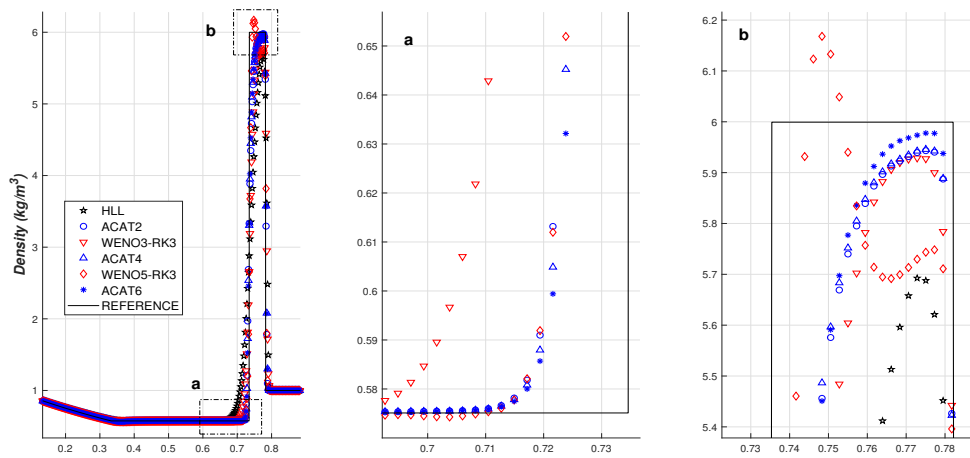


Figure 4.13: Test 4.6. 1D Euler equations: right blast wave of the Woodward & Colella problem. Numerical densities at time $t = 0.012$, using CFL= 0.8 (*left*) and zooms close to the shocks (*center and right*).

zooms close to the shocks.

Table 4.2 shows the CPU time rates for this last one-dimensional test. A non-optimized implementation using Matlab has been used for all the numerical methods. Therefore, this table has to be taken as a rough indication about computational cost. In particular, ACAT methods are highly parallelisable and do not need the storage of intermediate temporal stages: therefore, an optimized parallel implementation can lead to very different conclusions. With the implementations used here, ACAT2 is the cheapest method and its CPU time is taken as a reference. ACAT4 is competitive both in quality and computational cost compared to WENO-RK 3 and 5. The practical use of ACAT of order higher or equal than 6 requires an efficient implementation, otherwise the computational cost to increase the order is very big. The same happens with WENO-RK methods when the accuracy in time is increased due to the large number of stages required by SSPK methods.

ACAT2	ACAT4	ACAT6
1.00	5.88	12.46
WENO3-RK3	WENO5-RK3	
2.86	5.08	

Table 4.2: CPU time rates for the Woodward and Colella problem.

4.3.3 2D Equations

4.3.3.1 Test 4.7 Transport equation

Let us consider the 2D transport equation

$$u_t + au_x + bu_y = 0, \quad (4.3.13)$$

with initial conditions

$$u = \begin{cases} 1 & \text{if } x + y \leq 1/4, \\ 0 & \text{otherwise.} \end{cases} \quad (4.3.14)$$

We solve (4.3.13) on the spatial domain $[0, 2] \times [0, 2]$, using: $a, b = 1$, 100×100 -point grid, CFL=0.5, and $t = 1$. Figure 4.14 shows a 1D cut over the line $y = x$ of the solutions obtained with ACAT2, ACAT4, WENO3-RK3 and WENO5-RK3 at time $t = 1$.

4.3.3.2 Test 4.8 - 4.10 Euler equations

Let us consider the two-dimensional Euler equations for gas dynamics

$$u_t + f(u)_x + g(u)_y = 0, \quad (4.3.15)$$

where

$$u = \begin{pmatrix} \rho \\ \rho v \\ \rho w \\ E \end{pmatrix}, \quad f(u) = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ \rho v w \\ v(E + p) \end{pmatrix}, \quad g(u) = \begin{pmatrix} \rho w \\ \rho v w \\ \rho w^2 + p \\ w(E + p) \end{pmatrix}.$$

Here, ρ is the density; v, w are the components of the velocity in the x and y directions; E , the total energy per unit volume; p , the pressure. We consider the equation of state

$$p(\rho, v, w, E) = (\gamma - 1)(E - \frac{\rho}{2}(v^2 + w^2)), \quad (4.3.16)$$

and γ is the ratio of specific heat capacities of the gas taken as 1.4.

We solve numerically (4.3.15) using ACAT2 and ACAT4 for three of the nineteen configurations of the 2-D Riemann problems presented in [66] whose initial conditions are given in Tables 4.3-4.4. These initial conditions consist of constant states at every quadrant of the spatial domain that are chosen so that the 1D Riemann problems corresponding to two adjacent states consist of only one one-dimensional simple wave: a shock S , a rarefaction wave R , or a slip line i.e. a contact discontinuity with discontinuous tangential velocity J . The sub-indexes $(l, r) \in \{(2, 1), (3, 2), (3, 4), (4, 1)\}$ indicate the involved quadrants. For shocks and rarefactions an over-arrow indicate the direction (backward or forward). And for contact discontinuities a sign $+/-$ is used (instead of the over-arrow), to denote whether it is a positive or negative slip line.

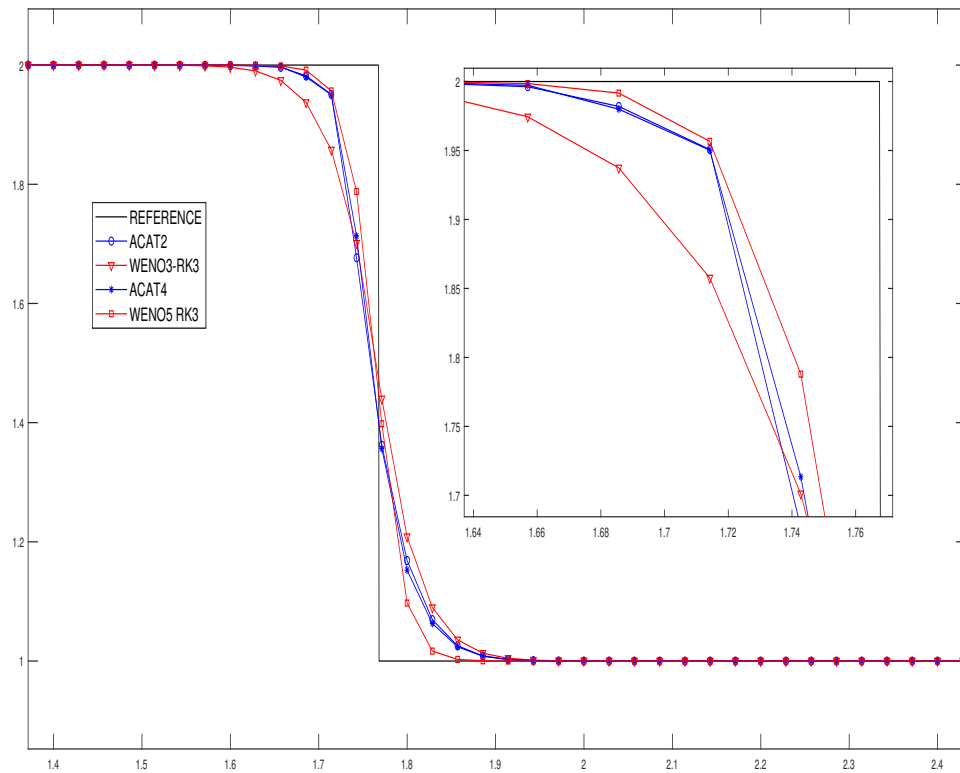


Figure 4.14: Test 4.7. 2D Transport equation: solution obtained with ACAT2, ACAT4, WENO3 RK3 and WENO5 RK3 at time $t = 1$: cut with a vertical plane passing through the line $y = x$. Subplot: zoom close to the discontinuity

These Riemann problems are numerically solved using a 400×400 -point grid and free boundary conditions. The CFL condition used to set the time steps is the following

$$\Delta t = \frac{\text{CFL}}{2} \min \left(\frac{\Delta x}{\text{smax}_x}, \frac{\Delta y}{\text{smax}_y} \right),$$

where

$$\text{smax}_x = \max_{i,j} \{ |v_{i,j}^n| + c_{i,j} \}, \quad \text{smax}_y = \max_{i,j} \{ |w_{i,j}^n| + c_{i,j} \},$$

with

$$c = \sqrt{\frac{\gamma p}{\rho}}.$$

The CFL parameter is set to 0.475.

Lax		Configuration 4						
$p_2 = 0.35$	$\rho_2 = 0.5065$	$p_1 = 1.1$	$\rho_1 = 1.1$					
$u_2 = 0.8939$	$v_2 = 0.0$	$u_1 = 0.0$	$v_1 = 0.0$			$\overleftarrow{S}_{2,1}$		
$p_3 = 1.1$	$\rho_3 = 1.1$	$p_4 = 0.35$	$\rho_4 = 0.5065$		$\overrightarrow{S}_{3,2}$		$\overrightarrow{S}_{4,1}$	
$u_3 = 0.8939$	$v_3 = 0.8939$	$u_4 = -0.0$	$v_4 = 0.8939$			$\overleftarrow{S}_{3,4}$		

Table 4.3: 2D Euler equations: test 8. Initial condition.

Lax		Configuration 6						
$p_2 = 1.0$	$\rho_2 = 2.0$	$p_1 = 1.0$	$\rho_1 = 1.0$					
$u_2 = 0.75$	$v_2 = 0.5$	$u_1 = 0.75$	$v_1 = -0.5$			$J_{2,1}^-$		
$p_3 = 1.0$	$\rho_3 = 1.0$	$p_4 = 1.0$	$\rho_4 = 3.0$		$J_{3,2}^+$		$J_{4,1}^+$	
$u_3 = -0.75$	$v_3 = 0.5$	$u_4 = -0.75$	$v_4 = -0.5$			$J_{3,4}^-$		

Table 4.4: 2D Euler equations: test 9. Initial condition.

Lax		Configuration 8						
$p_2 = 1.0$	$\rho_2 = 1.0$	$p_1 = 0.4$	$\rho_1 = 0.5197$					
$u_2 = -0.6259$	$v_2 = 0.1$	$u_1 = 0.1$	$v_1 = 0.1$			$\overleftarrow{R}_{2,1}$		
$p_3 = 1.0$	$\rho_3 = 0.8$	$p_4 = 1.0$	$\rho_4 = 1.0$		$J_{3,2}^-$		$\overleftarrow{R}_{4,1}$	
$u_3 = 0.1$	$v_3 = 0.1$	$u_4 = 0.1$	$v_4 = -0.6259$			$J_{3,4}^-$		

Table 4.5: 2D Euler equations: test 10. Initial condition.

Figures 4.15, 4.16 and 4.17 show the numerical solutions for the density density given by ACAT2 and ACAT4. We include in each figure a general view of the numerical density given by ACAT2 (left-top) and ACAT4 (right-top); the smoothness indicators ψ_x^1 (left-center) and ψ_x^2 (right-center) in the x -direction; the smoothness indicators ψ_y^1 (left-bottom) and ψ_y^2 (right-bottom) in the y -direction. In all cases, the solutions are stable and similar of those obtained in [67] with a finite volume method. Observe how the indicators

ψ_x^2 and ψ_y^2 detect better the smoothness regions than ψ_x^1 and ψ_y^1 , what implies a better resolution in the numerical solutions obtained with ACAT4. However, the computational cost increases with the order as it happens for 1d problems, see table 4.6.

ACAT2 1.00	ACAT4 9.98	ACAT6 96.91
WENO3-RK3 3.23	WENO5-RK3 9.968	

Table 4.6: 2D Euler equations test 10: CPU time rates.

In Figure 4.18 the numerical densities obtained with ACAT2, ACAT4, WENO3 RK3, and WENO5 RK5 at time $t = 0.25$ are compared.

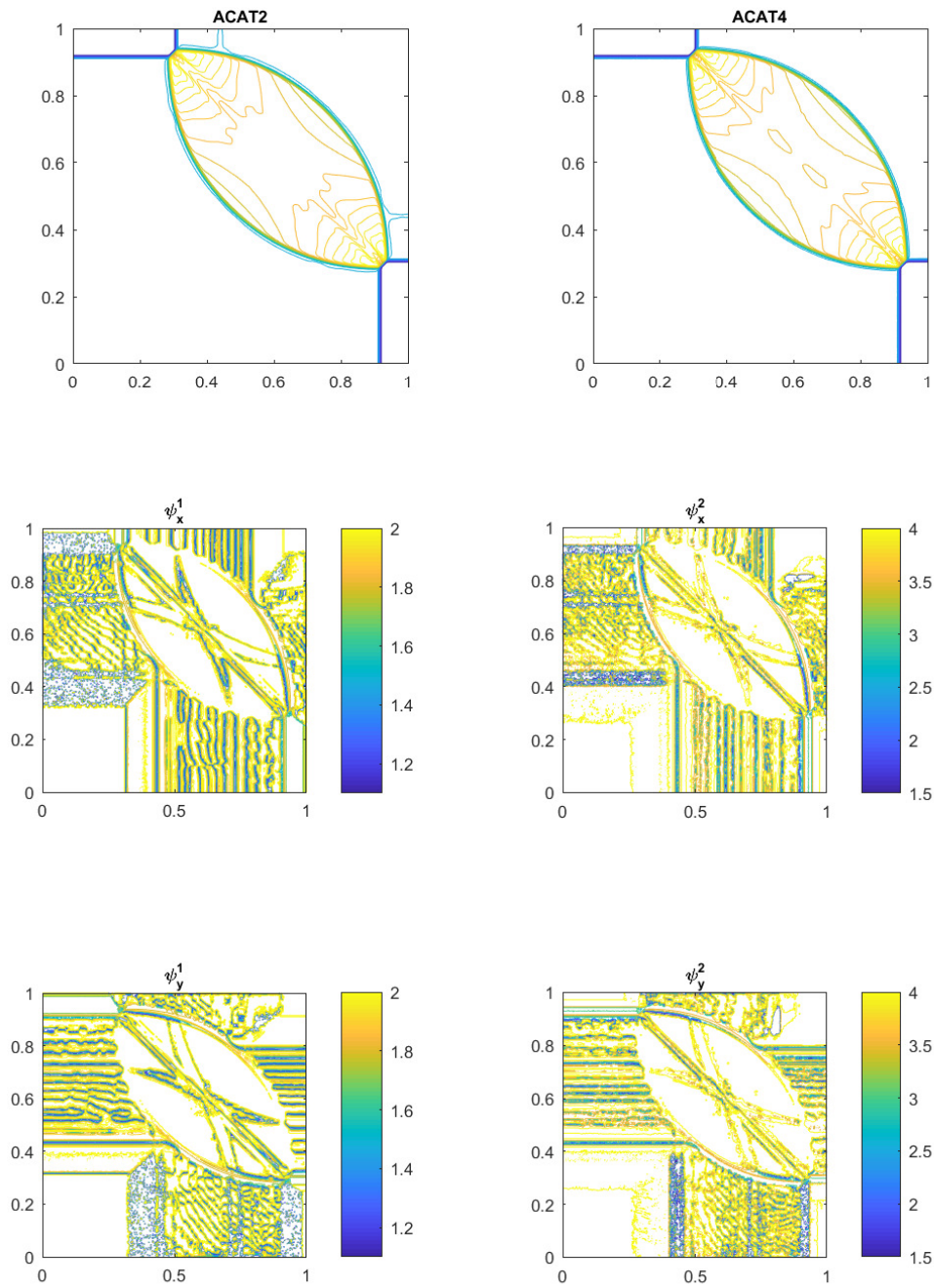


Figure 4.15: Test 4.8. 2D Euler equations: contour plots of the density at time $t = 0.25$ obtained with ACAT2 (*left-top*) and ACAT4 (*right-top*). Contour plots of the smoothness indicators ψ_x^1 (*left-center*), ψ_x^2 (*right-center*), ψ_y^1 (*left-bottom*) and ψ_y^2 (*right-bottom*).

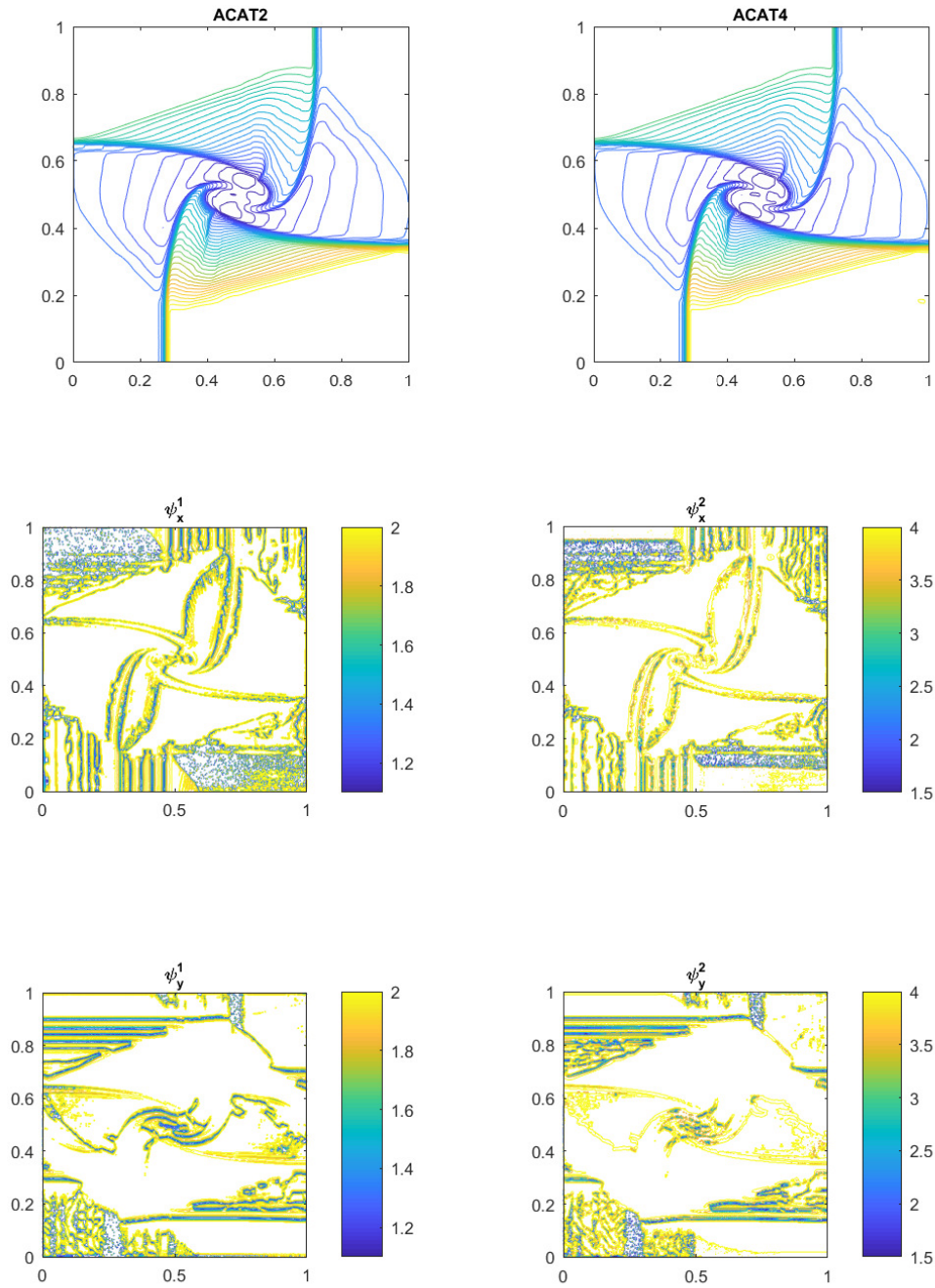


Figure 4.16: Test 4.9. 2D Euler equations: contour plots of the density at time $t = 0.3$ obtained with ACAT2 (*left-top*) and ACAT4 (*right-top*). Contour plots of the smoothness indicators ψ_x^1 (*left-center*), ψ_x^2 (*right-center*), ψ_y^1 (*left-bottom*) and ψ_y^2 (*right-bottom*).

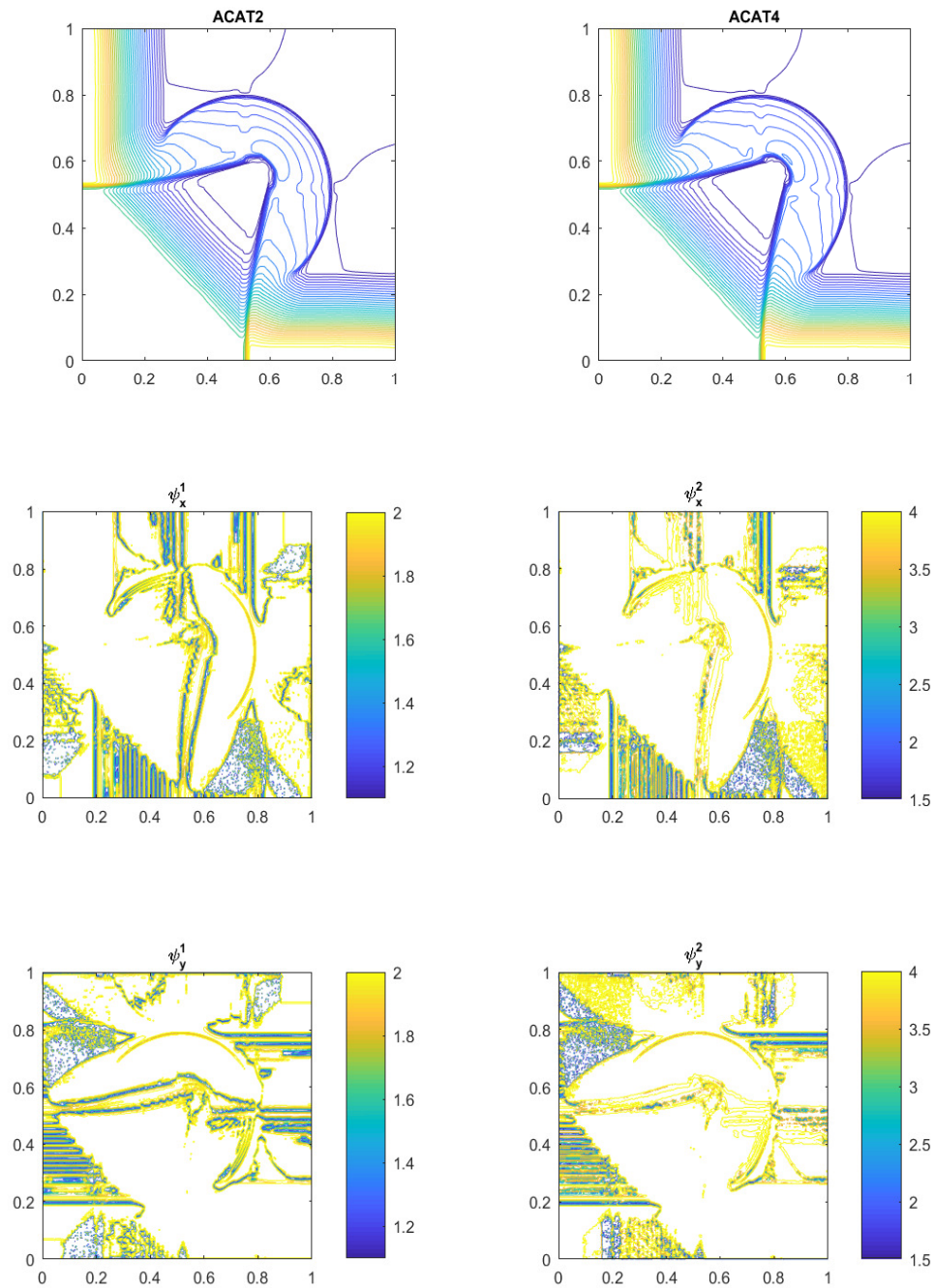


Figure 4.17: Test 4.10. 2D Euler equations: contour plots of the density at time $t = 0.25$ obtained with ACAT2 (*left-top*) and ACAT4 (*right-top*). Contour plots of the smoothness indicators ψ_x^1 (*left-center*), ψ_x^2 (*right-center*), ψ_y^1 (*left bottom*) and ψ_y^2 (*right-bottom*).

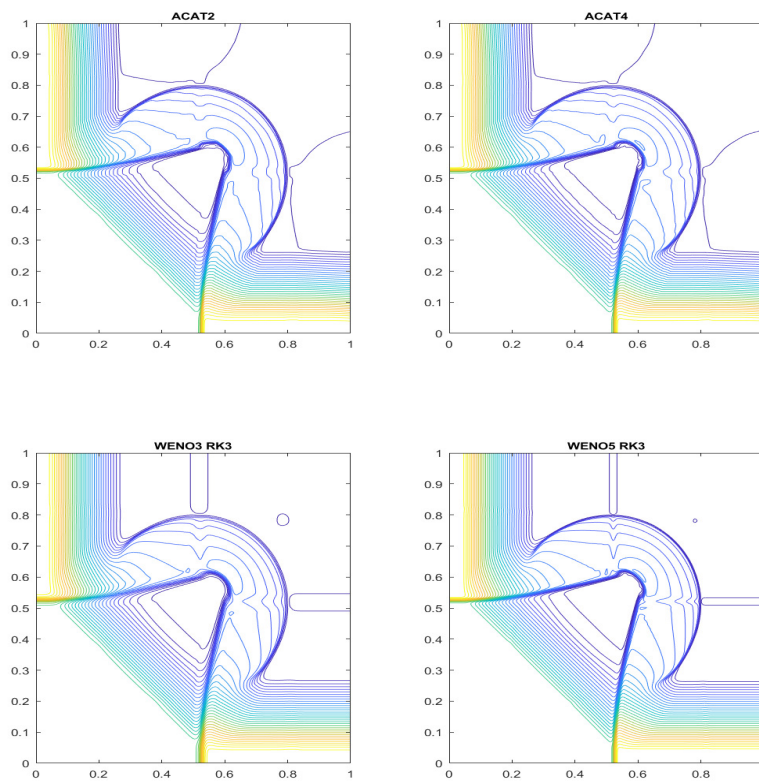


Figure 4.18: Test 4.10 2D Euler equations: contour plots of the density at time $t = 0.25$ obtained with ACAT2 (*left-top*), ACAT4 (*right-top*), WENO3 RK3 (*left-bottom*) and WENO5 RK3 (*right-bottom*).

Chapter 5

Approximate Taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions

As it has been seen in Chapter 4 LAT and CAT methods produce oscillations close to the discontinuities of the solution. The use of Weighted Essentially Non-Oscillatory (WENO) reconstructions (see [15], [16]) to compute the first-order time derivatives allows one to prevent these oscillations: this technique has been used in [10] and also in Section 3.3.2. The goal of this chapter is to explore the potentiality of the combination WENO-CAT by considering different WENO implementations.

WENO methods present high-order accuracy in smooth zones and avoid oscillatory behaviours close to discontinuities through the construction of non-linear weights based on some smooth indicators. Many variants of the original WENO reconstruction have been introduced. For instance, in FWENO methods introduced in [17], new smoothness indicators have been proposed that require a lower number of calculations than the ones proposed by Jiang and Shu.

On the other hand, the expression of the weights in the original WENO method leads to an undesired loss of accuracy near critical points. Different variants have been introduced to deal with this difficulty: see [51], [68], [69], [70]. To the best of our knowledge the only approach that allows one to unconditionally attain the optimal order of accuracy regardless of the order of critical points is, for third-order reconstructions, the OWENO3 method introduced in [18] and, for reconstructions of order higher than 3, the OWENO methods presented in [19]. In this latter reference, the Jiang-Shu smoothness indicators are used to define the weights (for third-order methods these indicators coincide with those of FWENO methods). In this work, the following WENO reconstructions will be used:

- OWENO3 method for third-order reconstructions;
- WENO methods based on the expression of the OWENO weights and the smoothness indicators of FWENO, so that they are both fast and optimal.

For shortness, we will refer to these methods as FOWENO reconstructions.

In this chapter we introduce two new families of high-order numerical methods based on FOWENO reconstructions for the spacial discretization and on LAT or CAT for the time discretization. These methods will be compared between them and against the standard WENO-TVDRK schemes in a number of test cases ranging from scalar linear 1D problems to nonlinear systems of conservation laws in 2D.

The chapter is organized as follows. In section 5.1, the Approximate Taylor Lax-Wendroff [10] and the Compact Approximate Taylor Lax-Wendroff (see chapter 5) methods are briefly recalled. In section 5.2, we introduce the fast and optimal WENO reconstructions: the general idea behind the fast smoothness indicators described in [17] is given as well as their extension to the OWENO smoothness indicators [17] for high-order optimal reconstructions. In section 5.3, the ingredients already described in section 5.1 and 5.2 are combined to construct FOWENO-LAT and FOWENO-CAT methods. Section 5.4 focuses on the comparison of the numerical methods obtained by combining WENO or FOWENO spatial discretization with TVDRK, LAT, or CAT time discretization. A number of tests involving the 1D linear transport equation, Burgers equation, and the 1D and 2D Euler equations of gas dynamics are considered. The quality of the solutions and the CPU run-time are compared and discussed. Finally in Section 5.5 numerical errors corresponding to the ACAT methods and the WENO/FOWENO-APT methods for some selected problems are shown and the efficiencies of the methods are compared.

5.1 Approximate Taylor Methods

For the sake of simplicity, let us describe briefly the AT methods for the one-dimensional scalar case.

5.1.1 Lax-Wendroff Approximate Taylor Methods

In Lax-Wendroff Approximate Taylor(LAT) methods, the time derivatives $\partial_t^k u$ are approximated by applying a first-order numerical differentiation formula in space to some approximations

$$\tilde{f}_i^{(k-1)} \cong \partial_t^{k-1} f(u)(x_i, t_n) \tag{5.1.1}$$

that will be computed by using recursively Taylor expansions in time.

LAT methods are based on centered $(2p + 1)$ -point numerical differentiation formulas

$$f^{(k)}(x_i) \simeq D_{p,i}^k(f, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f(x_{i+j}). \tag{5.1.2}$$



The following notation

$$D_{p,i}^k(f_*, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k f_{i+j}, \quad (5.1.3)$$

will be used to indicate that the formula is applied to some approximations f_i of f and not to its exact point values $f(x_i)$. In cases where there are two or more indexes, the symbol $*$ will be used to indicate with respect to which the differentiation is applied. For instance:

$$\begin{aligned} \partial_x^k u(x_i, t_n) &\simeq D_{p,i}^k(u_*^n, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p}^p \delta_{p,j}^k u_{i+j}^n, \\ \partial_t^k u(x_i, t_n) &\simeq D_{p,n}^k(u_i^*, \Delta t) = \frac{1}{\Delta t^k} \sum_{r=-p}^p \delta_{p,r}^k u_i^{n+r}. \end{aligned}$$

Once the approximations (5.1.1) have been computed, the time derivatives of the solution are approximated by:

$$\partial_t^k u(x_i, t_n) \simeq \tilde{u}_i^{(k)} = -D_{p,i}^1(\tilde{f}_*^{(k-1)}, \Delta x) = -\frac{1}{\Delta x} \sum_{j=-p}^p \delta_{p,j}^1 \tilde{f}_{i+j}^{(k-1)}.$$

A recursive procedure is followed to compute the approximation of the time derivatives: once u_i^l , $l = 0, \dots, k$ have been computed, a Taylor expansion of degree k is used to compute approximations $\tilde{f}_i^{k-1, n+r}$ of $f(u(x_i, (n+r)\Delta t))$, $r = -p, \dots, p$; the centered differentiation formula is then used to obtain $\tilde{f}_i^{(k-1)}$; and, finally, the first-order derivative in space is applied to $\tilde{f}_{i+j}^{(k-1)}$, $j = -p, \dots, p$ to compute u_i^{k+1} . Once all the time derivatives are approximated, the Taylor expansion

$$u_i^{n+1} = u_i^n + \sum_{k=1}^m \frac{\Delta t^k}{k!} \tilde{u}_i^{(k)} + \mathcal{O}(\Delta t^{m+1}) \quad (5.1.4)$$

is used to update the numerical solutions.

The procedure can be summarized as follows:

1. Define

$$\tilde{f}_i^{(0)} = f(u_i^n).$$

2. Compute

$$\tilde{u}_i^{(1)} = -D_{p,i}^1(\tilde{f}_*^{(0)}, \Delta x). \quad (5.1.5)$$

3. For $k = 2, \dots, m$:

(a) Compute

$$\tilde{f}_i^{k-1, n+r} = f \left(u_i^n + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} \tilde{u}_i^{(l)} \right), \quad r = -p, \dots, p.$$

(b) Compute

$$\tilde{f}_i^{(k-1)} = D_p^{k-1}(\tilde{f}_i^{k-1,*}, \Delta t). \quad (5.1.6)$$

(c) Compute

$$\tilde{u}_i^{(k)} = -D_{p,i}^1(\tilde{f}_i^{(k-1)}, \Delta x). \quad (5.1.7)$$

4. Update the solution by (5.1.4).

The order of the method is $\min(m, 2p)$.

Remark 5.1.1 Although, for the sake of clarity, m and p have been considered as two arbitrary positive integers in the presentation of LAT methods, in [10] m is an odd number (since the method is combined with WENO reconstructions) and p is chosen adequately to obtain order m . More precisely, in formulas (5.1.7),

$$p = \left\lceil \frac{m+1-k}{2} \right\rceil,$$

where $\lceil \cdot \rceil$ is the ceiling function, and in formulas (5.1.6)

$$p = \frac{m-1}{2}.$$

LAT methods can be written in conservative form. To see this, let us introduce the family of interpolatory numerical differentiation formulas

$$f^{(k)}(x_i + q\Delta x) \simeq A_{p,i}^{k,q}(f, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-p+1}^p \gamma_{p,j}^{k,q} f(x_{i+j}), \quad (5.1.8)$$

that approximates the k -th derivative of a function at the point $x_i + q\Delta x$ using its values at the $2p$ points $x_{i-p+1}, \dots, x_{i+p}$. The symbol $*$ will be used again to indicate with respect to which index the differentiation is performed.

The following relation holds (see [12]):

$$D_{p,i}^k(f, \Delta x) = \frac{1}{\Delta x} \left(A_{p,i}^{k-1,1/2}(f, \Delta x) - A_{p,i-1}^{k-1,1/2}(f, \Delta x) \right). \quad (5.1.9)$$

Using this equality with $k = 1$, we can write LAT methods in the form

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} \left(F_{i-1/2}^p - F_{i+1/2}^p \right), \quad (5.1.10)$$

where

$$F_{i+1/2}^p = \sum_{k=1}^m \frac{\Delta t^{k-1}}{k!} A_{p,i}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x). \quad (5.1.11)$$

5.1.2 Compact Approximate Taylor methods

CAT methods are based on the conservative expression (5.1.10)-(5.1.11), with the difference that now only the values

$$u_{i-p+1}^n, \dots, u_{i+p}^n, \quad (5.1.12)$$

are used to compute the numerical flux $F_{i+1/2}$, so that a centered $(2p+1)$ -point stencil is used to compute u_i^{n+1} . The numerical flux is thus computed as follows:

$$F_{i+1/2}^p = \sum_{k=1}^m \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x). \quad (5.1.13)$$

where

$$\tilde{f}_{i,j}^{(k-1)} \cong \partial_t^{k-1} f(u)(x_{i+j}, t_n), \quad j = -p+1, \dots, p \quad (5.1.14)$$

are *local* approximations of the time derivatives of the flux. By *local* we mean that these approximations depend on the stencil, i.e.

$$i_1 + j_1 = i_2 + j_2 \not\Rightarrow \tilde{f}_{i_1,j_1}^{(k-1)} = \tilde{f}_{i_2,j_2}^{(k-1)}.$$

Local approximations of the time derivatives of the solution

$$\tilde{u}_{i,j}^{(k)} \cong \partial_t^{(k)} u(x_{i+j}, t_n), \quad j = -p+1, \dots, p$$

are obtained then by using the non-centered differentiation formulas

$$\tilde{u}_{i,j}^{(k)} = -A_{p,0}^{1,j}(\tilde{f}_{i,*}^{(k-1)}, \Delta x) = -\frac{1}{\Delta x} \sum_{r=-p+1}^p \gamma_{p,r}^{1,j} \tilde{f}_{i,r}^{(k-1)}.$$

Like in LAT methods, these local approximations of the time derivatives are recursively used to compute approximations of the flux forward and backward in time using Taylor expansions in a recursive way.

Given i , the procedure to compute $F_{i+1/2}^p$ is as follows:

1. Define

$$\tilde{f}_{i,j}^{(0)} = f(u_{i+j}^n), \quad j = -p+1, \dots, p.$$

2. For $k = 2 \dots m$:

(a) Compute

$$\tilde{u}_{i,j}^{(k-1)} = -A_{p,0}^{1,j}(\tilde{f}_{i,*}^{(k-2)}, \Delta x).$$

(b) Compute

$$\tilde{f}_{i,j}^{k-1,n+r} = f \left(u_{i+j}^n + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} \tilde{u}_{i,j}^{(l)} \right), \quad j, r = -p+1, \dots, p.$$

(c) Compute

$$\tilde{f}_{i,j}^{(k-1)} = A_{p,n}^{k-1,0}(\tilde{f}_{i,j}^{k-1,*}, \Delta t), \quad j = -p+1, \dots, p.$$

3. Compute $F_{i+1/2}^p$ by (5.1.13)

Once the numerical fluxes have been computed, the numerical solution is updated by using (3.1.28).

In [12] it has been shown that:

- The order of the method is $\min(m, 2p)$ so that the optimal choice is $m = 2p$: the corresponding numerical method will be represented by CAT2p in the sequel.
- CAT2p reduces to the standard Lax-Wendroff method for linear problems.
- CAT2p is linearly stable under the standard CFL condition.

The extension of LAT and CAT methods to systems is straightforward by applying the schemes component by component. The extension to multiple dimensions using Cartesian grids can be done through the methods of lines. For a 2D problem, CAT uses a rectangular stencil of p^2 points centered in a point $(x_{i+1/2}, y_{j+1/2})$ to compute the horizontal component of the numerical flux at $(x_{i+1/2}, y_j)$ and the vertical component at $(x_i, y_{j+1/2})$ on the basis of local approximations of the time derivatives and applications of Taylor expansions.

5.2 Fast and optimal WENO reconstructions

Approximate Taylor methods produce spurious oscillations near discontinuities due to the Gibbs phenomenon. In order to get rid of these oscillations, WENO reconstructions will be used to compute the first-order derivatives in time.

Given the point values of a function f at a stencil of $2p+1$ points:

$$S_i = \{f_{i-p}, \dots, f_{i+p}\},$$

where $f_j = f(x_j)$, WENO operators provide a reconstruction of f at

$$x_{i+1/2} = x_i + \frac{h}{2},$$

where h is the step of the mesh (assumed to be constant). This reconstruction is based on the Lagrange interpolation polynomials $p_s(x)$, $0 \leq s \leq p$ that interpolates the point values at $p + 1$ sub-stencils

$$S_{p,s} = \{f_{i-p+s}, \dots, f_{i+s}\}, \quad s = 0, \dots, p.$$

More precisely, the WENO strategy consists in defining the reconstruction as a convex combination

$$q(x_{i+1/2}) = \sum_{s=0}^p w_s p_s(x_{i+1/2}),$$

where the weights w_0, \dots, w_p satisfy $w_s \cong c_s$ on smooth zones, where c_0, \dots, c_p are the linear ideal weights satisfying

$$P(x_{i+1/2}) = \sum_{s=0}^p c_s p_s(x_{i+1/2}),$$

where $P(x)$ is the polynomial that interpolates all the point values of the stencil S_i . The weights w_i are function of some smoothness indicators. In FWENO methods introduced in [17], the following smoothness indicators have been proposed

$$I_s := \sum_{j=1}^p (f_{-p+i+s} - f_{-p-1+i+s})^2, \quad 0 \leq s \leq p, \quad (5.2.1)$$

that require a lower number of calculations than the smoothness indicators by Jiang and Shu (see [16]).

On the other hand, the expression of the weights in the original WENO method leads to an undesired loss of accuracy near critical points. To the best of our knowledge the only approach that allows to unconditionally attain the optimal order of accuracy regardless of the order of critical points is, for third-order reconstructions, the OWENO3 method introduced in [18] and, for reconstructions of order higher than 3, the OWENO methods presented in [19]. In this latter reference, the Jiang-Shu smoothness indicators are used to define the weights (for third-order methods these indicators coincide with (5.2.1)).

Let us summarize here the expression of FOWENO methods (see [18] and [19] for the accuracy analysis). The expression of FOWENO3, (i.e. OWENO3) is the following:

Given i and $\varepsilon > 0$,

1. Increase the dependence data stencil

$$\bar{S} = \{f_{i-1}, f_i, f_{i+1}, f_{i+2}\}, \quad (5.2.2)$$

with $f_i = f(x_i)$.

2. Compute the corresponding interpolating polynomials evaluated at $x_{i+1/2}$, which, both in case of reconstructions from point values and from cell averages, are given by

$$p_0(x_{i+1/2}) = -\frac{1}{2}f_{i-1} + \frac{3}{2}f_i, \quad p_1(x_{i+1/2}) = \frac{1}{2}f_i + \frac{1}{2}f_{i+1}. \quad (5.2.3)$$

3. Compute the corresponding Jiang-Shu smoothness indicators I_0 , I_1 and I_2 (including the one considering the rightmost node) by

$$I_0 = (f_i - f_{i-1})^2, \quad I_1 = (f_{i+1} - f_i)^2, \quad I_2 = (f_{i+2} - f_{i+1})^2. \quad (5.2.4)$$

4. Compute the preliminary weights $\tilde{\omega}_0$ and $\tilde{\omega}_1$:

$$\tilde{\omega}_s := \frac{I_s + \varepsilon}{I_0 + I_1 + 2\varepsilon}, \quad s = 0, 1 \quad (5.2.5)$$

5. Define τ by

$$\tau := dI, \quad d := (-f_{i-1} + 3f_i - 3f_{i+1} + f_{i+2})^2, \quad I := I_0 + I_1 + I_2. \quad (5.2.6)$$

6. Compute the corrector weight ω :

$$\omega = \frac{J}{J + \tau + \varepsilon}, \quad \text{with } J = I_0(I_1 + I_2) + (I_0 + I_1)I_2. \quad (5.2.7)$$

7. Compute the corrected weights ω_0 and ω_1 :

$$\omega_0 := \omega c_0 + (1 - \omega)\tilde{\omega}_0, \quad \omega_1 := \omega c_1 + (1 - \omega)\tilde{\omega}_1, \quad (5.2.8)$$

where c_0, c_1 are the ideal linear weights.

8. Obtain the OWENO reconstruction at $x_{i+1/2}$:

$$q(x_{i+1/2}) = \omega_0 p_0(x_{i+1/2}) + \omega_1 p_1(x_{i+1/2}).$$

Unlike FOWENO3, FOWENO(2p+1) reconstructions for $p \geq 2$ do not require to increase artificially the stencil. Their expression, combined with the smoothness indicators (5.2.1) can be summarized as follows:

Given i , the stencil S_i and $\varepsilon > 0$.

1. Compute the interpolating polynomials p_j , $j = 0 \leq j \leq p$,
2. Compute the fast smoothness indicators (5.2.1).

3. Compute the discriminant

$$D_p = |B_p - 4A_p C_p|,$$

with

$$A_p = \frac{1}{2} \sum_{j=-p}^p \delta_{p,j}^{2p} f_{i+j}, \quad B_p = \sum_{j=-p}^p \delta_{p,j}^{2p-1} f_{i+j}, \quad C_p = \sum_{j=-p}^p \delta_{p,j}^{2p-2} f_{i+j}. \quad (5.2.9)$$

for $j = -p, \dots, p$.

4. Obtain the squared undivided difference of order $2p$:

$$\tau_p = (2A_p)^2. \quad (5.2.10)$$

5. Compute

$$d_p := \frac{\tau_p^{a_1} D_p^{a_1}}{\tau_p^{a_1} + D_p^{a_1} + \epsilon}$$

for some a_1 chosen by the user such that $a_1 \geq 1$, as done in [19].

6. Compute

$$\alpha_s = c_s \left(1 + \frac{d_p}{I_s^{a_1} + \epsilon} \right)^{a_2}, \quad 0 \leq s \leq p, \quad (5.2.11)$$

where c_s are the ideal linear weights. a_2 is chosen by the user such that $a_2 \geq \frac{p+1}{2a_1}$, which is a sufficient condition to attain the optimal $(p+1)$ -th accuracy near discontinuities [17].

7. Generate the FOWENO weights:

$$\omega_s = \frac{\alpha_s}{\alpha_0 + \dots + \alpha_p}, \quad s = 0, \dots, p. \quad (5.2.12)$$

8. Obtain the reconstruction at $x_{i+1/2}$:

$$q_p(x_{i+1/2}) = \sum_{s=0}^p \omega_s p_s(x_{i+1/2}). \quad (5.2.13)$$

Combining the results obtained in [17] and [19] one can see that this method attains the optimal order regardless of the order of the critical point, without having to artificially tune ϵ .

5.3 FOWENO-AT Methods

With the FOWENO spatial reconstructions already defined, we incorporate them in the Approximate Taylor methods to avoid the appearance of oscillations near the discontinuities or shocks, substituting the first derivative in time of the Taylor expansion by those reconstructions. More precisely, in LAT methods of Section (5.1.5) is replaced by:

$$\tilde{u}_{t,i}^{(1)} = -\frac{\hat{f}_{i+1/2} - \hat{f}_{i-1/2}}{\Delta x}. \tag{5.3.1}$$

where $\hat{f}_{i+1/2}$ denotes the $(2p+1)$ th-order FOWENO flux splitting reconstructions at $x_{i+1/2}$. In CAT methods, (5.1.13) is replaced by:

$$F_{i+1/2}^p = \hat{f}_{i+1/2} + \sum_{k=2}^m \frac{\Delta t^{k-1}}{k!} A_{p,0}^{0,1/2}(\tilde{f}_{i,*}^{(k-1)}, \Delta x). \tag{5.3.2}$$

FOWENO reconstructions are computed in conserved variables using the procedure described in [20], so that their extension to systems is straightforward.

5.4 Numerical experiments

In order to simplify the notation and save space for the labels, from now on the following abbreviations will be used for the different numerical methods to be compared:

Abbreviation	Numerical method
<i>WqRs</i>	WENO q with SSPRK s
<i>WqCs</i>	WENO q with CAT s
<i>WqLs</i>	WENO q with LAT s
<i>FOWqRs</i>	FOWENO q with SSPRK s
<i>FOWqCs</i>	FOWENO q with CAT s
<i>FOWqLs</i>	FOWENO q with LAT s

Here, SSPRK denotes the well-known Strong Stability Preserving Runge-Kutta methods [25], q is the order of accuracy of the spatial WENO reconstructions and s is the order of accuracy of the time discretization. We present some numerical experiments using FOWENO and the traditional WENO [20] reconstructions combined with CAT{2, 4, 6}, LAT{3, 5, 7} and SSPRK{3, 4} over some classical 1D scalar conservation laws (linear transport and Burgers equations) and 1D and 2D systems (Euler equations of gas dynamics).

5.4.1 Scalar conservation laws

Let us consider first the one-dimensional scalar conservation law:

$$u_t + f(u)_x = 0. \quad (5.4.1)$$

5.4.1.1 Test 5.1 Transport equation

We consider (5.4.1) with linear flux function $f(u) = au$ in the spatial interval $x \in [0, 2]$ with initial condition:

$$u(x, 0) = \begin{cases} e^{-1200(x-1/3)^2} & 0 \leq x < 2/3, \\ 6(x - 2/3) & 2/3 \leq x < 5/6, \\ -6(x - 1) & 5/6 \leq x < 1, \\ 1 & 7/6 \leq x \leq 4/3, \\ \sqrt{1 - 100(x - 5/3)^2} & 3/4 < x \leq 2. \end{cases} \quad (5.4.2)$$

Figures 5.1, 5.2, 5.3 and 5.4 show the results obtained with the methods W3R3, W3C2, W3L3, W5R3, W5C4, W5L5, W7R4, W7C6, W7L7, FOW3R3, FOW3C2, FOW3L3, FOW5R3, FOW5C4, FOW5L5, FOW7R4, FOW7C6, and FOW7L7 at time $t = 2$. using a 200-point mesh, $a = 1$, periodic boundary conditions, and $\text{CFL} = \{0.5, 0.9\}$. This test is a slight modification of the one proposed by Jiang and Shu in [16].

From these plots we can conclude:

For $\text{CFL} = 0.5$

- Third-order reconstructions (Figure 5.1): FOWENO reconstructions give better results than WENO reconstructions in all cases. We stress the fact that, in spite of its lower order of accuracy, CAT2 gives very good results particularly when combined with FOW3 reconstruction: see enlarged views.
- Fifth-order reconstructions (Figure 5.2): SSPRK3 gives worse results than CAT4 and LAT5 in the two first areas of interest with both WENO5 and FOWENO5. While CAT4 and LAT5 give similar results when combined with W5, LAT5 gives better results for FOWENO5: see enlarged views.
- Seventh-order reconstructions (Figure 5.3) : WENO and FOWENO SSPRK4 give solutions that are slightly better than those given by CAT6 and LAT7.

For $\text{CFL} = 0.9$.

- Fifth-order reconstructions (Figure 5.4): LAT5 methods are not stable for this CFL value, and SSPRK4 methods give oscillatory solution, especially near discontinuities. CAT4 combined with FOWENO5 is stable and gives very good solutions: see enlarged views.

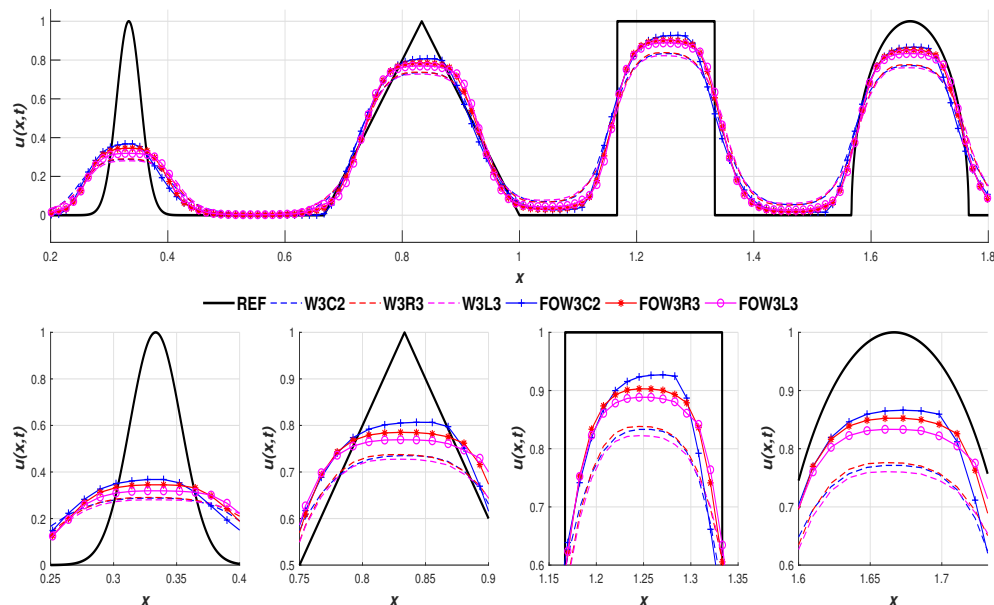


Figure 5.1: Test 5.1. Transport equation with initial conditions (5.4.2), CFL= 0.5 and $t = 2$ s. Methods based on 3rd-order reconstructions: general view (top) and zoom of the areas of interest (bottom).

Table 5.1 shows the CPU times corresponding to the different methods for $t = 2$. and CFL= 0.5. The values are obtained by averaging the computational cost of ten runs. The entries of the table show the ratio between the computational time of each method and the corresponding to W5R3 which is the reference.

FOW3C2	FOW3L3	FOW3R3	W3C2	W3L3	W3R3
0.3695	0.4509	0.8351	0.3742	0.734	0.6468
FOW5C4	FOW5L5	FOW5R3	W5C4	W5L5	W5R3
1.0546	0.7540	0.9980	1.1936	0.7589	1
FOW7C6	FOW7L7	FOW7R4	W7C6	W7L7	W7R4
2.5049	1.1818	4.4116	3.4330	1.715	5.1513

Table 5.1: CPU time ratios for Test 5.1: linear transport equation with initial conditions (5.4.2), CFL= 0.5, and $t = 2$.

The following conclusions can be drawn:

- The cheapest method is FOW3C2 (that is only second-order accurate in time) and the most expensive is W7R4 (due to the extra cost of the smoothness indicators and to the 10 stages of SSPRK4).

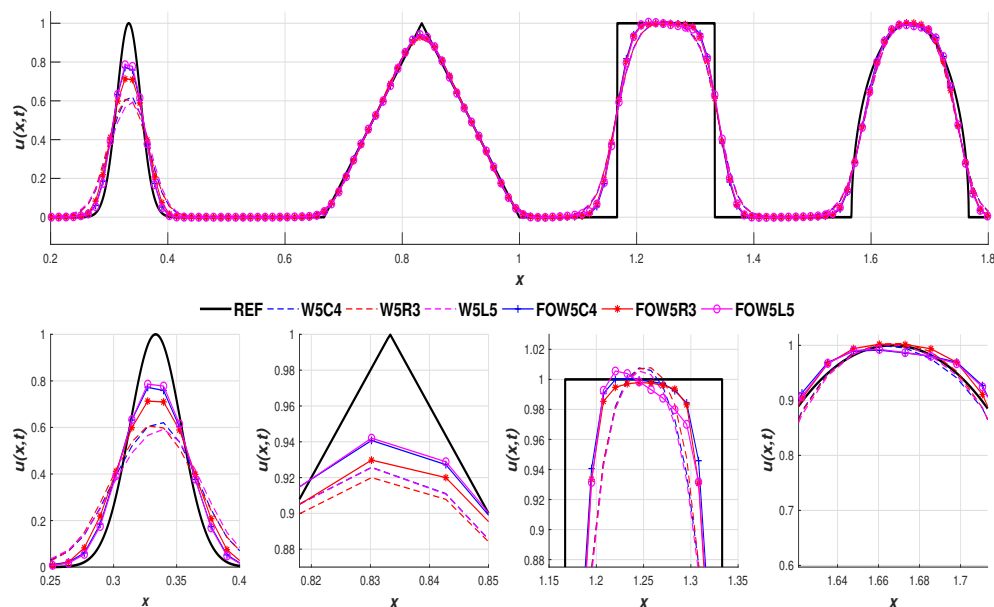


Figure 5.2: Test 5.1. Transport equation with initial conditions (5.4.2), CFL= 0.5 and $t = 2$ s. Methods based on 5th-order reconstructions: general view (*top*) and zoom of the areas of interest (*bottom*).

- Methods based on WENO reconstructions are more costly than their corresponding FOWENO counterparts with the only exception of FOW3R3 and W3R3. Moreover the differences increase with the order.
- Methods based on CATs are more costly than their $LAT(s + 1)$ counterparts with the only exception of CAT2. The differences increase with the order. Nevertheless, this extra cost is compensated by the better stability properties of CAT methods for CFL values bigger than 0.5.

5.4.1.2 Test 5.2 Burgers equation

Let us consider now Burgers equation i.e. (5.4.1) with $f(u) = u^2/2$, in the spatial interval $[0, 1]$ with initial condition

$$u(x, 0) = e^{-10(x-1/2)^2}. \quad (5.4.3)$$

Figure 5.5 shows the numerical solutions obtained with W3R3, W3C2, W3L3, W5R3, W5C4, W5L5, W7R4, W7C6, W7L7, FOW3R3, FOW3C2, FOW3L3, FOW5R3, FOW5C4, FOW5L5, FOW7R4, FOW7C6 and FOW7L7 methods using a 160-point mesh, periodic boundary conditions, CFL= 0.5, and $t = 2$ s. The numerical results are shown

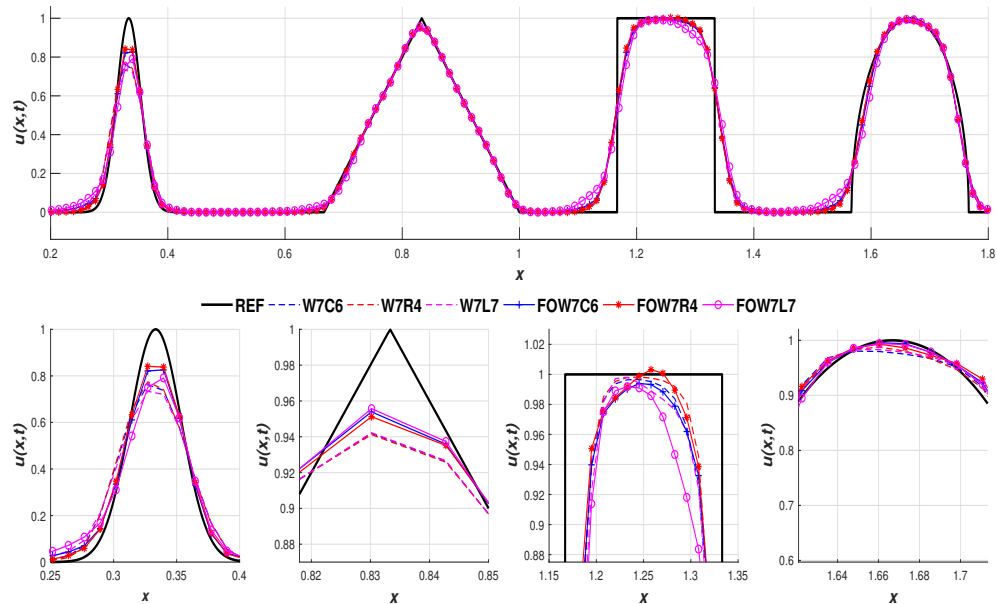


Figure 5.3: Test 5.1 Transport equation with initial conditions (5.4.2), and $t = 2s$. Methods based on 7th-order reconstructions with $CFL= 0.5$: general view (*top*) and zoom of the areas of interest (*bottom*).

in groups of three to facilitate the comparisons. From the enlarged views (close to the shock) the following conclusions can be drawn:

- Methods based on third-order reconstructions (Figure 5.5 row 2): all the methods based on WENO3 give essentially the same solutions. Some improvements are achieved with FOWENO3 and CAT2 is slightly sharper than the rest.
- Methods based on fifth-order reconstructions (Figure 5.5 row 3): the results are better than the ones corresponding to third-order reconstructions as expected. There are no big differences between them, but a slight improvement can be observed when FOWENO reconstructions are used.
- Methods based on seventh-order reconstructions (Figure 5.5 row 4): WENO7 and FOWENO7 reconstructions give non-oscillatory solutions and better results than third or fifth-order reconstructions for CAT6 and RK4, which is not the case for LAT7.

Concerning the quality of the numerical results with $CFL= 0.9$ or the computational cost, we draw similar conclusions to the previous test case.



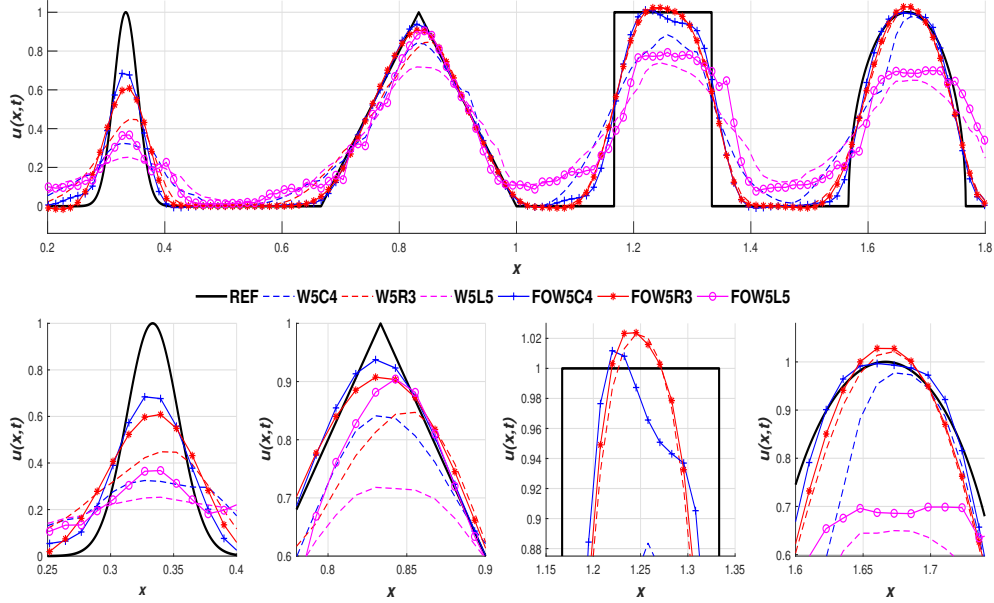


Figure 5.4: Test 5.1. Transport equation with initial conditions (5.4.2), and $t = 2$ s. Methods based on 5th-order reconstructions with $CFL = 0.9$: general view (*top*) and zoom of the areas of interest (*bottom*).

5.4.2 1D Systems of conservation laws

We consider the 1D Euler equations of gas dynamics:

$$\mathbf{w}_t + \mathbf{f}(\mathbf{w})_x = \mathbf{0}, \quad (5.4.4)$$

where

$$\mathbf{w} = \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}, \quad \mathbf{f}(\mathbf{w}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(E + p) \end{pmatrix}.$$

Here, ρ is the density, u the velocity, E the total energy per unit volume and p the pressure. We assume an ideal gas with the equation of state

$$p(\rho, e) = (\gamma - 1)\rho e,$$

where γ is the ratio of specific heat capacities of the gas and e the internal energy per unit mass given by:

$$E(\rho, u, e) = \rho(e + \frac{1}{2}u^2).$$

We consider the following 1D Riemann problems whose data are given in Table 5.2:

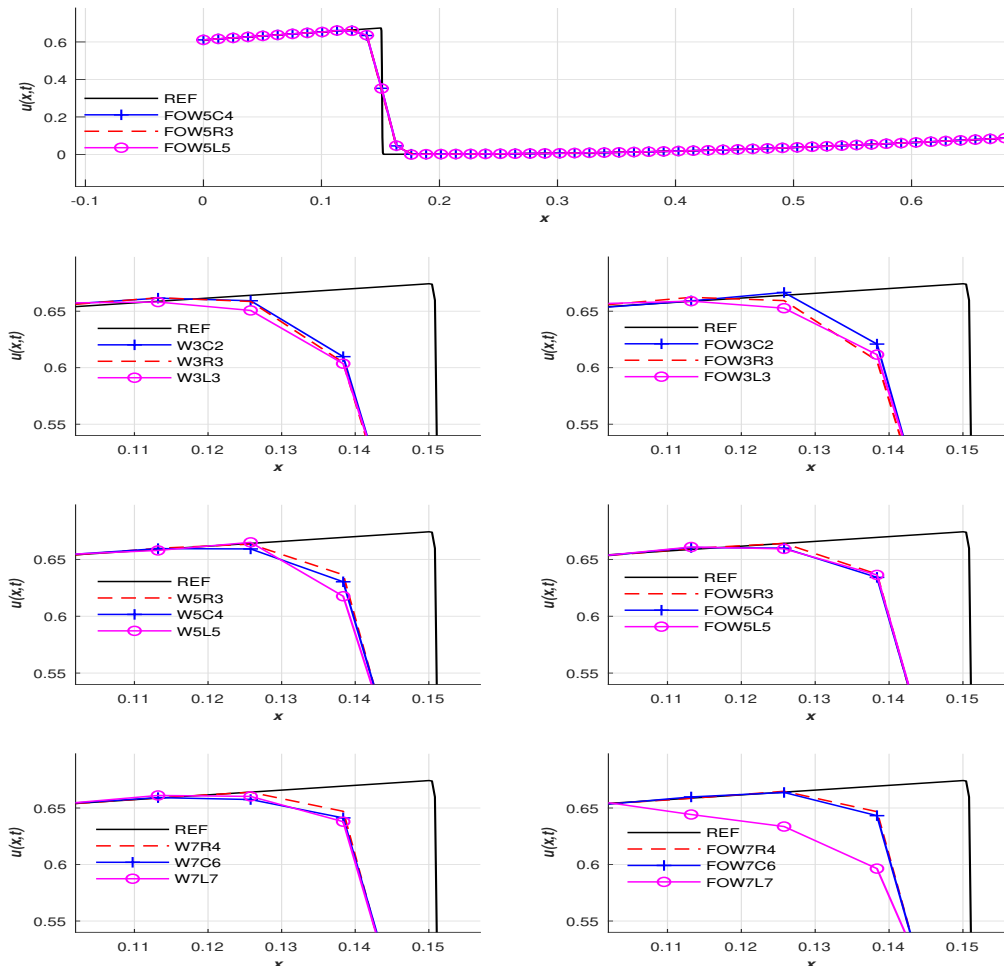


Figure 5.5: Test 5.2. Burgers equation with initial conditions (5.4.3), CFL= 0.5 and $t = 2s$. Row 1: methods based on 5th-order reconstructions: general view. Rows 2-4: zooms of an area of interest.

5.4.2.1 Test 5.3 Sod shock tube problem

The solution of this problem consists of a left rarefaction, a left contact and a right shock. More details in [61].

5.4.2.2 Test 5.4 123 Einfeldt problem

The solution consists of two strong rarefactions and a stationary contact discontinuity. The pressure p is small (close to vacuum). More details in [64]

5.4.2.3 Test 5.5 Left half of the blast wave problem

The solution contains a left rarefaction, a contact and a right shock. More details in [65].

5.4.2.4 Test 5.6 Right half of the blast wave problem

The solution contains a left shock, a contact discontinuity and a right rarefaction. More details in [65].

5.4.2.5 Test 5.7 Blast wave problem

The solution represents the collision of the right and left shocks corresponding to tests 3 and 4, and consists of a left facing shock (travelling very slowly to the right), a right contact discontinuity and a right shock wave. More details in [65].

The equations are solved in the spatial domain $x \in [0, 1]$ with outflow-inflow boundary conditions and a 200-point mesh. CFL= 0.9, 0.5, 0.25 are used for methods based on with 3rd, 5th, and 7th-order reconstructions respectively. We consider WENO reconstructions with $\epsilon = 1e - 6$ as in [20] and FOWENO reconstructions with $\epsilon = 1e - 100$ as in [17]. The numerical solutions are compared against the exact solution provided by the HE-E1RPEXACT solver introduced in [3]

Test	ρ_L	u_L	p_L	ρ_R	u_R	p_R	time (sec.)
3	1.0	0.0	1.0	0.125	0.0	0.1	0.25
4	1.0	-2.0	0.4	1.0	2.0	0.4	0.15
5	1.0	0.0	1000.0	1.0	0.0	0.01	0.012
6	1.0	0.0	0.01	1.0	0.0	100.0	0.035
7	.99924	19.5975	460.894	5.99242	-6.19633	46.0950	0.035

Table 5.2: Riemann problems for 1D Euler equations.

The numerical results are shown in Figures 5.6-5.15. Two figures are shown for every test case, the first one corresponds to densities and the second one to internal energies. In the first row of the figures corresponding to the densities, we show the global views of the reference and the numerical solutions obtained using third and fifth-order reconstructions. Rows 2-4 show enlarged views of the areas of interest labelled a , b and c in the global view of the reference solution. In the figures corresponding to internal energy we plot global views of the numerical results for third, fifth and seventh-order reconstructions (left column) and enlarged views of an interest area of each one of them (right column).

- **Test 5.3:** Figures 5.6 and 5.7. In general, all the solutions are acceptable and their quality improve with the order of accuracy. Methods based on FOWENO

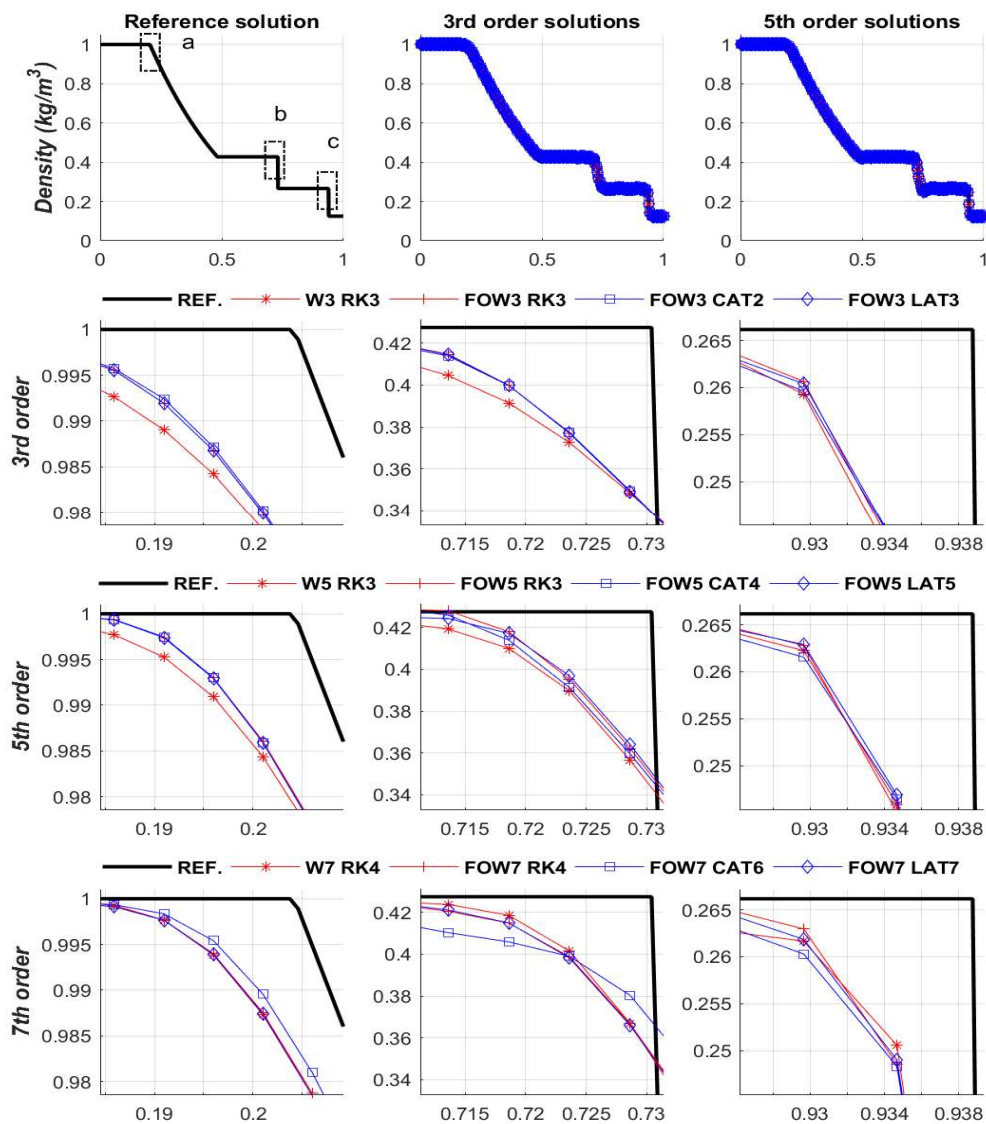


Figure 5.6: Test 5.3. 1D Euler equations. Sod problem: density. Row 1: exact solution (left), methods using 3rd-order (center) and 5th-order (right) reconstruction operators. Rows 2-4: zooms corresponding to areas *a*, *b* and *c*. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

reconstruction are slightly sharper than those based on WENO with exception of FOW7C6 near the contact discontinuity (the approximation obtained of this wave

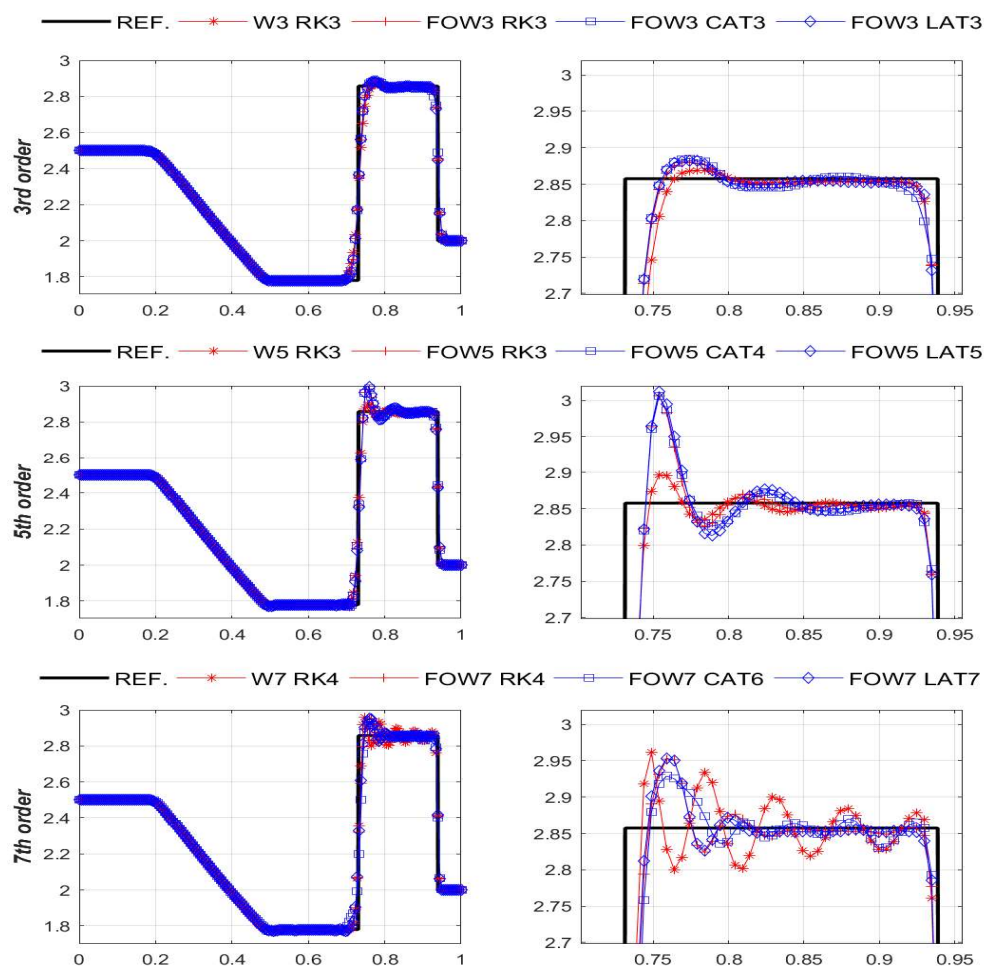


Figure 5.7: Test 5.3. 1D Euler equations. Sod problem: internal energy. Methods using 3rd order (*row 1*), 5th order (*row 2*), and 7th order (*row 3*) reconstruction operators. Left: general view. Right: zoom of an area of interest. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th order reconstructions respectively.

is worse but oscillations appear, even for long-time simulation). Concerning the internal energies, solutions obtained with LAT and CAT are less oscillatory: see the enlarged views.

- **Test 5.4:** Figures 5.8 and 5.9. This is a hard test in which significant differences between WENO and FOWENO reconstructions can be seen. For densities, FOW3C2 and FOW3L3 give the closest solutions to the reference in area *b*.

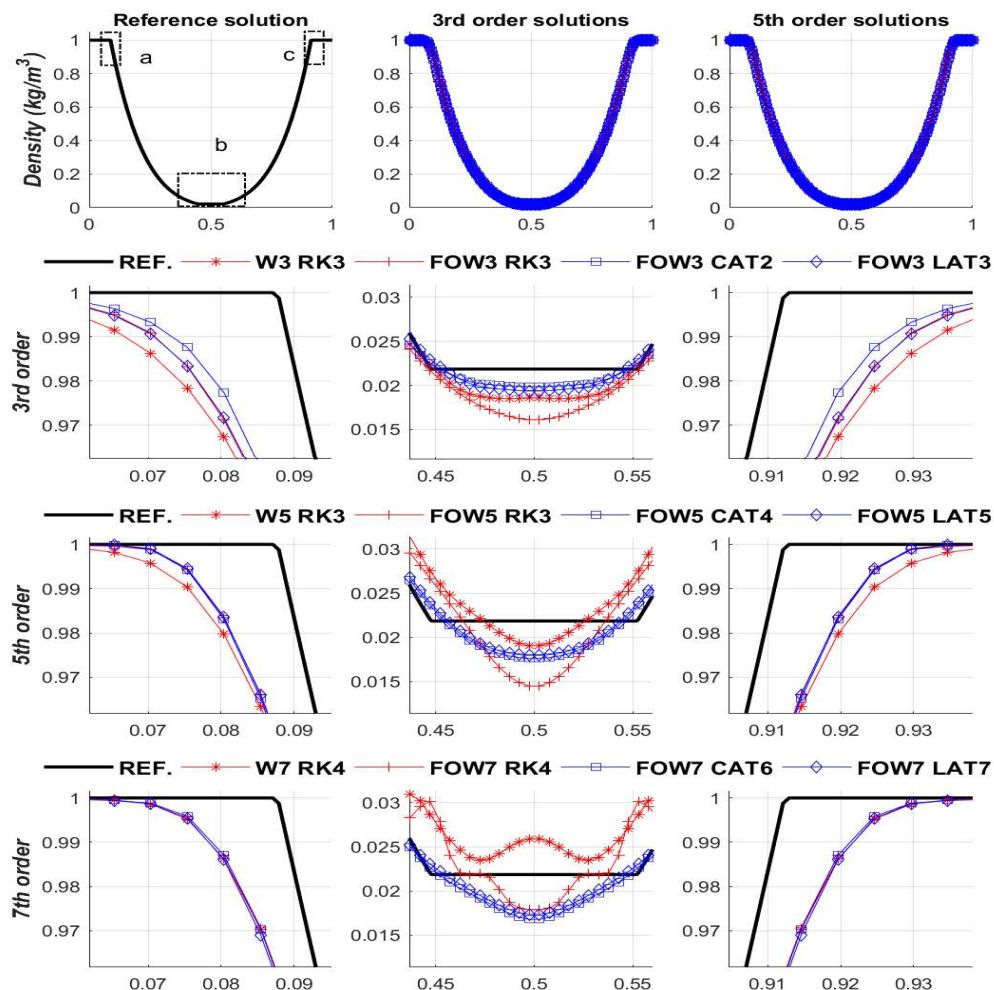


Figure 5.8: Test 5.4. 1D Euler equations. 123 Einfeldt problem: density. Row 1: exact solution (*left*), methods using 3rd (*center*) and 5th order (*right*) reconstruction operators. Rows 2-4: zooms corresponding to areas *a*, *b* and *c*. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

Moreover, all FOWENO-AT solutions are stable and non-oscillatory. For internal energies, solutions corresponding to WENO methods show oscillations but they are closer to the exact solution.

- **Test 5.5:** Figures 5.10 and 5.11. 3rd-order accuracy is not enough in this case to capture good solutions, especially in area *c*. FOW5CAT4 and FOW5LAT5 give

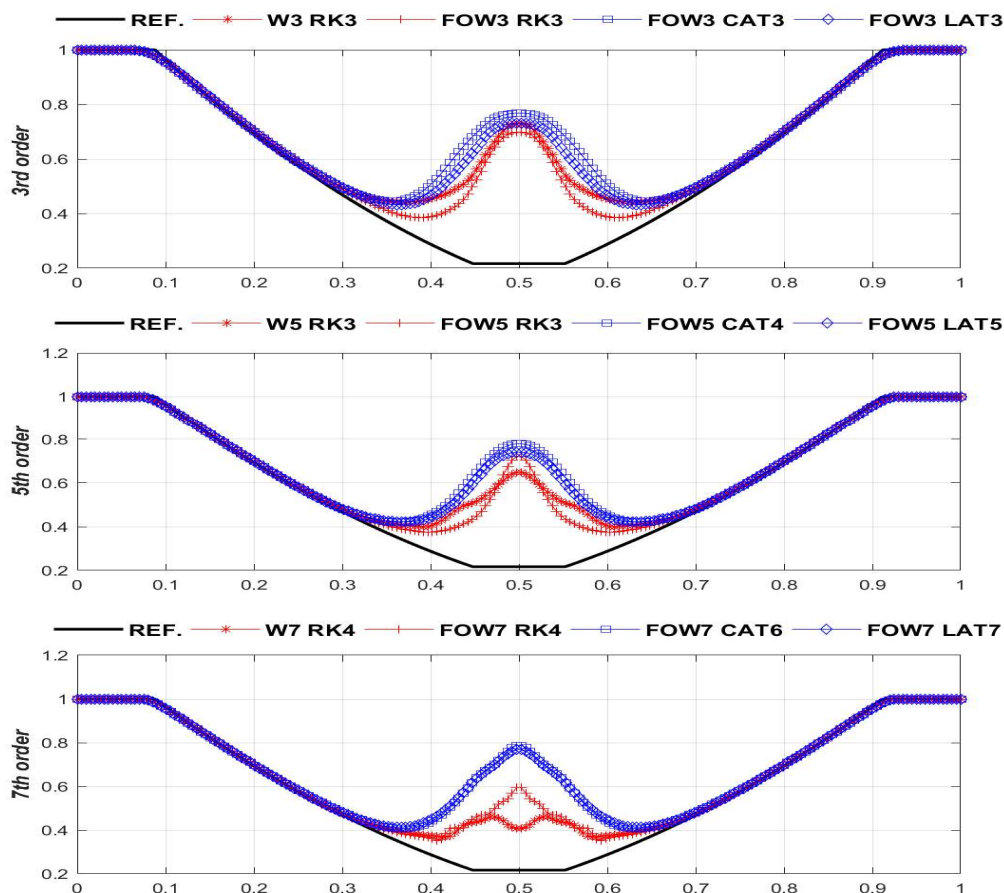


Figure 5.9: Test 5.4. 1D Euler equations. 123 Einfeldt problem: internal energy. Methods using 3rd-order (row 1), 5th-order (row 2), and 7th-order (row 3) reconstruction operators. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

better solutions than W5R3, which is under dissipative. However, for 7th-order reconstruction the situation is the opposite, due to the use of SSPRK_10_4 for WENO7. For internal energies, no significant differences are detected.

- **Test 5.6:** Figures 5.12 and 5.13. Similar conclusions to Test 5.5.
- **Test 5.7:** Figures 5.14 and 5.15. In order to compare the cpu times, CFL= 0.25 has been chosen for all the methods. Methods based on 7th-order reconstructions give the best approximations in areas *a* and *c* but produce some oscillations in area *b*. These oscillations are particularly noticeable in the top part of the internal energy

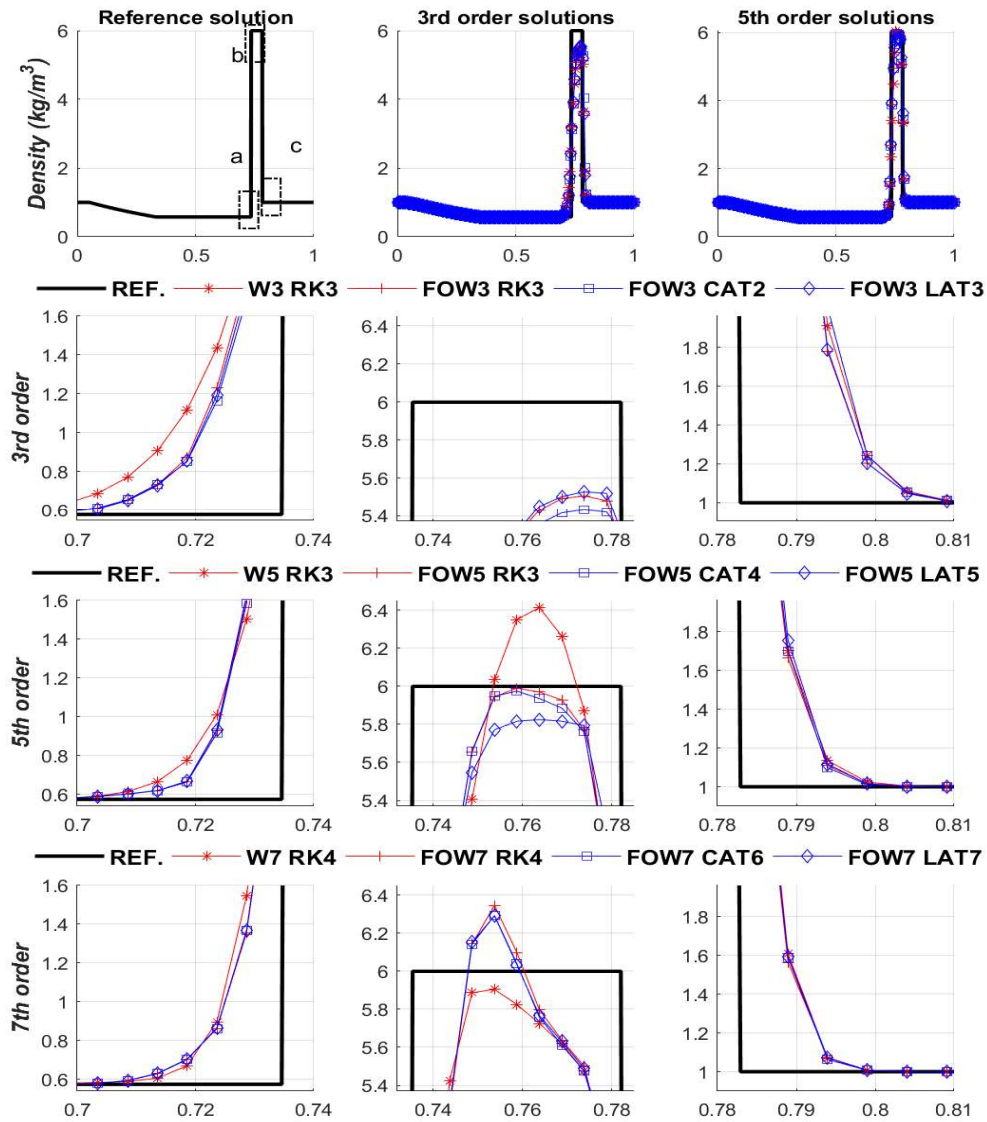


Figure 5.10: Test 5.5. 1D Euler equations. Left half of the blast wave problem of Woodward and Colella: density. Row 1: exact solution (*left*), methods using 3rd (*center*) and 5th-order (*right*) reconstruction operators. Rows 2-4: zooms corresponding to areas *a*, *b* and *c*. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

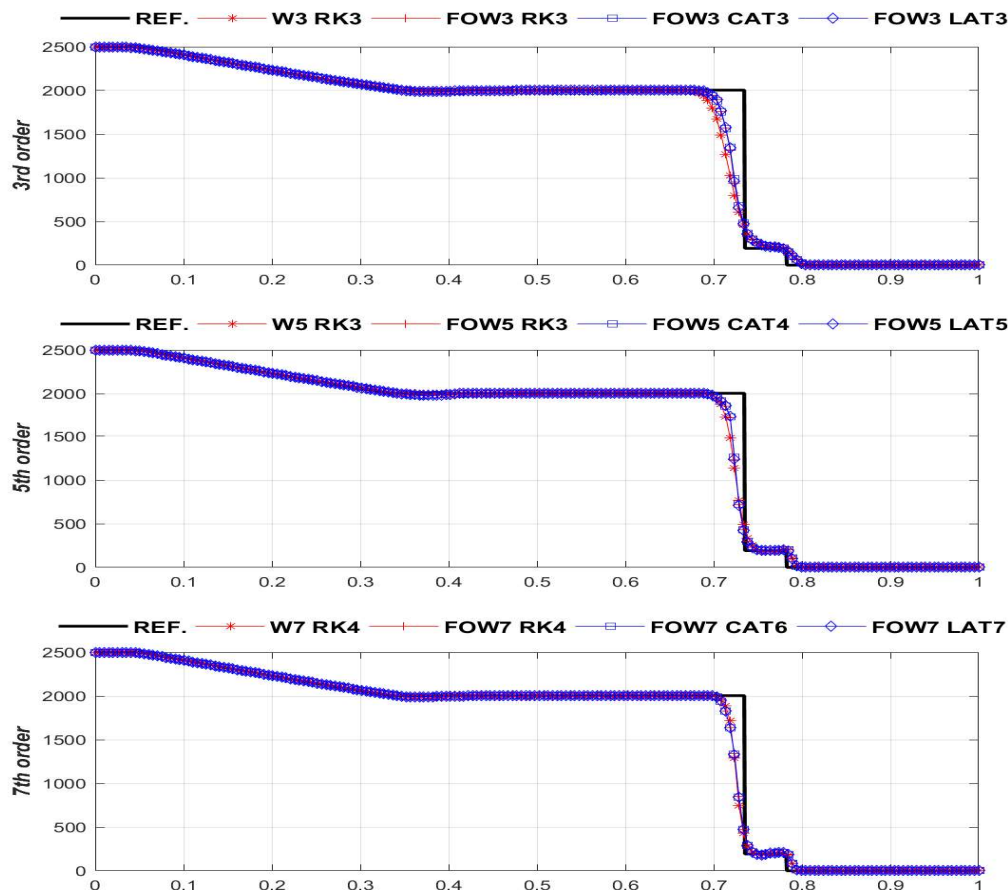


Figure 5.11: Test 5.5. 1D Euler equations. Left half of the blast wave problem of Woodward and Colella: internal energy. Methods using 3d-order (*row 1*), 5th-order (*row 2*), and 7th-order (*row 3*) reconstruction operators. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

solutions, in which the solutions provided by AT methods are less oscillatory. CPU times are shown in Table 5.3. WENO3-CAT2 (which is the faster method) is the reference. Some conclusions can be drawn from this table:

1. 3rd-order methods based on WENO are cheaper than FOMENO3: in this case the smooth indicators are the same and FOWENO has the extra computational cost due to the computation of the optimal weights.
2. For reconstructions of order 5 or greater, methods based on FOWENO are faster than those based on WENO.

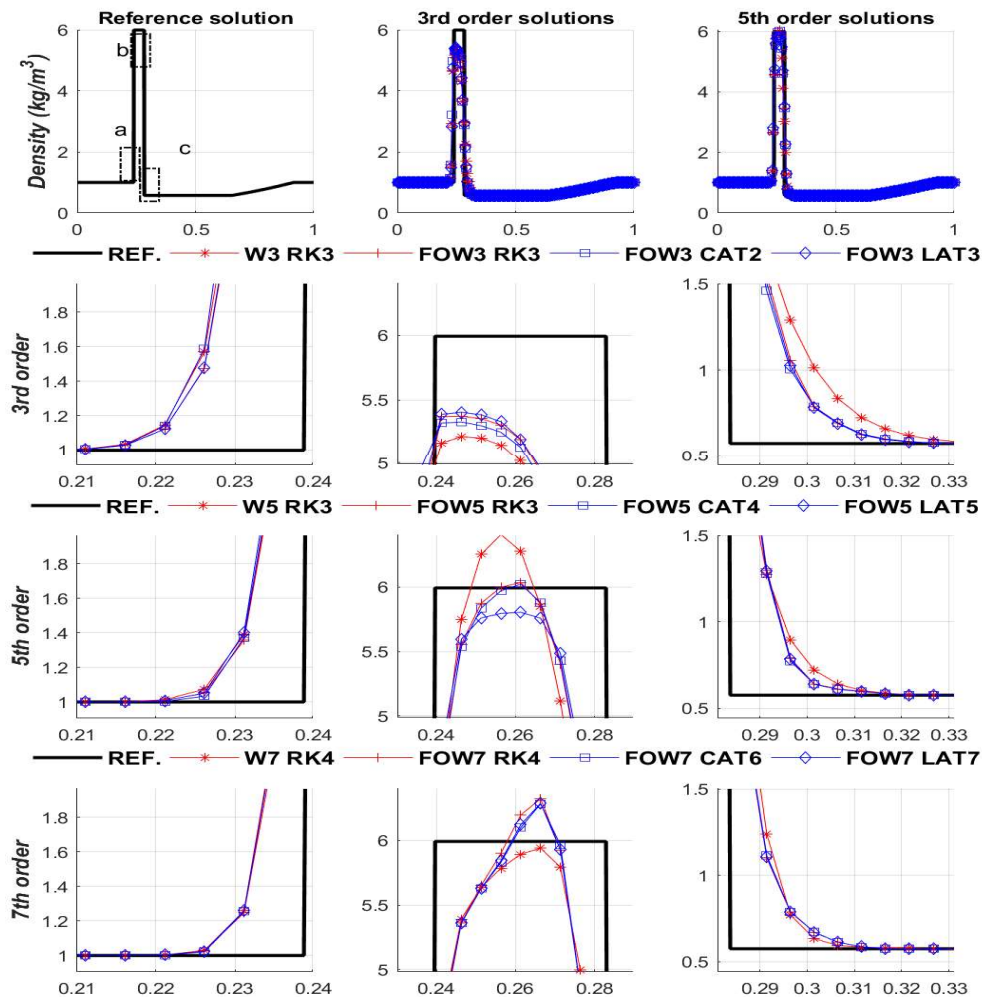


Figure 5.12: Test 5.6. 1D Euler equations. Right half of the blast wave problem of Woodward and Colella: density. Row 1: exact solution (*left*), methods using 3rd-order (*center*) and 5th-order (*right*) reconstruction operators. Rows 2-4: zooms corresponding to areas *a*, *b* and *c*. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

3. To pass from CAT2 to CAT4 using the same reconstruction operator multiplies the computational time approximately by 3. And to pass from CAT4 to CAT6 by a factor between 4 and 6.
4. To pass from LAT3 to LAT5 using the same reconstruction operator multiplies

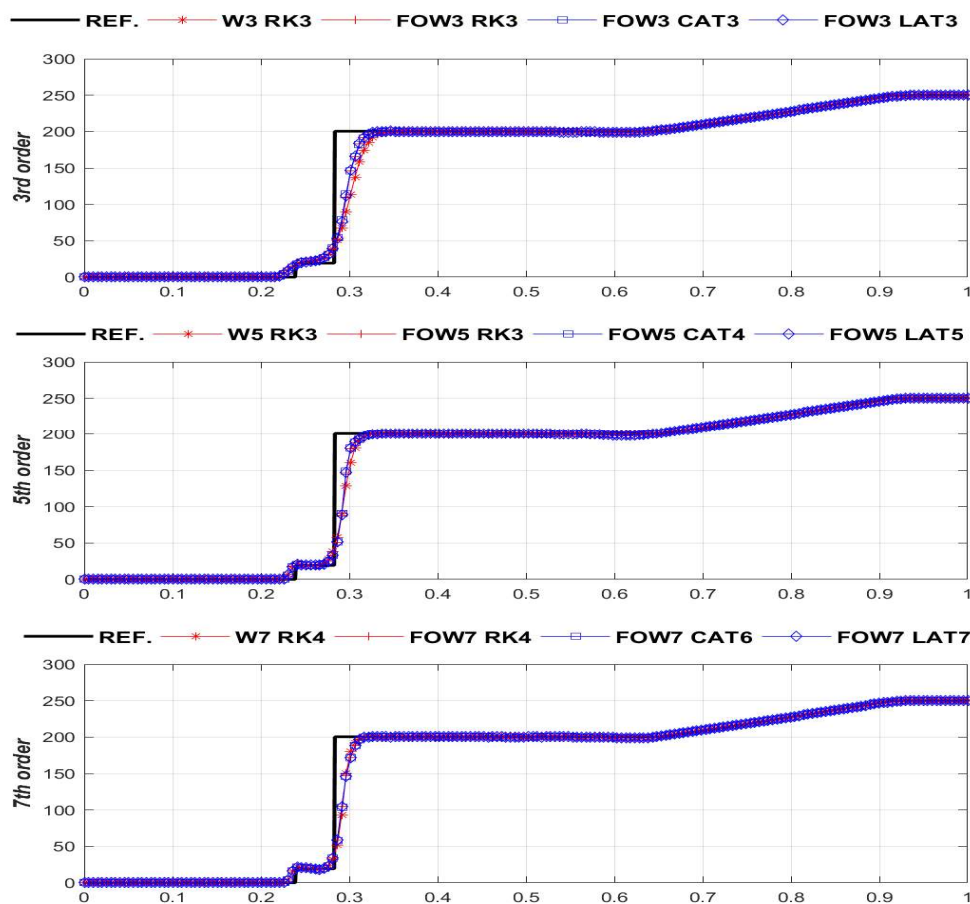


Figure 5.13: Test 5.6. 1D Euler equations. Right half of the blast wave problem of Woodward and Colella: internal energy. Methods using 3rd-order (*row 1*), 5th-order (*row 2*) and 7th-order (*row 3*) reconstruction operators. Left: general view. Right: zoom of an area of interest. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

the computational time approximately by 5. And to pass from LAT5 to LAT7 by a factor between 6 and 7.

- To pass from RK3 (SSPRK 3-3, i.e. third-order and 3 stages) to RK4 (SSPRK 4-10, i.e. fourth-order and 10 stages) the computational time approximately by a factor between 6 and 8.5.

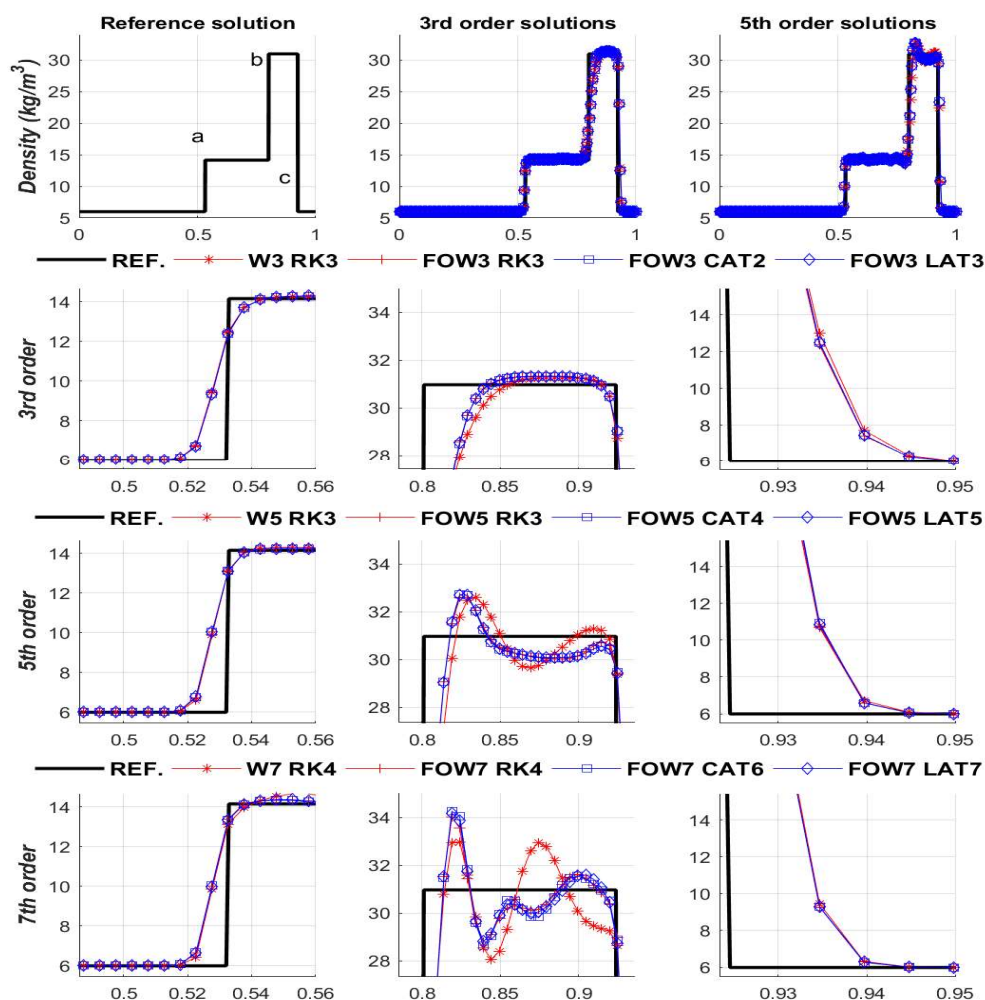


Figure 5.14: Test 5.7. 1D Euler equations. Woodward and Colella problem: density. Row 1: exact solution (*left*), methods using 3rd-order (*center*) and 5th-order (*right*) reconstruction operators. Rows 2-4: zooms corresponding to areas *a*, *b* and *c*. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

5.4.3 2D Systems of conservation laws

We consider now the two-dimensional Euler equations of gas dynamics:

$$\mathbf{w}_t + \mathbf{f}(\mathbf{w})_x + \mathbf{g}(\mathbf{w})_y = \mathbf{0} , \quad (5.4.5)$$



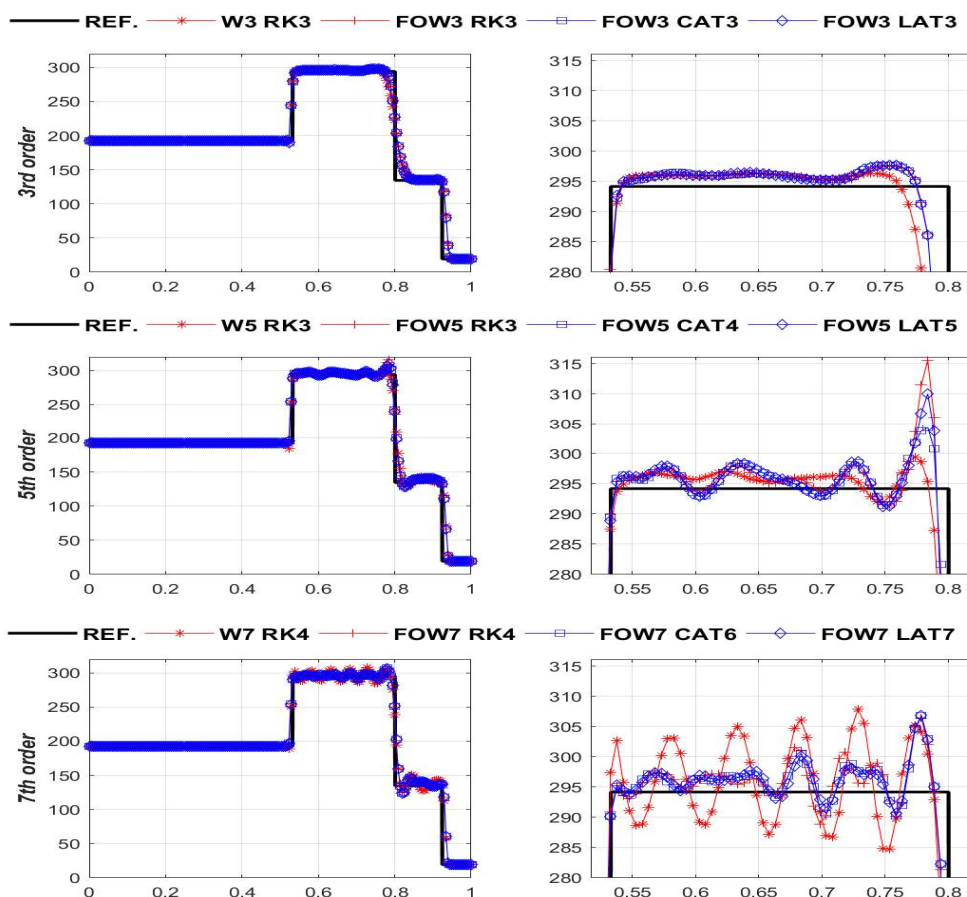


Figure 5.15: Test 5.7. 1D Euler equations. Woodward and Colella problem: internal energy. Methods using 3rd-order (row 1), 5th-order (row 2) and 7th-order (row 3) reconstruction operators. Left: general view. Right: zoom of an area of interest. Exact solution: black line. CFL= 0.9, 0.5, 0.25 for methods based on with 3rd, 5th, and 7th-order reconstructions respectively.

where

$$\mathbf{w} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}, \quad \mathbf{f}(\mathbf{w}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E + p) \end{pmatrix}, \quad \mathbf{g}(\mathbf{w}) = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v(E + p) \end{pmatrix}.$$

ρ is again the density; u, v are the components of the velocities in the x, y directions respectively; E , the total energy per unit volume; and p , the pressure. The equation of

FOW3C2	FOW3L3	FOW3R3	W3C2	W3L3	W3R3
1.1830	1.6352	2.8026	1.0000	1.3744	2.1764
FOW5C4	FOW5L5	FOW5R3	W5C4	W5L5	W5R3
5.0546	3.4400	3.2980	5.1642	3.7589	3.5268
FOW7C6	FOW7L7	FOW7R4	W7C6	W7L7	W7R4
23.8827	18.1818	19.7516	29.9430	22.7150	29.9490

Table 5.3: CPU time ratios for test 5.7. 1D Euler equations with the Woodward and Colella problem, CFL= 0.25, and $t = 0.035s$.

state

$$p(\rho, u, v, E) = (\gamma - 1) \left(E - \frac{\rho}{2}(u^2 + v^2) \right), \quad (5.4.6)$$

is assumed again where γ is the ratio of specific heat capacities of the gas.

From the nineteen configurations of the 2-D Riemann problems presented in [66] six relevant configurations have been selected, namely: 3, 6, 11, 13, 17 and 19. The initial data of the Riemann problems consist of constant states at every quadrant of the spatial domain that are chosen so that the 1D Riemann problems corresponding to two adjacent states consist of only one one-dimensional simple wave: a shock S , a rarefaction wave R , or a slip line i.e. a contact discontinuity with discontinuous tangential velocity J . The sub-indexes $(l, r) \in \{(2, 1), (3, 2), (3, 4), (4, 1)\}$ indicate the involved quadrants. For shock and rarefactions an over-arrow indicate the direction (backward or forward). And for contact discontinuities a sign $+/-$ is used (instead of the over-arrow), to denote whether it is a positive or negative slip line. Full information and analysis can be found in [66].

The methods are run in a 400×400 point mesh of the computational domain $[0, 1] \times [0, 1]$ with CFL= 0.475 and outflow-inflow boundary conditions. Lax-Friedrichs flux-splitting is used in both WENO and FOWENO implementations. Figures 5.16 to 5.22 show the numerical densities obtained for the Lax configurations 3, 6, 11, 13, 17 and 19, respectively. Only the numerical solutions obtained with methods based on FOWENO reconstructions of order 3 or 5 are plotted with the exception of Test 9 for which the solutions given by methods based on WENO reconstructions are also plotted for comparison. Plots are made in Matlab with 25 contour lines.

5.4.3.1 Test 5.8 - 5.13 Euler equations

In all cases methods based on third-order reconstructions give similar solutions to those provided in [67], even for FOW3C2 in spite of its lower order of accuracy in time. Qualitatively, no significant differences between the results obtained using CAT2 or LAT3 are detected. Methods based on fifth-order reconstructions are sharper in all cases, as expected. The quality of the solutions obtained with CAT and LAT are mostly identical again. A comparison between Figures 5.17 and 5.18 makes noticeable the improvements

Test	Lax 5.8 Configuration 3						
$p_2 = 0.3$	$\rho_2 = 0.5323$	$p_1 = 1.5$	$\rho_1 = 1.5$				
$u_2 = 1.206$	$v_2 = 0$	$u_1 = 1$	$v_1 = 0$		$\overleftarrow{S}_{2,1}$		
$p_3 = 0.029$	$\rho_3 = 0.138$	$p_4 = 0.3$	$\rho_4 = 0.5323$	$\overleftarrow{S}_{3,2}$			$\overleftarrow{S}_{4,1}$
$u_3 = 1.206$	$v_3 = 1.206$	$u_4 = 0$	$v_4 = 1.206$		$\overleftarrow{S}_{3,4}$		
Test 5.9	Lax Configuration 6						
$p_2 = 1$	$\rho_2 = 2$	$p_1 = 1$	$\rho_1 = 1$				
$u_2 = 0.75$	$v_2 = 0.5$	$u_1 = 0.75$	$v_1 = -0.5$		$J_{3,2}^+$	$J_{2,1}^-$	
$p_3 = 1$	$\rho_3 = 1$	$p_4 = 1$	$\rho_4 = 3$				$J_{4,1}^+$
$u_3 = -0.75$	$v_3 = 0.5$	$u_4 = -0.75$	$v_4 = -0.5$			$J_{3,4}^-$	
Test 5.10	Lax Configuration 11						
$p_2 = 0.4$	$\rho_2 = 0.5313$	$p_1 = 1$	$\rho_1 = 1$				
$u_2 = 0.8275$	$v_2 = 0$	$u_1 = 0.1$	$v_1 = 0$			$\overleftarrow{S}_{2,1}$	
$p_3 = 0.4$	$\rho_3 = 0.8$	$p_4 = 0.4$	$\rho_4 = 0.5313$	$J_{3,2}^+$			$\overleftarrow{S}_{4,1}$
$u_3 = 0.1$	$v_3 = 0$	$u_4 = 0.1$	$v_4 = 0.7276$			$J_{3,4}^+$	

Test 5.11	Lax Configuration 13						
$p_2 = 1$	$\rho_2 = 2$	$p_1 = 1$	$\rho_1 = 1$				
$u_2 = 0$	$v_2 = 0.3$	$u_1 = 0$	$v_1 = -0.3$			$J_{2,1}^-$	
$p_3 = 0.4$	$\rho_3 = 1.0625$	$p_4 = 0.4$	$\rho_4 = 0.5313$	$\overleftarrow{S}_{3,2}$			$\overleftarrow{S}_{4,1}$
$u_3 = 0$	$v_3 = 0.8145$	$u_4 = 0$	$v_4 = 0.4276$			$J_{3,4}^-$	
Test 5.12	Lax Configuration 17						
$p_2 = 1$	$\rho_2 = 2$	$p_1 = 1$	$\rho_1 = 1$				
$u_2 = 0$	$v_2 = -0.3$	$u_1 = 0$	$v_1 = -0.4$			$J_{2,1}^-$	
$p_3 = 0.4$	$\rho_3 = 1.0625$	$p_4 = 0.4$	$\rho_4 = 0.5197$	$\overleftarrow{S}_{3,2}$			$\overrightarrow{R}_{4,1}$
$u_3 = 0$	$v_3 = 0.2145$	$u_4 = 0$	$v_4 = -1.1259$			$J_{3,4}^-$	
Test 5.13	Lax Configuration 19						
$p_2 = 1$	$\rho_2 = 2$	$p_1 = 1$	$\rho_1 = 1$				
$u_2 = 0$	$v_2 = -0.3$	$u_1 = 0$	$v_1 = 0.3$			$J_{2,1}^+$	
$p_3 = 0.4$	$\rho_3 = 1.0625$	$p_4 = 0.4$	$\rho_4 = 0.5197$	$\overleftarrow{S}_{3,2}$			$\overrightarrow{R}_{4,1}$
$u_3 = 0$	$v_3 = 0.2145$	$u_4 = 0$	$v_4 = -0.4259$			$J_{3,4}^-$	



provided by FOWENO compared to standard WENO.

Table 5.4 shows the CPU time rates for Test 5.9. Again W3C2 is the cheapest one and its CPU time is takes as the reference. For 3rd-order methods, FOW3R3 is the most expensive method. However, for 5th-order methods FOW5L5 is the cheapest one and W5C4, the most expensive one.

W3R3	W3C2	W3L3	W5R3	W5C4	W5L5
2.5269	1.0000	1.1228	4.7006	5.5358	3.715
FOW3R3	FOW3C2	FOW3L3	FOW5R3	FOW5C4	FOW5L5
2.9967	1.2697	1.8280	4.0197	5.1386	3.3760

Table 5.4: CPU time rates for 2D numerical solutions of Test 5.9.

5.5 Comparison of errors and efficiency

Throughout this work, the numerical results obtained with FOWENO-CAT and ACAT methods have been compared with those obtained with some other WENO-based methods but not between them. Moreover, a qualitative point of view has been used to compare the numerical solutions provided by different methods and efficiency plots have not been shown so far. The objective of this section is three-fold:

- To compare FOWENO-CAT and ACAT methods between them.
- To compare the different methods from a quantitative point of view.
- To compare the efficiency of the methods.

Nevertheless, these objectives are no easy to achieve due to the following reasons:

- While ACAT and FOWENO-CAT methods are of even order, WENO-RK or WENO-LAT methods are of odd order.
- The main advantage of methods based on CAT are the possibility of considering larger time steps, so that if comparisons are performed with $CFL = 0.5$ or lower, they are less efficient than other methods, while comparisons performed with CFL close to 1 are not fair for methods that are not stable or oscillatory for those values of the CFL parameter.

- The implementation of CAT, FOWENO-CAT, or ACAT methods carried out to compute the numerical results shown in this work is not optimal and does not take advantage of the potentiality of these methods due to the facts that they are highly parallelizable and do not need the storage of intermediate temporal stages.

Therefore, the conclusions drawn from the comparisons of this section have to be considered as preliminary: a more rigorous and systematic comparison of optimized implementation in GPU architectures is among the lines to be developed in the short future.

In this section we consider again the linear transport equation, Burgers equation, and the 1D and 2D Euler equations. Since the graphs of the numerical solutions obtained with the different methods have been already shown in previous chapters, we only consider here error tables (computed using the exact solution if available or reference solutions obtained with a fine mesh if not) and efficiency curves.

5.5.1 1D Scalar equations

5.5.1.1 Test 5.14. Linear transport equation

Let us consider the linear transport equation

$$u_t + u_x = 0 \quad (5.5.1)$$

with the piecewise continuous initial condition

$$u_0(x) = \begin{cases} 1 & \text{if } \frac{1}{2} \leq x \leq 1; \\ 0 & \text{if } 0 \leq x < \frac{1}{2} \text{ or } \frac{3}{2} < x \leq 2; \\ -1 & \text{if } 1 < x \leq \frac{3}{2}. \end{cases} \quad (5.5.2)$$

We solve in the spatial interval $[0, 2]$, from time $t = 0$ to $t = 4$, periodic boundary conditions, $N = \{50, 100, 200, 400, 800, 1600\}$ point meshes and $\text{CFL} = \{0.5, 0.9\}$. Table 5.5 shows the error in L^1 -norm provided by the numerical solutions of WENO-CAT, WENO-LAT, WENO-RK FOWENO-CAT, FOWENO-LAT, FOWENO-RK and ACAT methods. The reference solution is the exact one. Efficiency plots are shown in Figures 5.23 and 5.24.

The following conclusions can be drawn:

- ACAT methods give the lower errors for both $\text{CFL} = 0.5$ and 0.9 , although it is only second order accurate.
- The error of all methods increase when CFL goes from 0.5 to 0.9 , except ACAT2 and ACAT4.

- The error corresponding to FOWENO implementations are always lower than their WENO counterparts.
- For CFL = 0.5 ACAT2 is the most efficient method among those whose order of accuracy is 2 or 3 followed by FOWENO-CAT methods. FOWENO5L5 is the more efficient among those whose order is 3 or 5 followed by ACAT4.
- For CFL = 0.9 ACAT2 and ACAT4 are the most efficient methods.

Since the solution of this problem is piecewise constant, ACAT4 reduces to second order close to the discontinuities so that its efficiency and errors are almost identical to those of ACAT2. However, the ACAT4 advantages over ACAT2 are relevant when solving smooth solutions that involve critical points as test 4.3.1.1 in section 4.3.

5.5.1.2 Test 5.15. Burgers Equation

Let us consider now Burgers equation with initial condition

$$u_0(x) = \frac{1}{2} \sin(\pi x). \tag{5.5.3}$$

We solve numerically this problem in the spatial interval $[0, 2]$ using periodic boundary conditions from $t = 0$ to $t = 1.25$. Table 5.6 shows the errors in L^1 -norm provided by WENO-CAT, WENO-LAT, WENO-RK FOWENO-CAT, FOWENO-LAT, FOWENO-RK and ACAT methods using $N = \{50, 100, 200, 400, 800, 1600\}$ point mesh and CFL = $\{0.5, 0.9\}$. The reference solution is provided by a first-order method using a 25000-point mesh. Efficiency plots are shown in Figures 5.25 and 5.26.

The following conclusions can be drawn:

- In this case, the errors have not a significant increase when the CFL parameter goes from 0.5 to 0.9, except for WENO3-CAT2.
- The errors corresponding to FOWENO implementations are again lower than their WENO counterparts.
- ACAT2 methods give the lower errors for both CFL = 0.5 and 0.9 among all the second and third order methods.
- ACAT4 is the most accurate method but also the less efficient one due to the facts that WENO methods behave correctly with CFL=0.9 in this case and to the non-optimal implementation.



CFL=0.5							
N	W3C2	W3L3	W3R3	W5C4	W5L5	W5R3	ACAT2
50	0.4997	0.4957	0.4766	0.2580	0.2647	0.2567	0.1433
100	0.2703	0.2673	0.2628	0.1451	0.1499	0.1446	0.0715
200	0.1637	0.1612	0.1589	0.0815	0.0829	0.0816	0.0357
400	0.0986	0.0968	0.0956	0.0457	0.0461	0.0459	0.0178
800	0.0593	0.0579	0.0573	0.0257	0.0258	0.0258	0.0089
1600	0.0357	0.0346	0.0342	0.0146	0.0145	0.0145	0.0044
CFL=0.9							
N	FOW3C2	FOW3L3	FOW3R3	FOW5C4	FOW5L5	FOW5R3	ACAT4
50	0.3976	0.3874	0.3669	0.1988	0.2166	0.2055	0.1426
100	0.1987	0.1922	0.1951	0.1066	0.1135	0.1149	0.0719
200	0.1212	0.1132	0.1124	0.0579	0.0599	0.0645	0.0359
400	0.0719	0.0647	0.0646	0.0317	0.0321	0.0364	0.0179
800	0.0419	0.0380	0.0372	0.0174	0.0174	0.0207	0.0089
1600	0.0252	0.0217	0.0215	0.0096	0.0095	0.0118	0.0044
CFL=0.9							
N	W3C2	W3L3	W3R3	W5C4	W5L5	W5R3	ACAT2
50	1.0397	1.0061	0.5407	0.4751	0.6666	0.3109	0.1216
100	0.8768	0.8790	0.2990	0.3683	0.4822	0.2840	0.0662
200	0.6126	0.6485	0.1740	0.2353	0.3285	0.2403	0.0350
400	0.4065	0.4568	0.1013	0.1640	0.2300	0.1745	0.0182
800	0.2947	0.3087	0.0593	0.1036	0.1578	0.1310	0.0093
1600	0.2063	0.2152	0.0348	0.0567	0.1125	0.0902	0.0047
CFL=0.9							
N	FOW3C2	FOW3L3	FOW3R3	FOW5C4	FOW5L5	FOW5R3	ACAT4
50	1.0468	1.0021	0.3959	0.2540	0.5608	0.2947	0.1271
100	0.8262	0.8236	0.2030	0.1354	0.4184	0.1702	0.0703
200	0.5821	0.5793	0.1170	0.0738	0.2928	0.0968	0.0382
400	0.4032	0.4077	0.0673	0.0399	0.1954	0.0569	0.0200
800	0.2865	0.2891	0.0389	0.0216	0.1417	0.0310	0.0102
1600	0.1974	0.2022	0.0225	0.0117	0.1027	0.0181	0.0051

Table 5.5: Test 5.14. Linear transport equation with initial condition 5.5.2: errors in L^1 -norm for CFL= {0.5,0.9} and $t = 4$.

CFL=0.5							
N	W3C2	W3L3	W3R3	W5C4	W5L5	W5R3	ACAT2
50	0.00632	0.00633	0.00691	0.00334	0.00336	0.00367	0.00487
100	0.00142	0.00143	0.00149	0.00024	0.00024	0.00030	0.00079
200	0.00062	0.00063	0.00066	0.00012	0.00012	0.00015	0.00029
400	0.00026	0.00026	0.00028	0.00006	0.00007	0.00007	0.00011
800	0.00011	0.00012	0.00012	0.00003	0.00003	0.00003	0.00004
1600	0.00005	0.00005	0.00005	0.00001	0.00001	0.00001	0.00001
CFL=0.9							
N	FOW3C2	FOW3L3	FOW3R3	FOW5C4	FOW5L5	FOW5R3	ACAT4
50	0.00494	0.00506	0.00563	0.00277	0.00277	0.00300	0.00286
100	0.00113	0.00113	0.00114	0.00022	0.00023	0.00032	0.00037
200	0.00045	0.00045	0.00048	0.00011	0.00011	0.00014	0.00016
400	0.00019	0.00019	0.00020	0.00005	0.00006	0.00007	0.00006
800	0.00008	0.00008	0.00009	0.00003	0.00003	0.00003	0.00002
1600	0.00004	0.00004	0.00004	0.00001	0.00001	0.00001	0.00001
CFL=0.9							
N	W3C2	W3L3	W3R3	W5C4	W5L5	W5R3	ACAT2
50	0.00736	0.00751	0.00738	0.00308	0.00310	0.00373	0.00435
100	0.00211	0.00187	0.00158	0.00020	0.00024	0.00030	0.00064
200	0.00129	0.00086	0.00070	0.00011	0.00012	0.00014	0.00025
400	0.00062	0.00036	0.00029	0.00006	0.00006	0.00007	0.00010
800	0.00040	0.00016	0.00012	0.00003	0.00003	0.00003	0.00003
1600	0.00016	0.00007	0.00005	0.00001	0.00001	0.00001	0.00001
CFL=0.9							
N	FOW3C2	FOW3L3	FOW3R3	FOW5C4	FOW5L5	FOW5R3	ACAT4
50	0.00597	0.00613	0.00618	0.00244	0.00246	0.00300	0.00247
100	0.00138	0.00103	0.00121	0.00012	0.00015	0.00032	0.00024
200	0.00050	0.00043	0.00046	0.00007	0.00007	0.00014	0.00010
400	0.00032	0.00021	0.00020	0.00004	0.00005	0.00006	0.00004
800	0.00019	0.00011	0.00009	0.00003	0.00003	0.00003	0.00002
1600	0.00012	0.00005	0.00004	0.00001	0.00001	0.00001	0.00000

Table 5.6: Test 5.15. Burgers equations with initial conditions (5.5.3): errors in L^1 -norm for CFL= {0.5, 0.9} and $t = 1.25$.



5.5.2 Test 5.16. 1D Euler equations: the Sod shock tube problem

We consider again the 1D Euler equations with initial condition:

$$(\rho, v, p) = \begin{cases} (1, 0, 1) & \text{if } x < 1/2, \\ (0.125, 0, 0.1) & \text{if } x > 1/2. \end{cases} \quad (5.5.4)$$

We solve numerically this problem in the spatial interval $[0, 1]$ using inflow-outflow boundary conditions and $t = 0.25$. Table 5.7 shows the errors in L^1 -norm corresponding to the numerical solutions provided by WENO-CAT, WENO-LAT, WENO-RK FOWENO-CAT, FOWENO-LAT, FOWENO-RK and ACAT methods using $N = \{50, 100, 200, 400, 800, 1600\}$ point-meshes and $CFL = 0.5$. The exact solution is provided by the HE-E1RPEXACT solver: see [3].

The following conclusions can be drawn:

- The error corresponding to FOWENO implementations are slightly better than their WENO counterparts for second and third order.
- For fourth and fifth order solutions, there is not significant differences between FOWENO-APT (i.e. FOWENO-CAT and FOWENO-LAT) and WENO-APT. Meanwhile, FOWENO-RK works better than WENO-RK.
- FOWENO-APT methods are the most efficient ones.
- The errors corresponding to ACAT2 and FOWENO3-APT are similar but ACAT2 is again the fastest method.

5.5.3 Test 5.17. 2D Euler equations: Lax configuration 6

Let us consider finally the two-dimensional Euler equations of gas dynamics (5.4.5) with the Lax configuration 6 presented in [66]. We solve numerically this problem in the spatial interval $[0, 1] \times [0, 1]$, $CFL = 0.475$, inflow-outflow boundary conditions and $t = 0.3$. Table 5.8 shows the error in L^1 -norm corresponding to the solutions provided by WENO-CAT, WENO-LAT, WENO-RK FOWENO-CAT, FOWENO-LAT, FOWENO-RK and ACAT methods. The reference solution is computed using a 3200×3200 -point mesh and $CFL = 0.475$.

The following conclusions can be drawn:

- The errors corresponding to FOWENO implementations are again lower than their WENO counterparts for all orders.
- ACAT4 is most accurate method in all the variables, followed by the fourth and fifth order FOWENO methods.

	ρ	ρu	E	ρ	ρu	E	ρ	ρu	E
N	W3C2			W3L3			W3R3		
50	0.0177	0.3443	0.8752	0.0176	0.3445	0.8752	0.0175	0.3444	0.8752
100	0.0100	0.3360	0.8565	0.0097	0.3361	0.8565	0.0096	0.3361	0.8565
200	0.0055	0.3307	0.8586	0.0053	0.3308	0.8586	0.0053	0.3308	0.8586
400	0.0031	0.3282	0.8596	0.0030	0.3283	0.8596	0.0030	0.3283	0.8596
800	0.0019	0.3271	0.8601	0.0018	0.3271	0.8601	0.0018	0.3271	0.8601
1600	0.0012	0.3265	0.8604	0.0012	0.3265	0.8604	0.0012	0.3265	0.8604
N	FOW3C2			FOW3L3			FOW3R3		
50	0.0164	0.3435	0.8752	0.0160	0.3436	0.8752	0.0158	0.3436	0.8752
100	0.0090	0.3356	0.8565	0.0087	0.3357	0.8565	0.0085	0.3356	0.8565
200	0.0049	0.3305	0.8586	0.0047	0.3306	0.8586	0.0047	0.3306	0.8586
400	0.0027	0.3281	0.8596	0.0026	0.3282	0.8596	0.0026	0.3282	0.8596
800	0.0016	0.3270	0.8601	0.0016	0.3270	0.8601	0.0016	0.3270	0.8601
1600	0.0011	0.3265	0.8604	0.0011	0.3265	0.8604	0.0010	0.3265	0.8604
N	W5C4			W5L5			W5R3		
50	0.0130	0.3417	0.8753	0.0130	0.3417	0.8753	0.0127	0.3416	0.8753
100	0.0078	0.3347	0.8565	0.0078	0.3347	0.8565	0.0076	0.3346	0.8565
200	0.0043	0.3301	0.8586	0.0043	0.3301	0.8586	0.0042	0.3301	0.8586
400	0.0023	0.3279	0.8596	0.0023	0.3280	0.8596	0.0023	0.3279	0.8596
800	0.0014	0.3269	0.8601	0.0014	0.3269	0.8601	0.0014	0.3269	0.8601
1600	0.0009	0.3265	0.8604	0.0009	0.3265	0.8604	0.0010	0.3265	0.8604
N	FOW5C4			FOW5L5			FOW5R3		
50	0.0136	0.3412	0.8753	0.0134	0.3412	0.8753	0.0129	0.3411	0.8753
100	0.0078	0.3344	0.8565	0.0077	0.3344	0.8565	0.0075	0.3344	0.8565
200	0.0042	0.3300	0.8586	0.0042	0.3300	0.8586	0.0040	0.3300	0.8586
400	0.0023	0.3279	0.8596	0.0023	0.3279	0.8596	0.0022	0.3279	0.8596
800	0.0014	0.3269	0.8601	0.0014	0.3269	0.8601	0.0014	0.3269	0.8601
1600	0.0009	0.3264	0.8605	0.0009	0.3264	0.8604	0.0009	0.3264	0.8604
N	ACAT2			ACAT4					
50	0.0166	0.3436	0.8752	0.0153	0.3429	0.8752			
100	0.0096	0.3356	0.8565	0.0088	0.3352	0.8565			
200	0.0053	0.3306	0.8586	0.0048	0.3304	0.8586			
400	0.0029	0.3282	0.8596	0.0026	0.3281	0.8596			
800	0.0018	0.3271	0.8601	0.0015	0.3270	0.8601			
1600	0.0011	0.3265	0.8604	0.0010	0.3265	0.8604			

Table 5.7: Test 5.16. 1D Euler equations: Sod problem. Errors in L^1 -norm for ρ , ρu and E , using CFL= 0.5 and $t = 0.25$.



- ACAT2 is the most accurate among the second and third order methods and it is again the fastest one.

	ρ	ρu_x	ρu_y	E	ρ	ρu_x	ρu_y	E
N	W3C2				W3L3			
50	0.1171	0.1157	0.0803	0.0900	0.1171	0.1157	0.0803	0.0901
100	0.0774	0.0779	0.0521	0.0686	0.0774	0.0779	0.0524	0.0686
200	0.0486	0.0465	0.0341	0.0474	0.0485	0.0466	0.0342	0.0474
400	0.0287	0.0268	0.0208	0.0275	0.0283	0.0268	0.0207	0.0275
	W3R3				ACAT2			
50	0.1168	0.1153	0.0802	0.0899	0.0986	0.0935	0.0693	0.0917
100	0.0772	0.0777	0.0523	0.0685	0.0633	0.0603	0.0433	0.0626
200	0.0485	0.0466	0.0341	0.0473	0.0371	0.0327	0.0264	0.0389
400	0.0287	0.0268	0.0208	0.0275	0.0209	0.0165	0.0146	0.0216
	FOW3C2				FOW3L3			
50	0.1013	0.0998	0.0701	0.0829	0.1013	0.0999	0.0701	0.0831
100	0.0642	0.0630	0.0436	0.0600	0.0643	0.0630	0.0436	0.0601
200	0.0381	0.0350	0.0267	0.0387	0.0381	0.0350	0.0268	0.0388
400	0.0204	0.0205	0.0190	0.0211	0.0204	0.0205	0.0190	0.0211
	FOW3R3				ACAT4			
50	0.1013	0.0998	0.0701	0.0831	0.0784	0.0834	0.0592	0.0715
100	0.0643	0.0631	0.0437	0.0602	0.0530	0.0501	0.0330	0.0522
200	0.0380	0.0350	0.0268	0.0388	0.0245	0.0264	0.0208	0.0350
400	0.0203	0.0205	0.0191	0.0201	0.0131	0.0124	0.0162	0.0167
	W5C4				W5L5			
50	0.0804	0.0790	0.0593	0.0731	0.0804	0.0790	0.0593	0.0731
100	0.0501	0.0495	0.0353	0.0555	0.0501	0.0495	0.0353	0.0555
200	0.0285	0.0264	0.0208	0.0357	0.0285	0.0264	0.0208	0.0357
400	0.0146	0.0134	0.0172	0.0177	0.0146	0.0134	0.0172	0.0177
	W5R3				FOW5C4			
50	0.0814	0.0806	0.0591	0.0707	0.0814	0.0804	0.0591	0.0706
100	0.0506	0.0498	0.0354	0.0524	0.0505	0.0498	0.0354	0.0524
200	0.0288	0.0263	0.0237	0.0320	0.0285	0.0263	0.0207	0.0320
400	0.0147	0.0135	0.0182	0.0178	0.0145	0.0130	0.0108	0.0178
	W5L3				FOW5R3			
50	0.0815	0.0804	0.0591	0.0706	0.0719	0.0703	0.0530	0.0672
100	0.0505	0.0498	0.0354	0.0523	0.0432	0.0415	0.0303	0.0473
200	0.0285	0.0262	0.0207	0.0318	0.0231	0.0206	0.0169	0.0273
400	0.0144	0.0130	0.0108	0.0175	0.0145	0.0130	0.0102	0.0172

Table 5.8: Test 5.17. 2D Euler equations. Lax problem 6. Errors in L^1 -norm for ρ , ρu_x , ρu_y and E , using CFL= 0.475 and $t = 0.3$.



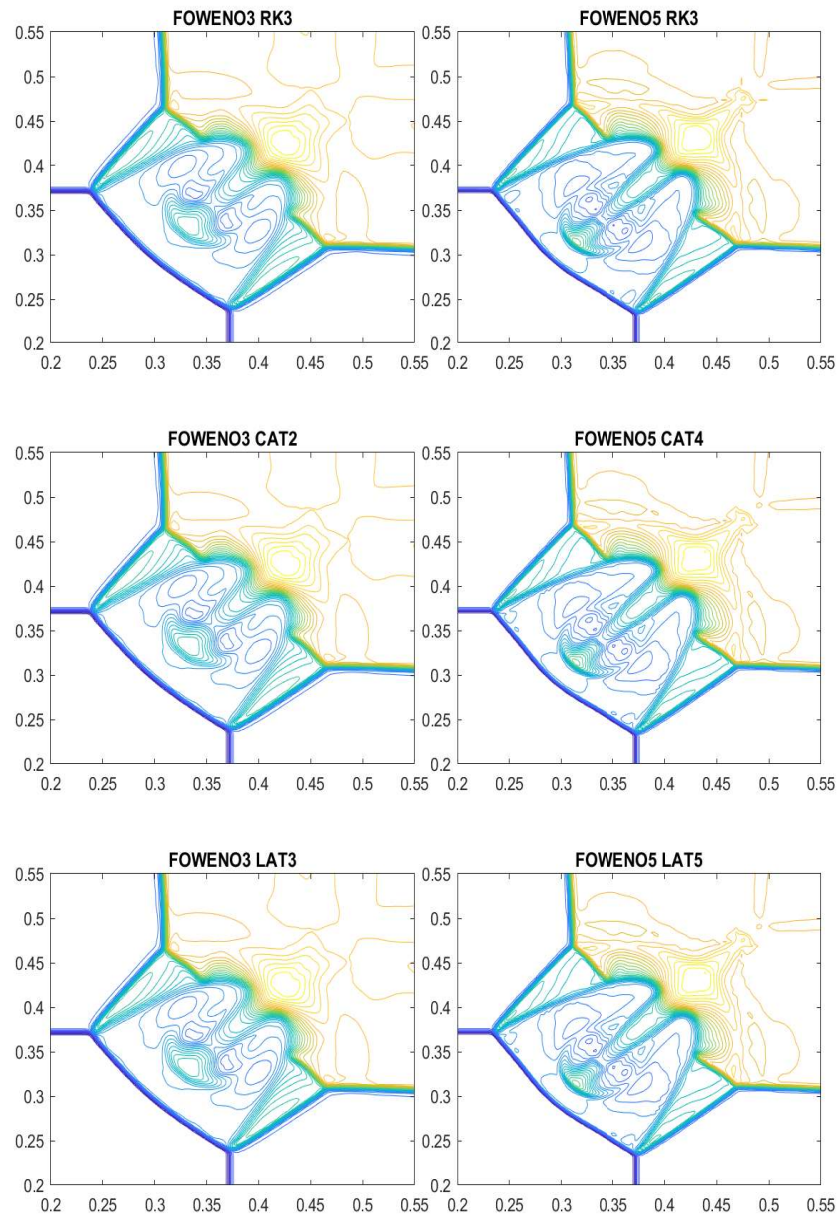


Figure 5.16: Test 5.8. 2D Euler equations. Lax configuration 3: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.

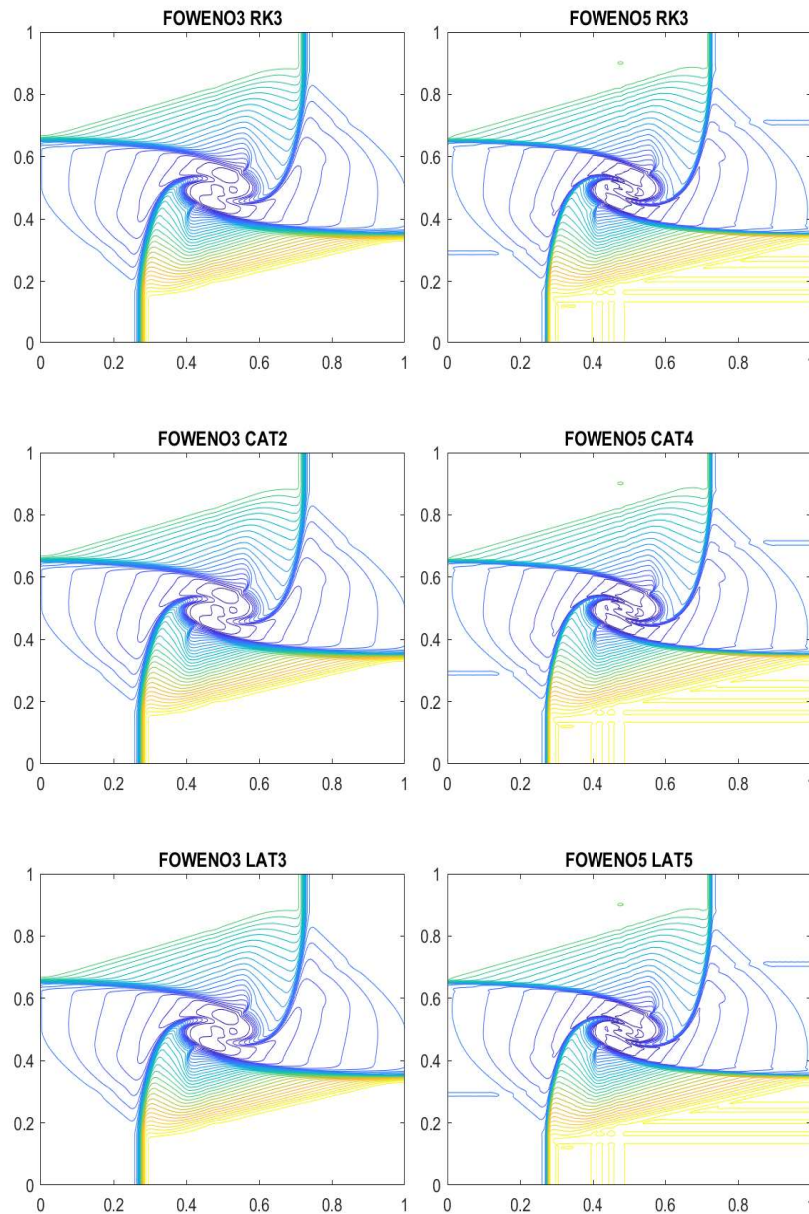


Figure 5.17: Test 5.9. 2D Euler equations. Lax configuration 6: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.

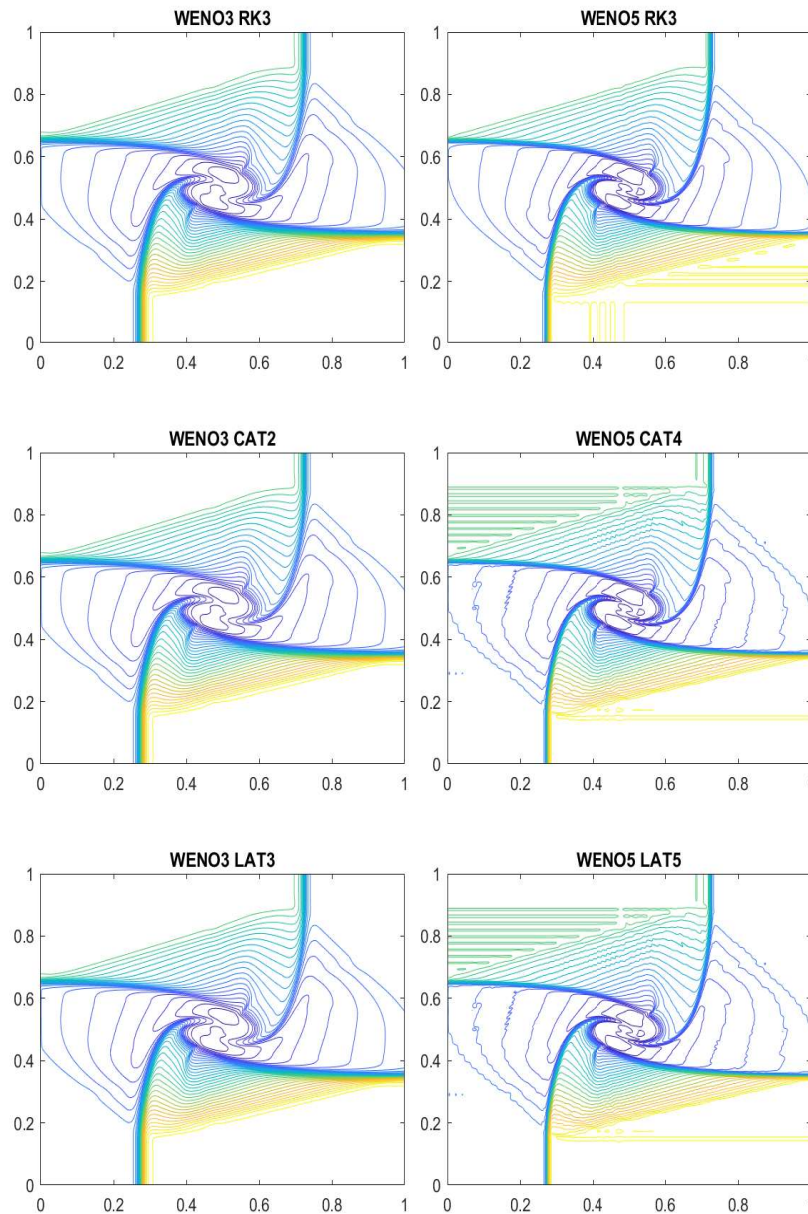


Figure 5.18: Test 5.9. 2D Euler equations. Lax configuration 6: density computed with WENO-RK, WENO-CAT and WENO-LAT.

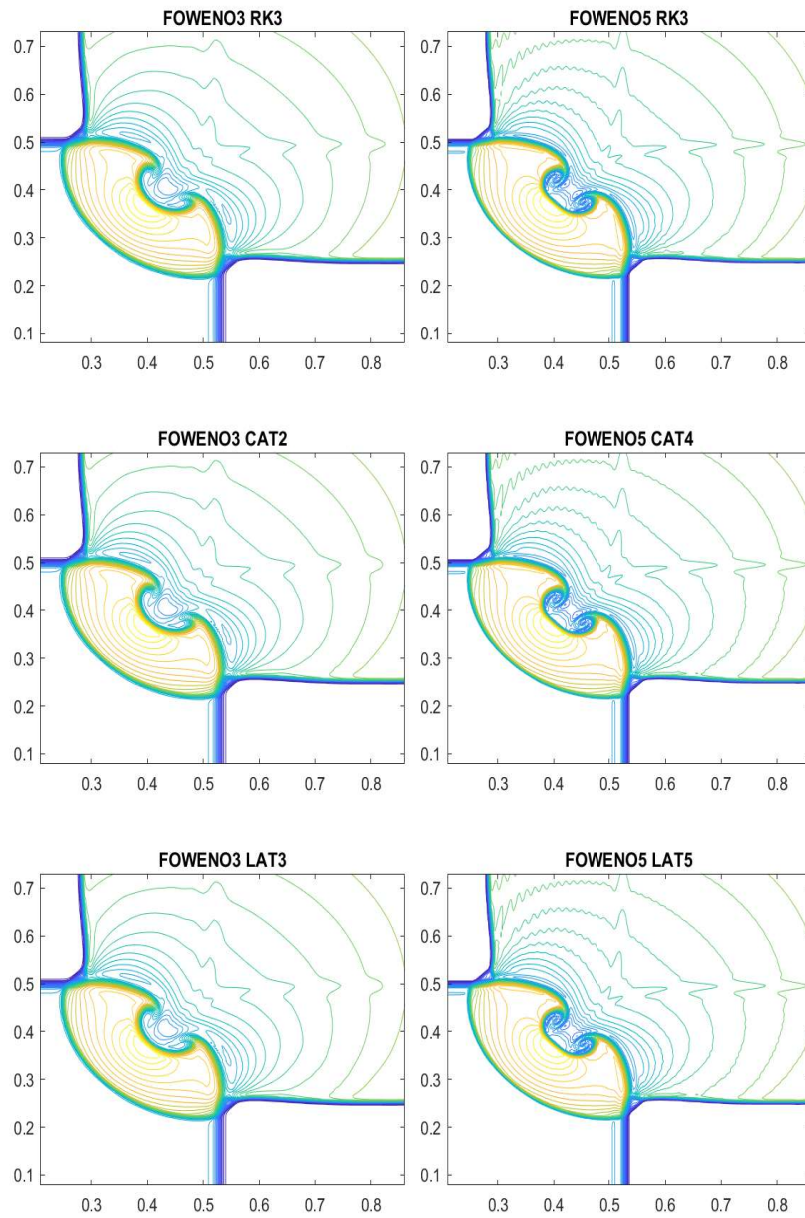


Figure 5.19: Test 5.10. 2D Euler equations. Lax configuration 11: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.

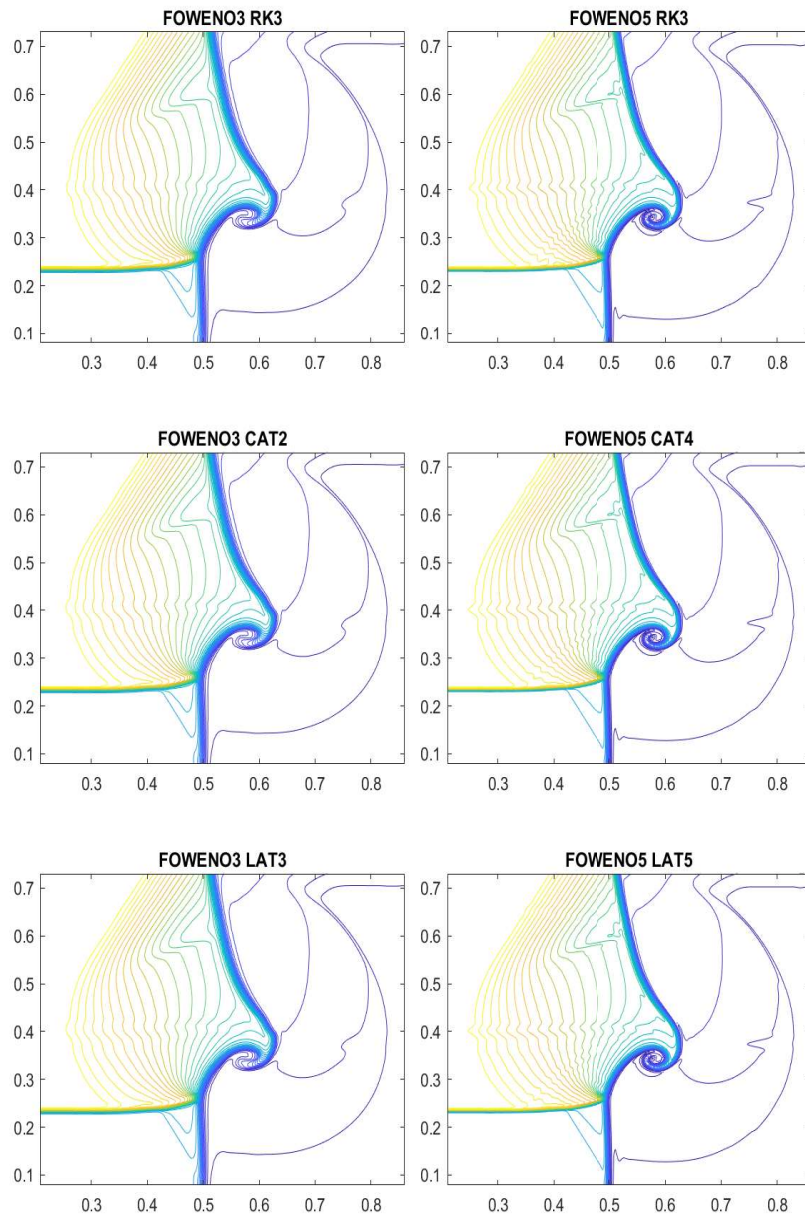


Figure 5.20: Test 5.11. 2D Euler equations. Lax configuration 13: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.

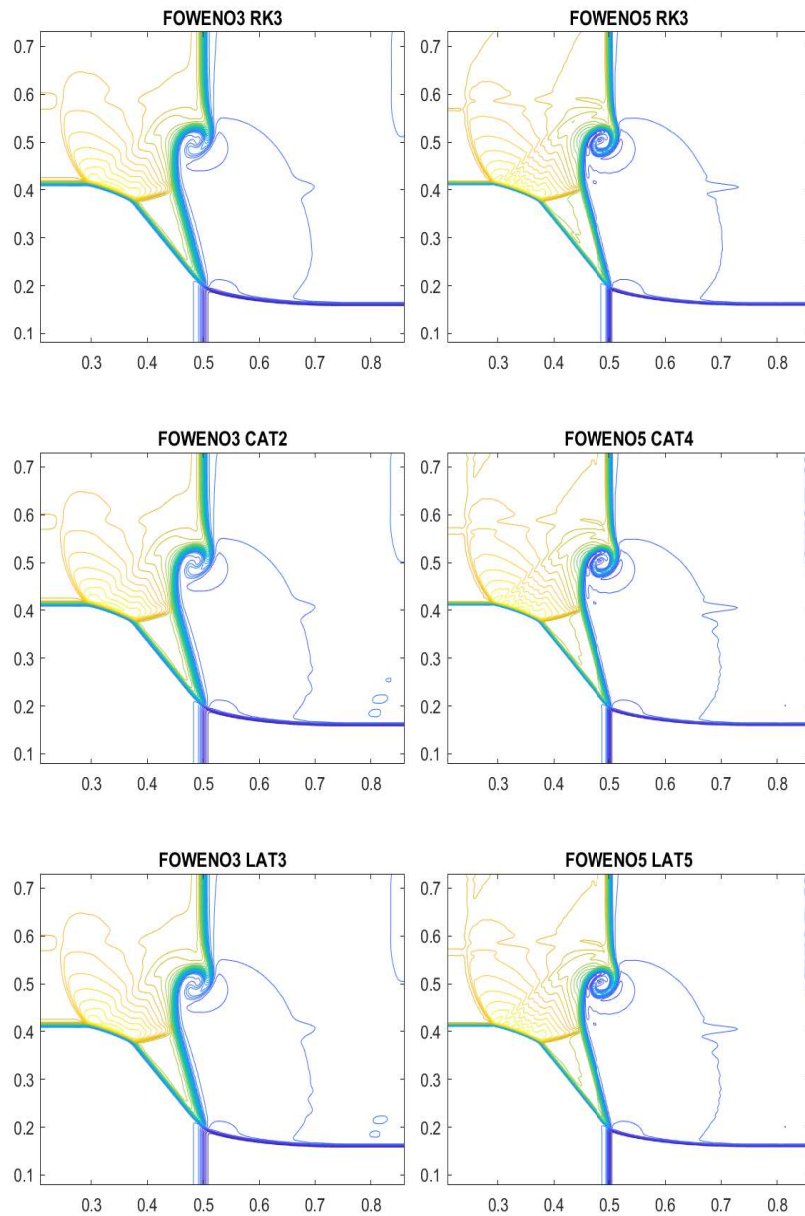


Figure 5.21: Test 5.12. 2D Euler equations. Lax configuration 17: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.

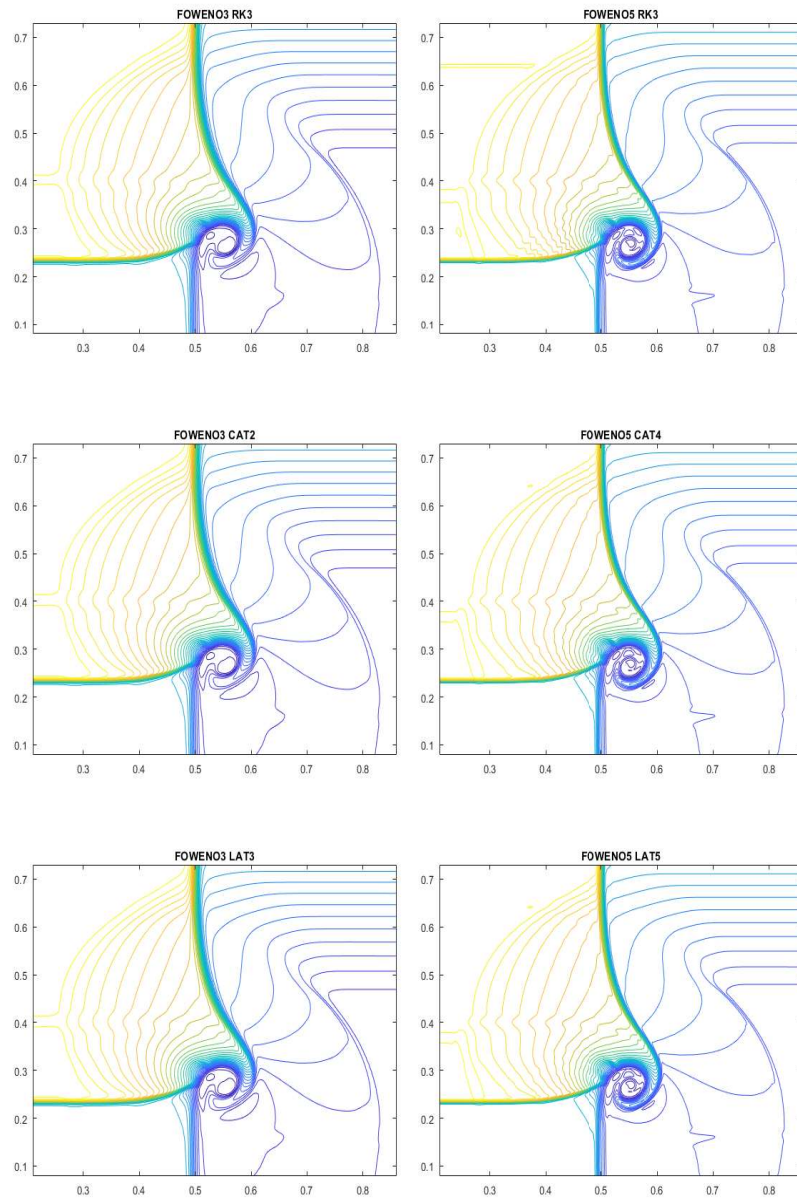


Figure 5.22: Test 5.13. 2D Euler equations. Lax configuration 19: density computed with FOWENO-RK, FOWENO-CAT and FOWENO-LAT.

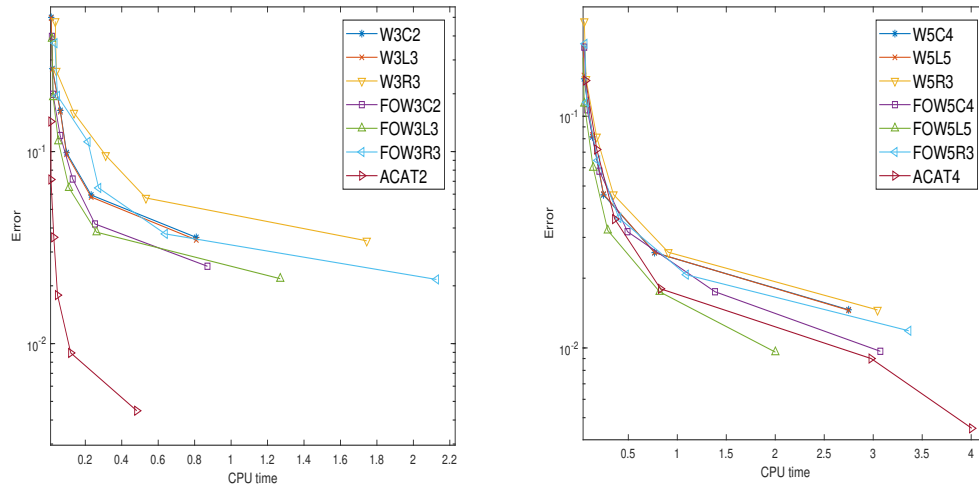


Figure 5.23: Test 5.14. Linear transport equation with initial condition 5.5.2: efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 4$ and CFL= 0.5. Second and third-order methods (*left*) and fourth or fifth-order methods (*right*).

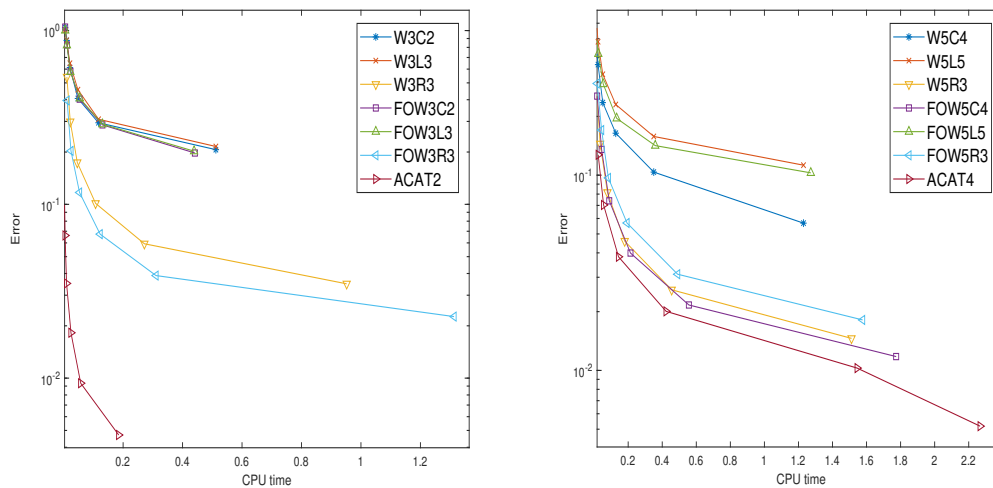


Figure 5.24: Test 5.14. Linear transport equation with initial condition 5.5.2: efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 4$ and CFL= 0.9. Second or third-order methods (*left*) and fourth or fifth-order methods (*right*).



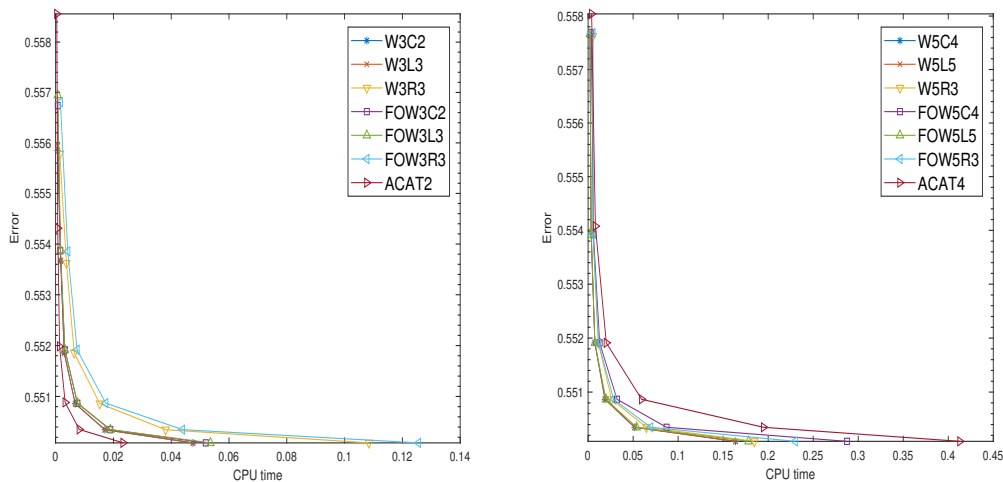


Figure 5.25: Test 5.15. Burgers equation with initial conditions (5.5.3): efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 1.25$ and $CFL = 0.5$. Second and third-order methods (*left*) and fourth or fifth-order methods (*right*).

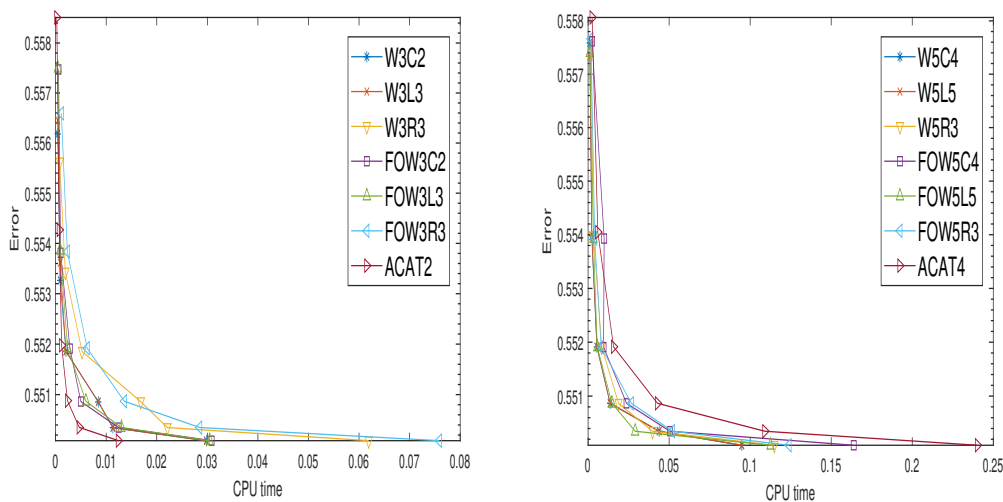


Figure 5.26: Test 5.15. Burgers equation with initial conditions (5.5.3): efficiency plot for WENO-CAT, WENO-LAT, WENO-RK, FOWENO-CAT, FOWENO-LAT, FOWENO-RK, and ACAT solutions at $t = 1.25$ and $CFL = 0.9$. Second and third-order methods (*left*) and fourth or fifth-order methods (*right*).

Chapter 6

Conclusions and future work

In this thesis, a new family of high-order numerical methods for systems of conservation laws is introduced. The methods of this family are based on Approximate Taylor techniques and they have been described in Chapter 3. In that chapter:

- High-order Lax-Wendroff methods for the linear transport equation are reviewed, including the study of the order, a heuristic study of the L^2 -stability, and the computation and properties of the coefficients.
- Next, these methods are extended to nonlinear conservation laws with arbitrary even order $2p$ of accuracy, the so-called Compact Approximate Taylor (CAT) methods. Unlike previous applications of Taylor methods to conservation laws, CAT methods have $(2p+1)$ -point centered stencils, like Lax-Wendroff methods for linear problems. Moreover, since they inherit the stability properties of Lax-Wendroff methods, they are linearly L^2 -stable under the standard CFL condition.
- Two shock-capturing techniques are considered to cure the spurious oscillations appearing close to discontinuities: in FL-CAT2 method the second-order CAT method has been combined with a robust first-order method on the basis of a standard flux limiter; in CAT-WENO methods, WENO reconstructions are used to compute the first time derivative of the solution.
- These new methods are compared in a number of test cases with WENO-RK methods (Finite Differences WENO reconstructions in space, TVD-RK in time) and with LAT-WENO methods introduced in [10] (Finite Differences WENO reconstruction for the first time derivative, Approximate Taylor in time). The linear transport equation, Burgers equation, the 1D compressible Euler and the MHD equations are considered. For $CFL \leq 0.5$ all the numerical methods work correctly, and the results obtained with all the methods using WENO reconstructions are similar, while the FL-CAT method is more diffusive as expected. Nevertheless, CAT methods are more expensive in computational time and number

of operations due to its local character (FL-CAT is less expensive than WENO-CAT as reconstructions are avoided). However, the extra computational cost of CAT methods is compensated by the fact that they still give good solutions with CFL values close to 1.

In addition, two proper and original ways to keep under control the spurious oscillations generated by CAT methods close to shocks are applied. First, the Adaptive Compact Approximate Taylor Method is described in Chapter 4. In that chapter:

- An order adaptive version of the Compact Approximate Taylor methods (ACAT) is presented, including the flux-limiter technique (FL-CAT2 or ACAT2) for the low order scheme; a new family of high-order smoothness indicators is introduced that are able to detect the smoothness of the numerical solution in centered stencils; the theoretical analysis of the order of these smoothness indicators is performed; the extension to 2D problems of both CAT and ACAT methods is presented.
- These new shock-capturing Methods are compared in a number of test cases with WENO-RK methods (Finite Differences WENO reconstructions in space, TVD-RK in time). The linear transport equation, Burgers equation, the 1D and 2D compressible Euler equations are considered. For $CFL \leq 0.5$ all the numerical methods work correctly, and the results obtained using WENO reconstructions are similar to the ACAT that, again, allow one to select larger time steps.

And finally, the combination of LAT and CAT methods with fast and optimal WENO reconstructions is studied in Chapter 5. In that chapter:

- Several shock-capturing high-order finite difference methods for 1D and 2D systems of conservation laws are presented and compared in a number of test cases. Two different high-order reconstruction operators are considered: standard WENO and FOWENO operators. The latter combines the use of fast smooth indicators (that coincide with the original smooth indicators in the third-order case) and the computation of optimal weights that allow one to preserve the accuracy of the reconstructions close to critical point regardless of their order. For the best of our knowledge, this is the first time that these two techniques have been combined.
- Concerning the time discretization TVDRK, LAT and CAT methods are considered.
- The numerical tests show that, for third-order reconstructions, FOWENO is more expensive than WENO due to the computation of the optimal weights, as it happens for CWENO [71], M-WENO [69] and other WENO versions. Nevertheless this extra cost is relatively small and it is compensated by the quality of the solutions close to critical points. For order 5 or higher, methods based on FOWENO reconstructions give better solutions and are cheaper than those based on standard WENO: the

extra cost due to the computation of the optimal weights is compensated by the lower cost required by the computation of the smoothness indicators.

Concerning the time discretization, the following conclusions can be drawn from the numerical tests:

- CAT2 combined with 3d-order reconstructions is a good choice in 1D and 2D: the quality of the solutions is comparable to those obtained with LAT3 or RK3, but with a significantly lower cost.
 - LAT methods are cheaper for reconstructions of order 7 or higher in 1D and of order 5 or higher in 2d.
 - In some cases, the extra cost of CAT methods can be compensated by the fact that bigger values of the CFL parameter can be taken with good results.
 - For 1D problems, SSPRK3 gives results that are competitive both in quality and computational time but SSPRK4 increases a lot the computational time.
- A subsection is added to this chapter (not included in the article [14]) where we compare the errors and efficiencies of the FOWENO-LAT, FOWENO-CAT, and ACAT methods. This information helps to support the conclusions.

Future developments include:

- Optimized implementation of CAT methods in GPU architectures. The implementation of CAT, FOWENO-CAT, or ACAT methods carried out to compute the numerical results shown in this work is not optimal and does not take advantage of the potentiality of these methods: they are highly parallelizable and do not need the storage of intermediate temporal stages. Therefore, the comparisons of computational costs or efficiency curves shown in the previous chapters lead only to partial conclusions. Next developments include the implementation of the methods in GPU architectures and the systematic comparison between them. The best methods will be considered as candidates to be included in the HySEA package [72] generated by the EDANYA team of the University of Málaga for the simulation of geophysical flows.
- Combination of CAT methods with the MOOD strategy. Instead of using a priori smoothness indicators to cure the spurious oscillations close to discontinuities, the MOOD strategy is based on an a posteriori analysis of the updated numerical solution: see [21], [22], [23],... This analysis is performed at every time step and it is followed by a local recalculation of the solution where it is necessary using a more robust numerical method. Besides the spurious oscillations, this methodology allows one to control aspects such as the positivity of the numerical method. CAT methods are excellent candidates to be combined with this technique, due to their

good stability properties and the minimal size of their stencils. The idea would be to update the numerical solutions at every time step with $CAT2P$. Then, this first numerical solution is analyzed and the cells where wrong solutions are detected (due to spurious oscillations, negative values of densities, NaN results, etc) are marked. Next, the numerical solutions at the marked cells are computed again using now $CAT2(P - 1)$. This new numerical solution is then analyzed and the procedure follows in a recursive way. In the worst-case scenario, the numerical solution will be updated in part of the domain with a robust first order numerical method. This strategy may lead to efficient and robust high-order numerical methods.

- Extension to systems of balance laws. The mathematical models which are at the basis of the HySEA package contain source terms and/or non-conservative products: shallow water models, multilayer models, avalanche models, etc. Therefore the integration of the methods developed here in this package requires their generalization to systems of balance laws as a first step. Moreover, beside the order of accuracy and the stability properties, the well-balanced property of the methods (i.e. their capability of preserving some or all the stationary solutions of the system) play a key role in these applications. The first goal will be to derive high-order numerical methods for the shallow water model that preserve water at rest solutions. Then, more complex systems and more demanding well-balance properties will be addressed.
- Extension to nonconservative hyperbolic systems. The correct definition and approximation of weak solutions for nonconservative systems is one of the major challenges in the field of hyperbolic PDEs: standard finite-difference or finite-volume methods fail in general to converge to the 'correct' weak solutions due to the viscous terms of the numerical methods. Therefore, the only numerical methods for which convergence can be proved or at least observed are free-viscosity methods (such as Glimm's method) or methods in which the numerical viscosity and dispersion are controlled. In this spirit, Well-Controlled Dissipation (WCD) methods, based in Taylor expansions, allow one to correctly capture the shocks in nonconservative problems: see [24]. Although these methods use high-order Taylor developments, they are only first order accurate even in regions where the solution is smooth. The original motivation of this thesis was to develop high-order methods based on Taylor approach that could be easily combined with WCD methods so that shocks are correctly capture but the numerical method is high-order accurate in smoothness regions. The experience acquired in the development of this work will allow us to address this goal in a short future.

Bibliography

- [1] P. Lax and B. Wendroff. Systems of conservation laws. *Communications Pure and Applied Mathematics*, 13(2):217–237, 1960.
- [2] R.J. LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems (Classics in Applied Mathematics)*. Society for Industrial and Applied Mathematics, Philadelphia, PA. USA., 1 edition, 2007.
- [3] E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, third edition, 2009.
- [4] G. Zwas and S. Abarbanel. Third and fourth order accurate schemes for hyperbolic equations of conservation law form. *Mathematics of Computation*, 25(114):229–236, 1971.
- [5] E.F. Toro, R.C. Millington, and L.A.M Nejad. Towards very high order godunov schemes. *Godunov Methods. Theory and Applications E.F. Toro ed., Kluwer/Plenum Academic Publishers*, pages 907–940, 2001.
- [6] V.A. Titarev and E.F. Toro. ADER: Arbitrary high order godunov approach. *Journal of Scientific Computing*, 17:609–618, 2002.
- [7] T. Schwartzkopff, C. D. Munz, and E.F. Toro. A high-order approach for linear hyperbolic systems in 2d. *Journal of Scientific Computing*, 17:231–240, 2002.
- [8] C. Enaux, M. Dumbser, and E.F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227(2):3971–4001, 2008.
- [9] M. Dumbser, D. Balsara, E.F. Toro, and C.D. Munz. A unified framework for the construction of one-step finite-volume and discontinuous galerkin schemes. *Journal of Computational Physics*, 227:8209–8253, 2008.
- [10] D. Zorío, A. Baeza, and P. Mulet. An approximate lax–wendroff-type procedure for high order accurate schemes for hyperbolic conservation laws. *Journal of Scientific Computing*, 71:246–273, 2017.



- [11] J. Qiu and C.-W. Shu. Finite difference weno schemes with lax-wendroff-type time discretizations. *SIAM Journal on Scientific Computing*, 26(6):2185–2198, 2003.
- [12] H. Carrillo and C. Parés. Compact approximate taylor methods for systems of conservation laws. *Journal of Scientific Computing*, 80:1832–1866, 2019.
- [13] H. Carrillo, E. Macca, C. Parés, G. Russo, and D. Zorío. An order-adaptive compact approximation taylor method for systems of conservation laws. *arXiv:2007.01416*.
- [14] H. Carrillo, C. Parés, and D. Zorío. Lax wendroff approximate taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions. *arXiv:2002.084261 [math.NA] 2020*, 2020.
- [15] X.D. Liu S. Osher and T.Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115:200 – 212, 1994.
- [16] G.S. Jiang and C.W. Shu. Efficient implementation of Weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.
- [17] A. Baeza, R. Bürger, P. Mulet, and D. Zorío. On the efficient computation of smoothness indicators for a class of weno reconstructions. *Journal of Scientific Computing*, 80:1240–1263, 2019.
- [18] A. Baeza, R. Bürger, P. Mulet, and D. Zorío. An efficient third-order WENO scheme with unconditionally optimal accuracy. *SIAM Journal on Scientific Computing (To appear)*, 2020.
- [19] A. Baeza, R. Bürger, P. Mulet, and D. Zorío. Weno reconstructions of unconditionally optimal high order. *SIAM Journal on Numerical Analysis*, 57:2760–2784, 2019.
- [20] C. W. Shu and S. Osher. Primitive, conservative and adaptive schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 83:32–78, 1989.
- [21] S. Clain, , S. Diot, and R. Loubère. A high-order finite volume method for systems of conservation laws—multi-dimensional optimal order detection (mood). *Journal of computational Physics*, 230:4028–4050, 2011.
- [22] S. Clain, , S. Diot, and R. Loubère. Multi-dimensional optimal order detection (mood) — a very high-order finite volume scheme for conservation laws on unstructured meshes. *Finite Volumes for Complex Applications VI Problems & Perspectives*, VI:263–271, 2011.
- [23] R. Loubère, M. Dumbser, and S. Diot. A new family of high order unstructured mood and ader finite volume schemes for multidimensional systems of hyperbolic conservation laws. *Communication in Computational Physics*, 16:718–763, 2014.



- [24] A. Beljadid, P. LeFloch, S. Mishra, and C. Parés. Schemes with well-controlled dissipation. hyperbolic systems in nonconservative form. *Communications in Computational Physics*, 21(4):913–946, 2017.
- [25] S. Gottlieb, D. Ketcheson, and C.W. Shu. *Strong Stability Preserving Runge-Kutta and multistep time discretizations*. Word Scientific, 1 edition, 2011.
- [26] A. Harten, B. Engquist, S. Osher, and S. Chakravathy. Uniformly high order accurate essentially non-oscillatory schemes, iii. *Journal of Computational Physics*, 71:231–303, 1987.
- [27] E.F. Toro. Primitive, conservative and adaptive schemes for hyperbolic conservation laws. *Numerical Methods for Wave Propagation. Academic Publishers*, 1:323–385, 1998.
- [28] M. Lukáčová-Medvid’ová and G. Warnecke. Lax-wendroff type second order evolution galerkin methods for multidimensional hyperbolic systems. *East-West J. Numer. Math.*, 8:127–152, 2000.
- [29] H. Hugoniot. Sur la propagation du mouvement dans les coprs et spécialement dans les gaz parfaits. *J. Ecole Polytechnique*, 57:3–97, 1887.
- [30] W. J. M. Rankine. On the thermodynamic theory of waves of finite longitudinal disturbance. *Phil. Trans. Roy. Soc. London*, 160:277–288, 1870.
- [31] A. J. Chorin and J. E. Marsden. *A mathematical introduction to fluid mechanics*. Springer, New York, 3rd edition, 2000.
- [32] C. Hirsch. *Numerical computation of internal and external flows (volume 1): fundamentals of numerical discretization*. John Wiley & Sons, Inc., New York, NY, USA, 1988.
- [33] C. Hirsch. *Numerical computation of internal and external flows (volume 2): computational methods for inviscid and viscous flow*. John Wiley & Sons, Inc., New York, NY, USA, 1988.
- [34] R.J. LeVeque. *Numerical Methods for conservation laws*. Springer Basel AG. lectures in Mathematics ETH Zurich., 2 edition, 1992.
- [35] P. D. Lax. Shock waves and entropy. In E.A. Zarantonello, editor, *Contributions to nonlinear functional analysis*, pages 603–634. Academic Press, 1971.
- [36] O. Oleinik. Discontinuous solutions of nonlinear differential equations. *Amer. Math. Soc. Transl. Ser. 2*, 26:95–172, 1957.

- [37] S.N. Kružkov. First order quasilinear equations with several independent variables. *Mat. Sb. (N.S.)*, 81 (123):228–255, 1970.
- [38] B. Wendroff. The Riemann problem for materials with nonconvex equation of state. *J. Math. Anal. Appl.*, 38:454–466, 1972.
- [39] T.-P. Liu. The entropy condition and the admissibility of shocks. *Journal of Mathematical Analysis and Applications*, 53:78–88, 1976.
- [40] C. M. Dafermos. Polygonal approximations of solutions of the initial value problem for a conservation law. *J. Math. Anal. Appl.*, 38:33–41, 1972.
- [41] H Holden, L. Holden, and R. Høegh-Krohn. A numerical method for first order nonlinear scalar conservation laws in one dimension. *Comput. Math. Appl.*, 15(6-8):595–602, 1988. Hyperbolic partial differential equations. V.
- [42] H. Holden and N.H. Risebro. *Front tracking for hyperbolic conservation laws*, volume 152 of *Applied Mathematical Sciences*. Springer, Heidelberg, second edition, 2015.
- [43] L. C. Evans. *Partial differential equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 1998.
- [44] C. W. Shu. Total-variation-diminishing time discretizations. *SIAM Journal on Scientific and Statistical Computing*, 9(6):1073–1084, 1988.
- [45] S. Gottlieb and C. W. Shu. Total variation diminishing Runge–Kutta schemes. *Mathematics of Computation*, 67(221):73–85, 1998.
- [46] D. Ketcheson. Highly efficient strong stability-preserving runge–kutta methods with low-storage implementations. *SIAM Journal on Scientific Computing*, 30:2113–2136, 01 2008.
- [47] S. Osher. Riemann solvers, the entropy condition, and difference approximations. *SIAM J. Numer. Anal.*, 21(2):217–235, 1984.
- [48] E. Tadmor. Numerical viscosity and the entropy condition for conservative difference schemes. *Math. Comp.*, 43(168):369–381, 1984.
- [49] X-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115:200–212, 1994.
- [50] F. Aràndiga and R. Donat. Nonlinear multiscale decompositions: the approach of A. Harten. *Numer. Algorithms*, 23:175–216, 2000.
- [51] F. Aràndiga, A. Baeza, A.M. Belda, and P. Mulet. Analysis of WENO schemes for full and global accuracy. *SIAM Journal of Numerical Analysis*, 49:893 – 915, 2011.

- [52] C. W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77:439–471, 1988.
- [53] R. D. Richtmyer and K. W. Morton. Difference methods for initial-value problems. *Interscience Tracts in Pure and Appl. Math. Interscience, New York*, (4), 1967.
- [54] R.W. MacCormack. The effect of viscosity in hypervelocity impact cratering. *AIAA Pape, Cincinnati, Ohio*, pages 69–354, 1969.
- [55] B. Fornberg. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of Computation*, 51:699–706, 1988.
- [56] B. Fornberg. Classroom note: calculation of weights in finite difference formulas. *SIAM Review.*, 40:685–691, 1998.
- [57] R.F. Warming and B.J. Hyett. The modified equation approach to the stability and accuracy analysis of finite-difference methods. *Journal of Computational Physics*, 14(2):159–179, 1974.
- [58] F. Kemm. A comparative study of tvd-limiters – well-known limiters and an introduction of new ones. *International Journal for Numerical Methods in Fluids*, 67:404–440, 2010.
- [59] R.J. LeVeque. *Finite volume methods for hyperbolic problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press., 1 edition, 2002.
- [60] C. W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. Technical report, Institute for Computer Applications in Science and Engineering (ICASE), 1997.
- [61] G.A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27(1):1–31, 1978.
- [62] M. Brio and C. Wu. An upwind differencing scheme for the equations of ideal magnetohydrodynamics. *Journal of Computational Physics*, 75:400–422, 1998.
- [63] P.L. Roe. Characteristic-based schemes for the euler equations. *Annu. Rev. Fluid Mech.*, 18:337–365, 1986.
- [64] B. Einfeldt, P.L. Roe, C.D. Munz, and B. Sjogreen. On Godunov-type methods near low densities. *Journal of Computational Physics*, 92:273–295, feb 1991.
- [65] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 1:115–173, 1984.

- [66] P. Lax and Liu Xu-Dong. Solution of two-dimensional riemann problems of gas dynamics by positive schemes. *SIAM Journal on Scientific Computing*, 19F(2):319–340, 1998.
- [67] A. Kurganov and E. Tadmor. Solution of two-dimensional riemann problems for a gas dynamics without riemann problem solvers. *Numer. Methods Partial Differential Equations*, 18:584–608, 2002.
- [68] F. Arándiga, M.C. Martí, and P. Mulet. Weights design for maximal order WENO scheme. *Journal of Scientific Computing*, 60:641 – 659, 2014.
- [69] A. Henrick, T. Aslam, and J. Powers. Mapped weighted essentially non-oscillatory schemes: Achieving optimal order near critical points. *Journal of Computational Physics*, 207(2):542 – 567, 2005.
- [70] N. K. Yamaleev and M.H. Carpenter. A systematic methodology to for constructing high-order energy stable weno schemes. *Journal of Computational Physics*, 11:4248–4272, 2009.
- [71] I. Cravero, G. Puppo, M. Sempliche, and G. Visconti. CWENO: uniform accurate reconstruction for balance laws. *Mathematics of Computation*, 87(312):1689 – 1719, 2018.
- [72] EDANYA. Hysea (hyperbolic systems and efficient algorithms) software. <http://edanya.uma.es/hysea/index.php>, 2020.